Code-aided Iterative Techniques in OFDM Systems

This thesis is submitted in partial fulfilment of the requirements for

Doctor of Philosophy (Ph.D.)

Yu Zhang

Communications Research Group

Department of Electronics

University of York

January 2007

Abstract

Inspired by the 'turbo principle', this thesis deals with two iterative technologies in orthogonal frequency division multiplexing (OFDM) systems: iterative interference cancelation in space-frequency block coded OFDM (SFBC-OFDM) and iterative channel estimation/tracking in OFDM Access (OFDMA) with particular application to Worldwide Inter-operability for Microwave Access (WiMAX) systems.

The linear matched filter (MF) decoding in SFBC-OFDM is simple yet obtains maximum-likelihood (ML) performance based on the assumption that the channel frequency response remains constant within a block. However, frequency response variations gives rise to inter-channel interference (ICI). In this thesis, a parallel interference cancelation (PIC) approach with soft iterations will be proposed to iteratively eliminate ICI in G_4 SFBC-OFDM. Furthermore, the information from outer convolutional decoder is exploited and fed back to aid the inner PIC process to generate more accurate coded bits for the convolutional decoder. Therefore, inner and outer iterations work in a collaborative way to enhance the performance of interference cancelation.

Code-aided iterative channel estimation/tracking has the ability of efficiently improving the quality of estimation/tracking without using additional pilots/training symbols. This technique is particularly applied to OFDMA physical layer of WiMAX systems according to the Institute of Electrical and Electronics Engineers (IEEE) 802.16 standard. It will be demonstrated that the performance of the pilot-based channel estimation in uplink (UL) transmission and the channel tracking based on the preamble symbol in downlink (DL) transmission can be improved by iterating between the estimator and the detector the useful information from the outer convolutional codes.

The above two issues will be discussed in Chapter 5 and Chapter 6, and before this, Chapter 2 to Chapter 4 will introduce some background techniques that are used throughout the thesis.

Acknowledgement

I would like to express my sincere gratitude to my supervisor, Professor Alister G. Burr. Thank you for your heuristic ideas, constructive suggestions, patient instructions and continuous support during my Ph.D research. You have successfully kept my research progress and conducted my work into a right direction. Without your help, my research would never go smoothly, or even be possible. Thank you for your great work and precious time on correcting my English.

I would also like to deliver my grateful thanks to my thesis advisor, Dr. Yuriy, V. Zakharov. Thank you for always being supportive and providing valuable advice on my annual research progress report.

More thanks go to my colleagues in Communications Research Group, for all your assistance in my life and study, which makes me feel like in a big warm family. Particularly, great thanks to Dr. Huiheng Mai, who has once been a consultant of my project and provided me a lot of knowledge on channel estimation. Also, big thanks to Dr. George White, who has helped me in turbo codes. Special thanks to my boyfriend, who has been sharing happiness and facing difficulties with me.

Finally, I would like to give my deepest thankfulness go to my dear parents, who are far away in China, but always care about me, encourage me and try their best to support me. Thank you for your selfless and endless love.

Declaration

Some of the work presented in this thesis has been published at or submitted to academic conferences. These publications are listed at the end of the thesis.

To the best knowledge of the author, all work in this thesis claimed is original. Any research not original is clearly stated, and references and acknowledgements to other researchers have been given as appropriate.

Contents

List of Figures

List of Tables

List of Abbreviations and Acronyms

2D	two-dimension
$2\times1D$	two cascaded one-dimension
3G	third generation
4G	fourth generation
acf	autocorrelation function
AWGN	Additive White Gaussian Noise
BER	bit error rate
BPSK	binary phase-shift keying
BS	base station
BT	British Telecommunications
BW-REQ	bandwidth request
ccf	cross-correlation function
CDMA	code division multiplexing access
COFDM	coded orthogonal frequency division multiplexing
CP	cyclic prefix
DAB	digital audio broadcasting
DFT	discrete Fourier transform
DL	downlink
DSL	digital subscriber line
DSP	digital signal processing
DVB	digital video broadcasting
E	expectation
Eb/N0	bit-energy-to-noise-density ratio
ECC	error-control coding
EM	expectation maximization
FCH	frame control header
FDD	frequency division duplexing
FEC	forward error-control

FFT	fast Fourier transform
FUSC	fully used sub-channelization
GWSSUS	Gaussian wide-sense stationary uncorrelated scatterers
H-FDD	half frequency division duplexing
ICI	inter-channel interference
IDFT	inverse discrete Fourier transform
IEEE	Institute of Electrical and Electronics Engineers
IFT	Inverse Fourier Transform
IFFT	inverse fast Fourier transform
IMT-2000	International Mobile Telecommunications-2000
ISI	inter-symbol interference
ITU-IB	ITU-R Indoor B channel
ITU-PB	ITU-R Pedestrian B channel
ITU-R	International Telecommunications Union - Radiocommunications
ITU-VA	ITU-R Vehicular A channel
LLR	log-likelihood ratio
LOS	line of sight
LS	least square
M	maximization
MAP	maximum a posteriori
MF	matched filter
MIMO	multiple-input multiple-output
ML	maximum likelihood
MMSE	minimum mean square error
MRC	maximum-ratio combining
MSE	mean square error
LMMSE	linear minimum mean square error
NLOS	non line of sight
NSC	non-systematic convolutional
OFDM	orthogonal frequency division multiplexing

OFDMA	orthogonal frequency division multiplexing access
OPUSC	optional partially used sub-channelization
PCK	perfect channel knowledge
pdf	probability density function
PIC	parallel interference cancelation
psd	power spectral density
PSK	phase-shift keying
PUSC	partially used sub-channelization
QAM	quadrature amplitude modulation
rms	root-mean-square
RNG	ranging
RSC	recursive systematic convolutional
Rx	receive antenna
SC	single carrier
SIHO	soft-in-hard-out
SISO	soft-in-soft-out
SNR	signal-to-noise ratio
SFBC	space-frequency block codes
SFBC-OFDM	space-frequency block coded OFDM
S-OFDMA	scalable orthogonal frequency division multiplexing
SS	subscriber station
STBC	space-time block codes
STBC-OFDM	space-time block coded OFDM
STTC	space-time trellis codes
TDD	time division dulexing
TG	task group
TGe	task group e
Tx	transmit antenna
UL	uplink
WiMAX	Worldwide Inter-operability for Microwave Access

WLAN	wireless local area network
WMAN	wireless metropolitan area network
ZF	zero-forcing

Chapter 1

Introduction

1.1 Motivations

The motivations of this thesis are to find possible solutions to two existing problems in the decoding process of orthogonal frequency division multiplexing (OFDM) [6] systems: inter-channel interference (ICI) in space-frequency block coded OFDM (SFBC-OFDM) over frequency selective fading channels and channel estimation in OFDM with particular application to Worldwide Inter-operability for Microwave Access (WiMAX) system. It is realized that iterative decoding, the real 'turbo principle', is able to provide superb performance without introducing too much complexity. Therefore in this thesis, it will be applied to OFDM systems to solve the above two problems.

1.1.1 Motivation I

Orthogonal space-time block codes (STBC) [7] is one of the multiple-input multiple-output (MIMO) [8] techniques that is able to realize the full spatial diversity of the sys-

tem. Based on the assumption of quasi-static flat fading channels, a simple linear block decoding algorithm can be achieved, which is able to give the maximum likelihood (ML) performance.

It is well-known that OFDM is able to transfer a frequency selective fading channel into a number of flat fading sub-channels. Therefore, it has been suggested to combine OFDM with orthogonal STBC in the form of SFBC-OFDM to allow STBC to perform well in multipath environment. The original simple linear decoding algorithm for STBC can be equivalently applied to SFBC-OFDM under the assumption that the channel frequency response remains approximately flat within a block. However, this assumption is not true in a severe multipath fading channel or OFDM with small number of sub-channels. In these two cases, frequency response variations between sub-channels will give rise to ICI during the conventional decoding process, resulting in a degradation in the performance.

Our objective is to cancel the above ICI caused by channel frequency response variations in SFBC-OFDM over quasi-static frequency selective fading channels. A zero-forcing (ZF) [9] algorithm has been proposed for $\mathbf{G}_2$ (Almamouti codes) [10]] SFBC-OFDM to eliminate ICI in such a system, albeit with some sacrifice in the diversity. However, the ZF algorithm is too complicated for $\mathbf{G}_4$ [11] SFBC-OFDM. Therefore instead, a parallel interference cancelation (PIC) scheme with soft iterations based on the conventional linear decoder is proposed to mitigate the effect of ICI in $\mathbf{G}_4$ SFBC-OFDM. When outer convolutional coding exists, the useful information from the decoder is further exploited and used to help improve the interference cancelation.

Fig. 1.1 illustrates the generic receiver structure of code-aided iterative PIC. The inner soft PIC iterations extract useful data information from interfered soft output of the linear SFBC decoder. The performance of interference cancelation is improved after several iterations, but the improvement is limited due to the non-optimal initial values from the linear decoder. However, the initial values in inner PIC iterations can be largely corrected by feeding back the information from the outer soft-in-soft-out (SISO) convolutional de-

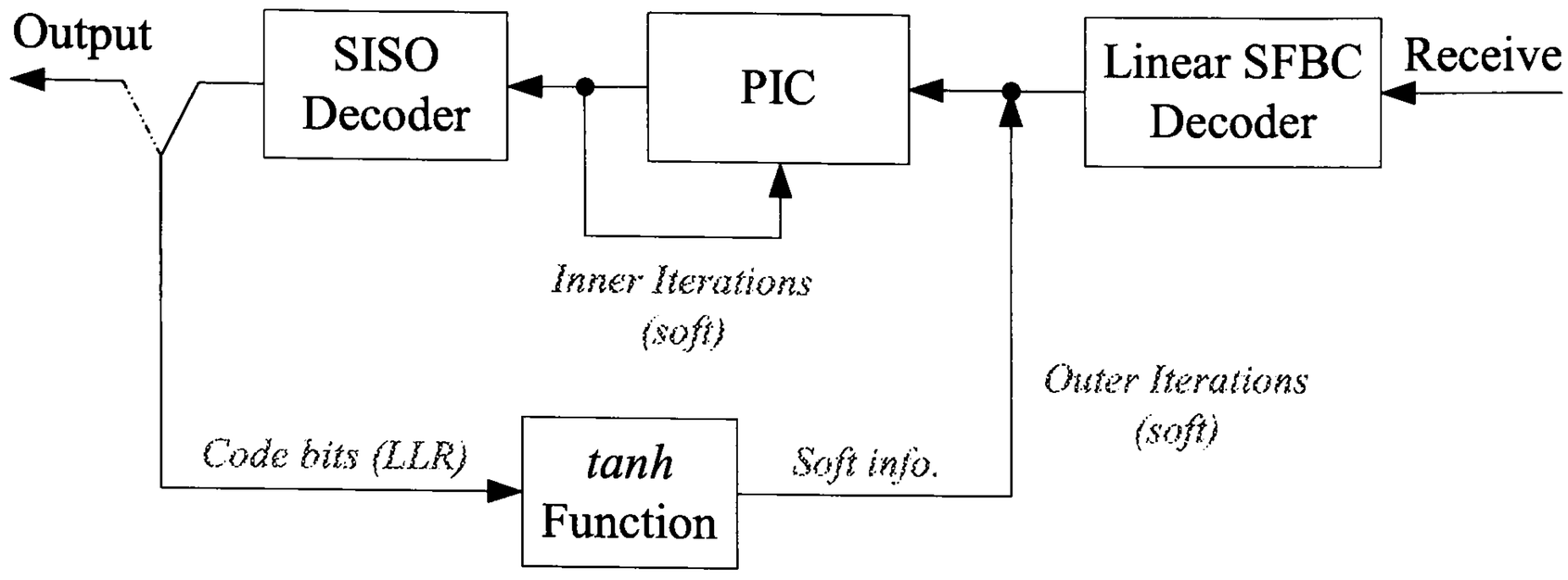


Figure 1.1: Generic receiver structure of code-aided iterative PIC in SFBC-OFDM

coder, which makes the inner PIC iterations to work more efficiently and provide better coded information for the outer SISO decoder. In other words, the two iterations help each other in an iterative manner and the performance of interference cancelation is expected to be greatly improved from one iteration to the next.

1.1.2 Motivation II

In wireless communications, channel estimation plays an important part in coherent receiver design because the performance of the system depends critically on the quality of channel estimation. Compared to the perfectly known channel, a degradation in the bit error rate (BER) performance will be introduced by the estimation. Although occupying some extra bandwidth and reducing the system efficiency, pilot-based channel estimation can offer a satisfied performance at a moderate level of complexity.

WiMAX technology has been rapidly developed and will play a key role in the fixed broadband wireless metropolitan area networks. In this thesis, the practical BER performance of channel estimation in physical layer of Institute of Electrical and Electronics Engineers (IEEE) 802.16 scalable OFDM Access (OFDMA) is investigated. The channel estimation schemes in uplink (UL) and downlink (DL) OFDMA are different depending

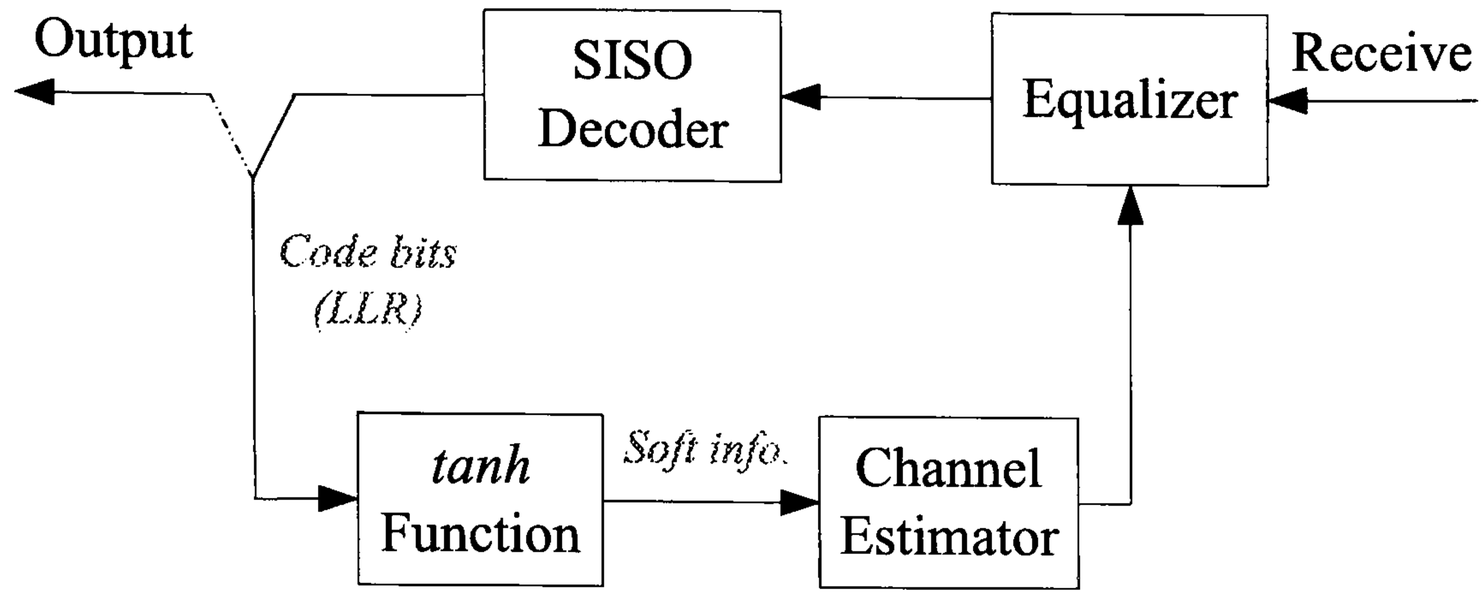


Figure 1.2: Generic receiver structure of code-aided iterative pilot-based channel estimation in OFDM/OFDMA

on the distribution of pilot symbols.

In UL OFDMA, pilots are scattered across subcarriers and symbols according to the standard. Conventionally, channel estimation is implemented based on these original pilots. However, there is still room for improvement, which can again be realized by utilizing the iterative technique. In this thesis, code-aided iterative channel estimations are employed, where the output of convolutional decoder is fed back to improve the channel estimation. Fig. 1.2 shows the generic receiver structure of the coded-aided iterative pilot-based channel estimation in OFDM/OFDMA. Without iteration, the initial estimation is purely based on the originally known pilot symbols. The received signals are equalized using these initial estimates before sent to the SISO convolutional decoder. According to the 'turbo principle', the outputs of the decoder, which contain both original pilot symbols and decoded soft data symbols (regarded as pilots), are sent back to help enhance channel estimation in the next iteration. Since more 'pilot' information are exploited without adding extra pilots, the performance is expected to be improved after several iterations. Certainly, the complexity from iterations is increased because all the symbols are used for estimation.

In DL OFDMA, a preamble symbol occupying all OFDM subcarriers is transmitted at the beginning of the frame, followed by regularly scattered pilots across subcarriers and symbols. Similarly to the iterative channel estimation, a code-aided iterative channel tracking

can be implemented based on the preamble symbol. The receiver has similar structure as Fig. 1.2. Moreover, the channel estimates from each iteration of tracking can be further improved by channel estimation using scattered pilot symbols. This process is iterated between the decoder and estimators until the most estimation accuracy is obtained.

1.2 Thesis Outline

The rest of thesis is organized into the following chapters:

- **Chapter 2: Channel Modeling and Channel coding**

  In this chapter, two fundamental technologies used throughout the thesis will be introduced. One is channel modeling, firstly two separate models for frequency selective fading channels and time-varying channels respectively, followed by a combined model for both time and frequency selective channels. The other one is channel coding, including convolutional coding and turbo coding, and their decoding techniques. The BER performance of these two codes over time-varying fading channels will be shown at the end of this chapter.

- **Chapter 3: OFDM**

  This chapter will deal with three basic issues in OFDM: subcarriers, guard time and equalization. It will then give the BER performance of OFDM over multipath fading channels and demonstrate that coding is a necessary part to realize the frequency diversity in OFDM systems.

- **Chapter 4: Orthogonal STBC**

  In this chapter, the encoding structure for G_2 and G_4 orthogonal STBC and their corresponding decoding algorithms over quasi-static flat fading channels will be introduced. The simulation results will show the BER performance of orthogonal

STBC with various numbers of transmit and receive antennas, and the corresponding spatial diversity obtained in such systems.

- **Chapter 5: Interference Cancelation in SFBC-OFDM over Quasi-static Frequency Selective Fading Channels**

  Firstly, this chapter will describe two combined schemes for OFDM and orthogonal STBC: STBC-OFDM and SFBC-OFDM. Their different behavior in time and frequency selective fading channels, using the channel model given in Chapter 2, and their sensitivity and tolerance to channel time/frequency response variations will be investigated. Then, the chapter will concentrate on the interference cancelation schemes in the decoding process of SFBC-OFDM over quasi-static frequency selective fading channels. For $\mathbf{G}_2$ SFBC-OFDM, a ZF decoding method will be introduced; for $\mathbf{G}_4$ SFBC-OFDM, firstly, a PIC approach based on the conventional linear decoding will be proposed and its performance between using hard and soft iterations will be compared. Then, the outer convolutional coding will be added and both inner PIC and outer Log-MAP iterations will be implemented. Simulation results will show the performance under various number of inner/outer iterations. Specifically for outer iterations, comparisons between the performance of a SISO and a soft-in-hard-out (SIHO) decoder will be presented.

- **Chapter 6: Performance Evaluation of IEEE 802.16 WMAN Physical Layer OFDMA**

  This chapter will provide an overview of the physical layer of IEEE 802.16 scalable OFDMA, and evaluate the practical performance of an OFDMA receiver in a typical mobile channel. Firstly, the performance deterioration due to channel estimations will be investigated. It will be demonstrated that the performance of conventional pilot/preamble based channel estimations can be improved by applying a code-aided iterative approach. Besides, the performance of mismatched channel estimation will be considered. Secondly, the chapter will measure the BER degradation due to timing errors and introduce a timing recovery scheme. Finally, the impact from the carrier frequency offsets will be measured and then the necessity

of implementing frequency synchronization in IEEE 802.16 OFDMA will be discussed.

- **Chapter 7: Conclusions and Future Work**

  This chapter concludes the whole thesis by summarizing the work, highlighting the contributions of this thesis and presenting some possible future work based on the thesis.

Chapter 2

Channel Modeling and Channel Coding

Contents

This chapter describes two fundamental techniques used in the thesis: multipath channel modeling and channel coding.

2.1 Multipath Channel Modeling

A typical characteristic of a wireless system is that transmissions between transmitter and receiver are not restricted to one single path. Due to the reflection and scattering from buildings, etc., transmitted signals are usually received with their replicas from different paths. Normally, signals on these paths are subject to different phase shifts, time delays, and Doppler shifts if there is relative motion between transmitter and receiver. Interfer-

ence between these signals gives rise to a series of harmful influences on the original signals. These effects are known as multipath, among which the most significant problems are fading and dispersion.

The main cause of fading is that multiple signals are received with random relative phases. These signals add either constructively or destructively resulting in random variations in the amplitude of the received signal. Dispersion in time is due to different time of arrival of various paths. If the delay spread is comparable with the symbol period, the delayed version of the symbol may interfere with the next transmitted symbol, giving rise to ISI. These channels are called frequency selective fading channels, since they exhibit attenuations in frequency response. On top of these, if there is motion between transmitter and receiver, multiple paths are subject to Doppler frequency shifts. This is because the phase shift of each path is changing with time, either slowly or rapidly. In fact, different paths have different Doppler shifts depending on their angles of arrival [12]. The collective effect at the receiver is a dispersion in the frequency band, called Doppler Spread. The corresponding channels are referred to as time-variant fading channels.

To set up a computer simulation of a channel, it is helpful to build mathematical channel models with consideration of multipath effects. This section will first of all introduce two existing and commonly used channel models describing time-dispersion and time-variant fading properties of the channel respectively, after which a single combined channel model is proposed and used in some parts of the thesis.

2.1.1 Tapped Delay-line Model

This section deals with the time dispersion characteristic of the channel. It will use the tapped delay-line model [13] to describe a frequency selective and time-invariant radio channel.

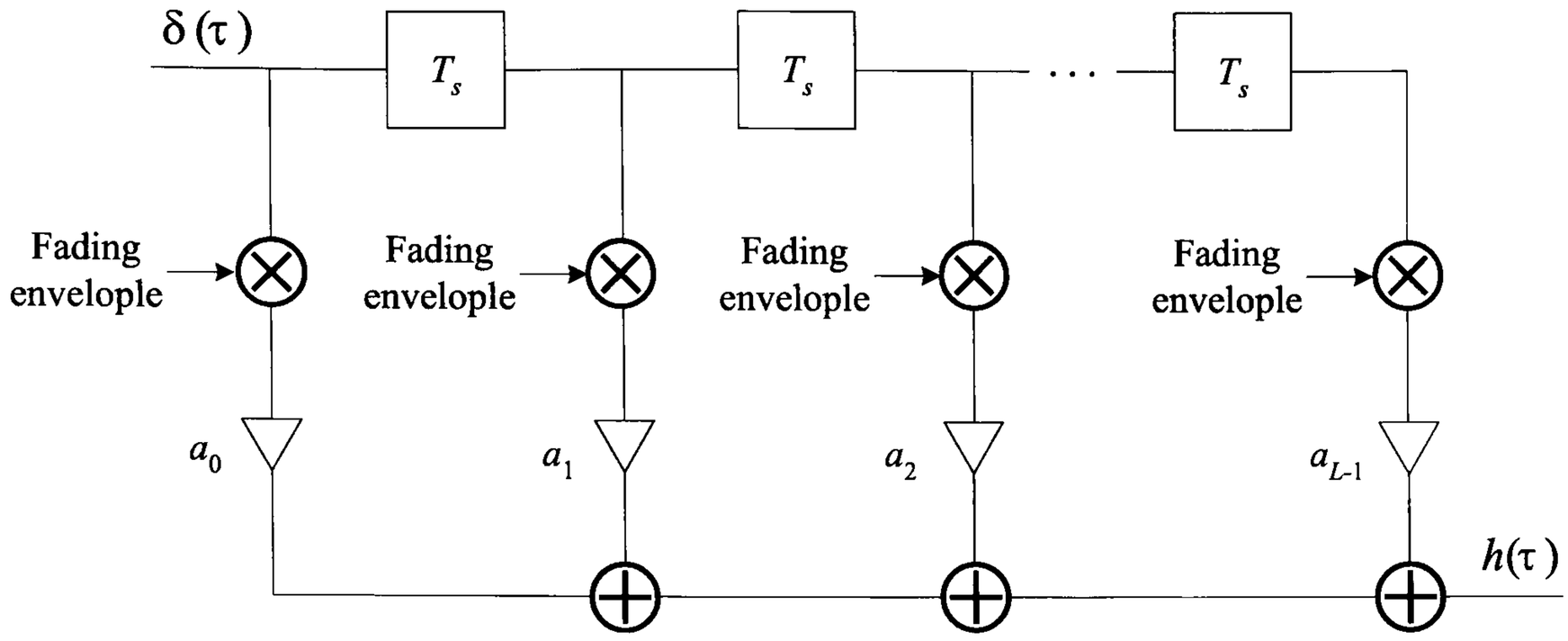


Figure 2.1: Tapped delay-line model

In general, the channel is modeled as a linear time-varying system [1], which can be described by its continuous time-varying impulse response, $h(\tau, t)$. This shows how the channel impulse response $h(\tau)$ varies with time t, where τ is the delay of the channel. If the time variation of the channel is relatively slow, e.g., the channel coherence time[1] is much longer than the symbol period or equivalently the Doppler bandwidth is much less than the signal bandwidth, the channel is regarded as quasi-stationary; in other words, channel varies with time, but are constant for periods of a few transmitted symbols. Furthermore, if the coherence time is much greater than the maximum delay, the channel is said to be separable, so that the delay parameter τ and the time t can be treated separately [1]. If the signals on different paths are uncorrelated and have Gaussian distributions, this leads to the familiar Gaussian wide-sense stationary uncorrelated scatterers (GWSSUS) model [13]. The channel impulse response is represented by a serious of impulses representing different paths with delays τ_i and complex amplitudes (incorporating also the phase) h_i, assuming invariant with time. The expression of this model is given in Eq. 2.1.

$$h(\tau) = \sum_{i=0} h_i \delta(\tau - \tau_i) \tag{2.1}$$

where i is the index of multipath components.

[1]The coherence time is for measuring the time over which the channel remains approximately constant.

For practical purposes, we are not interested in the response of the channel outside the bandwidth of the signal we intend to transmit over it [1]. Therefore, by lumping signals on different paths into delays of times of the symbol duration, the channel is sampled at the symbol rate to generate the tapped delay-line model (Fig. 2.1).

Each tap has a delay of multiples of the symbol period T_s and is weighted by a complex Gaussian coefficient a_j taking into account the shape of the signalling pulse at the receiver and the component amplitudes [14]. A fading envelope is multiplied to each tap, indicating the time-variant characteristics of the channel [1], which will be discussed in Section 2.1.2. This is shown in Eq. 2.2, where L is the channel order.

$$h(\tau) = \sum_{j=0}^{L-1} a_j \delta(\tau - jT_s) \tag{2.2}$$

Another simplified channel model is established when the symbol period is much greater than the channel delay spread, or equally the signal bandwidth is much less than the channel coherence bandwidth. In these cases multiple paths are reduced to one path, and therefore the tapped delay-line channel model is simplified to the first tap:

$$h(\tau) = a_0 \delta(\tau) \tag{2.3}$$

In this case, the channel is subject to frequency-flat fading, by contrast with the frequency selective one.

2.1.2 Jake's Model

Section 2.1.1 introduced a time-invariant frequency selective channel model: tapped delay-line model. In this section, the time-varying property the channel will be considered and the well-known Jake's Model [15] will be introduced for the simulation of time-variant Rayleigh fading channels. Among various Jake's simulators, the one proposed by Patzold [16] is chosen.

A Rayleigh process $\xi(t)$ is the absolute value of a zero mean complex Gaussian process $\mu(t)$:

$$\mu(t) = \mu_1(t) + j\mu_2(t) \tag{2.4}$$

$$\xi(t) = |\mu(t)| \tag{2.5}$$

where $\mu_i(t)$ $(i = 1, 2)$ are uncorrelated real components with variance of σ_0^2 for each dimension. The analytical cross-correlation function (ccf) between $\mu_1(t)$ and $\mu_2(t)$ is zero:

$$r_{\mu_1\mu_2}(t) = \mathrm{E}[\mu_1(t)\mu_2(t)] = 0 \tag{2.6}$$

A typical shape for the Doppler power spectral density (psd) of the complex Gaussian process $\mu(t)$ is given by the Jake's psd $S_{\mu\mu}(f)$ [15], the Inverse Fourier Transform (IFT) of which produces the corresponding autocorrelation function (acf) $r_{\mu\mu}(t)$ [16]:

$$S_{\mu\mu}(f) = \begin{cases} \frac{2\sigma_0^2}{\pi f_{max}\sqrt{1-(f/f_{max})^2}} & |f| \leq f_{max} \\ 0 & |f| > f_{max} \end{cases} \tag{2.7}$$

$$r_{\mu\mu}(t) = \mathrm{E}[\mu(t)\mu(t)] = 2\sigma_0^2 J_0(2\pi f_{max} t) \tag{2.8}$$

$$r_{\mu_i\mu_i}(t) = \mathrm{E}[\mu_i(t)\mu_i(t)] = \frac{1}{2} r_{\mu\mu}(t) \quad (i = 1, 2) \tag{2.9}$$

where $J_0(\centerdot)$ denotes the 0th-order Bessel function of the first kind; f_{max} is the maximum Doppler frequency that can be obtained by:

$$f_{max} = f_c \times \frac{v}{c} \tag{2.10}$$

where f_c is the carrier frequency; v and c are the relative velocity of the mobile object and the speed of light, respectively. For each dimension, Eq. 2.9 represents the analytical acf values of $\mu_1(t)$ or $\mu_2(t)$.

The principle of Patzold's simulation model for Rayleigh processes is to use a sum of sinusoids to generate a complex Gaussian random process $\mu(t)$ with Jake's psd $S_{\mu\mu}(f)$. The two real functions $\mu_1(t)$ and $\mu_2(t)$ are expressed as [16]:

$$\mu_i(t) = \sum_{n=1}^{N_i} c_{i,n} \cos(2\pi f_{i,n} t + \theta_{i,n}) \quad (i = 1, 2) \tag{2.11}$$

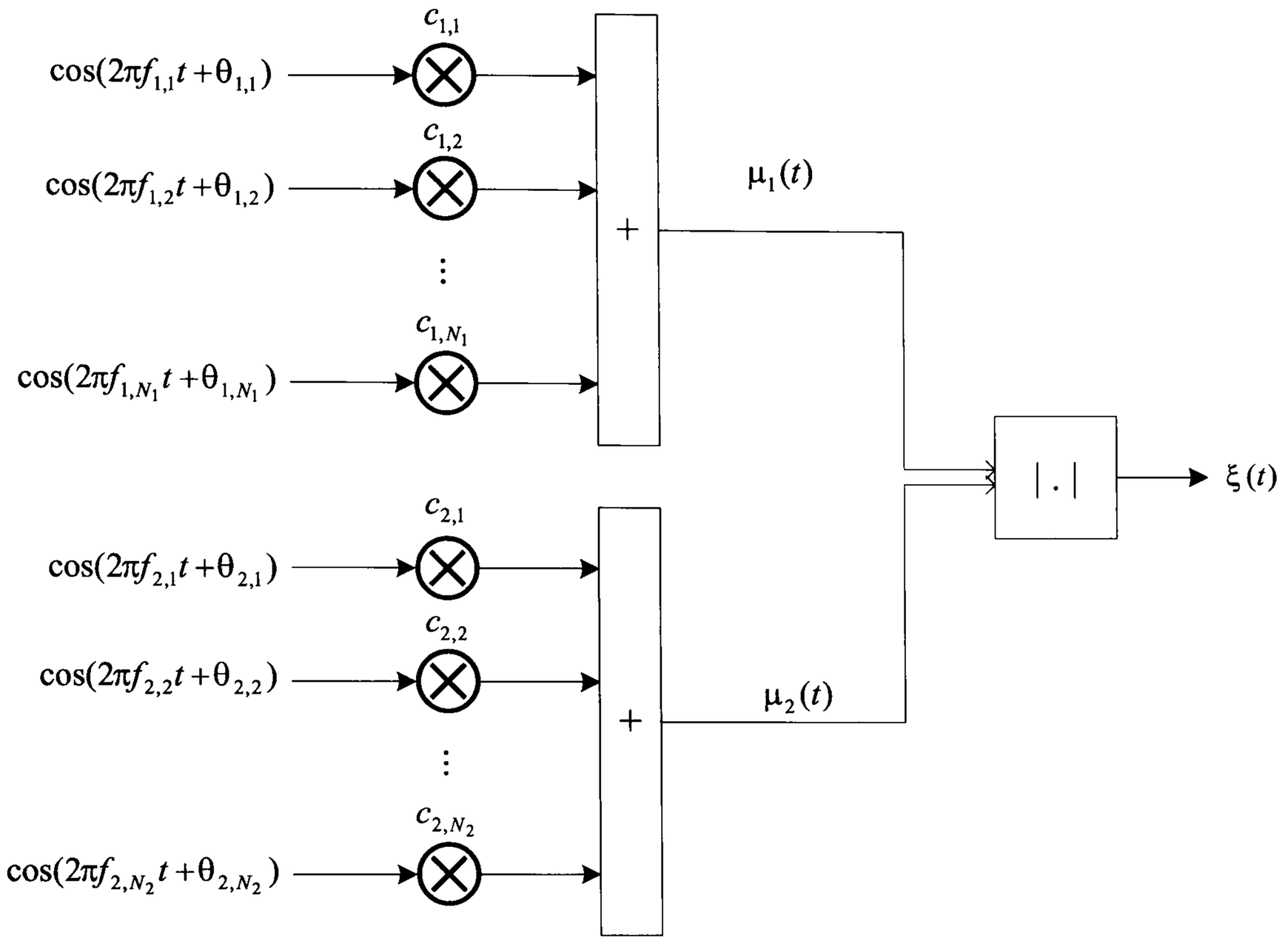


Figure 2.2: Patzold's simulation model for Rayleigh process with Jake's psd

where N_i is the number of sinusoids. The quantities $c_{i,n}$, $f_{i,n}$ and $\theta_{i,n}$ are simulation model parameters for Jake's Doppler psd function called Doppler coefficients, discrete Doppler frequencies and Doppler phases, respectively. Parameters $c_{i,n}$ and $f_{i,n}$ are deterministic. They are computed during the simulation setup phase by [16]:

$$c_{i,n} = \sigma_0\sqrt{2/N_i} \tag{2.12}$$

$$f_{i,n} = f_{max}\sin[\frac{\pi}{2N_i}(n-\frac{1}{2})] \quad (n=1,2...N_i) \tag{2.13}$$

The Doppler phases $\theta_{i,n}$ are random variables uniformly distributed over the interval $(0, 2\pi]$.

Fig. 2.2 shows the general structure of Patzold's simulation model for a Rayleigh process with Jake's psd in its continuous-time form [16]. From that, the discrete-time representation can be obtained by sampling at $t = kT_s$, where k is an integer and T_s is the symbol interval.

2.1.3 Time-variant Multipath Channel Model

By adding Jake's time-variant fading characteristics (fading envelope in Fig. 2.1) onto each tap of the tapped delay-line model, a time-variant multipath independent fading channel model is generated. It is assumed that each fading tap varies from symbol to symbol and remains uncorrelated with other taps.

According to Jake's model, independence between multiple fading taps is realized by randomly choosing Doppler phases $\theta_{i,n,l}$ (i=1,2; n=1,2...N_i; l=1,2...L), which are uniformly distributed over $(0, 2\pi]$, for each channel tap. As $\theta_{i,n,l}$ are mutually independent for different fading paths, the multiple channels are expected to be uncorrelated in fading.

The time-variant multipath channel model can be represented by a $(N + L - 1) \times N$ matrix $\tilde{\mathbf{h}}$:

$$\tilde{\mathbf{h}} = \begin{bmatrix} h_1^{(1)} & & & & & & & \\ h_2^{(2)} & h_1^{(2)} & & & & & 0 & \\ h_3^{(3)} & h_2^{(3)} & h_1^{(3)} & & & & & \\ \vdots & & & \ddots & & & & \\ h_L^{(L)} & \cdots & h_3^{(L)} & h_2^{(L)} & h_1^{(L)} & & & \\ & \ddots & & \cdots & & \ddots & & \\ & & & h_L^{(N)} & \cdots & h_3^{(N)} & h_2^{(N)} & h_1^{(N)} \\ & & & & h_L^{(N+1)} & \cdots & h_3^{(N+1)} & h_2^{(N+1)} \\ & & & & & h_L^{(N+2)} & \cdots & h_3^{(N+2)} \\ & & 0 & & & & \ddots & \vdots \\ & & & & & & & h_L^{(N+L-1)} \end{bmatrix} \tag{2.14}$$

where $h_l^{(k)}$ denotes the l-th (l=1,2...L) channel tap at a given time slot k (k=1,2...N); N is the number of symbols and L is the number of channel taps.

The transmitted symbols s_i $(i = 1, 2...N)$ are written into a matrix $\mathbf{s}$:

$$\mathbf{s} = \begin{bmatrix} s_1 & s_2 & \cdots & s_N \end{bmatrix}^{\mathrm{T}} \tag{2.15}$$

where $^{\mathrm{T}}$ denotes the vector transpose. Neglecting the effect of noise during the transmission, the received signal vector $\tilde{\mathbf{r}}$ can be obtained by

$$\tilde{\mathbf{r}} = \begin{bmatrix} \tilde{r}_1 & \tilde{r}_2 & \cdots & \tilde{r}_{N+L-1} \end{bmatrix}^{\mathrm{T}} = \tilde{\mathbf{h}}\mathbf{s} \tag{2.16}$$

Assume a particular case when the the channel remains static during the transmission period of N symbols:

$$h_l^{(1)} = h_l^{(2)} = \cdots = h_l^{(N)} = h_l \quad (l = 1, 2...L) \tag{2.17}$$

The matrix $\tilde{\mathbf{h}}$ can be simplified to:

$$\tilde{\mathbf{h}}' = \begin{bmatrix} h_1 & & & & & & & \\ h_2 & h_1 & & & & & 0 & \\ h_3 & h_2 & h_1 & & & & & \\ \vdots & & & \ddots & & & & \\ h_L & \cdots & h_3 & h_2 & h_1 & & & \\ & \ddots & & \cdots & & \ddots & & \\ & & & h_L & \cdots & h_3 & h_2 & h_1 \\ & & & & h_L & \cdots & h_3 & h_2 \\ & & & & & h_L & \cdots & h_3 \\ & & 0 & & & & \ddots & \vdots \\ & & & & & & & h_L \end{bmatrix} \tag{2.18}$$

Therefore, Eq. 2.16 is equivalent to taking the convolution of the time-invariant multipath channel $\mathbf{h}$ and the transmitted signal $\mathbf{s}$, where $\mathbf{h}$ is represented by

$$\mathbf{h} = \begin{bmatrix} h_1 & h_2 & \cdots & h_L \end{bmatrix} \tag{2.19}$$

2.2 Channel Coding

During radio transmissions, the original signals are likely to be corrupted by the channel and the noise. The signals are usually received with errors, which increases the unreliability of reconstructing the original information from the received data. The number of received bit errors divided by the total number of transmitted bits is defined as BER. BER performance is an effective way to evaluate the quality of a communication system. To reach a certain level of quality, various systems have different requirements on BER, e.g., 10^{-3} for a high-rate speech transmission and 10^{-6} for a low-rate data transmission. It is commonsense that, in most cases, the BER of a system decreases as the signal-to-noise ratio (SNR) increases. Therefore, a simple and intuitive way to reduce BER is to increase the SNR at the input to the demodulator. However, this benefit is gained at the sacrifice of transmit power.

Error-control coding (ECC) is a technique that has been widely used in digital communication systems. By adding redundant information to the transmitted data, it helps to correct the received errors and reconstruct the original data. The advantage of ECC is that the same BER may be achieved for a lower SNR in a coded system than in a comparable uncoded system [1]. This allows the power budget to be lowered and brings some other system advantages, e.g., coding gain.

Among all the powerful forward error-control (FEC) codes available, convolutional codes and turbo codes are deployed in this thesis.

2.2.1 Convolutional Coding

The convolutional encoder is composed of $(\nu - 1)$ serially concatenated shift register elements (implemented by linear delay operations) and several weighted modulo-M com-

biners. ν is known as the constraint length of the code and M is the code radix. In the case of binary codes[2], M=2. The weighting is simply a multiplication by 0 or 1, which is effectively realized by a disconnection (bit '0') or connection (bit '1') to the relevant shift register element. The outputs are additive combinations of the states of the connected shift registers.

A block of k_0 symbols is fed into the first shift register and moves along the rest at the speed of one block at a time. This indicates that the current output not only depends on the current block of k_0 symbols, but also on $(\nu - 1)$ previous data blocks of k_0 symbols. For every k_0 input symbols, the output contains n_0 coded symbols which are multiplexed to form a codeword. Note that the states of shift registers are set to be zeros after the useful data transmission by sending $(\nu - 1)k_0$ more zeros. This ensures that the initial states of registers are zeros before any data transmission begins. Supposing the length of the input data sequence is Nk_0, the rate of a convolutional code is obtained:

$$R = \frac{Nk_0}{Nk_0\frac{n_0}{k_0} + (\nu - 1)k_0\frac{n_0}{k_0}} = \frac{Nk_0}{Nn_0 + (\nu - 1)n_0} \tag{2.20}$$

Usually, $N \gg \nu$,. Therefore, an approximate code rate is given by

$$R \approx \frac{k_0}{n_0} \tag{2.21}$$

Fig. 2.3 gives an example of a convolutional encoder. D represents a shift register element, therefore the constraint length ν of the example convolutional encoder is 3. During each transmission, one data bit d is input ($k_0 = 1$). generating two coded bits c_1 and c_2 ($n_0 = 2$). This gives a code rate R of 1/2. According to the connection status between the two outputs and the shift registers, a generator matrix $\mathbf{G}$ is defined for this example of convolutional code:

$$\mathbf{G} = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 1 \end{bmatrix} \tag{2.22}$$

where each row represents an output and each column indicates the connection status.

[2]Only the binary case is considered in this thesis.

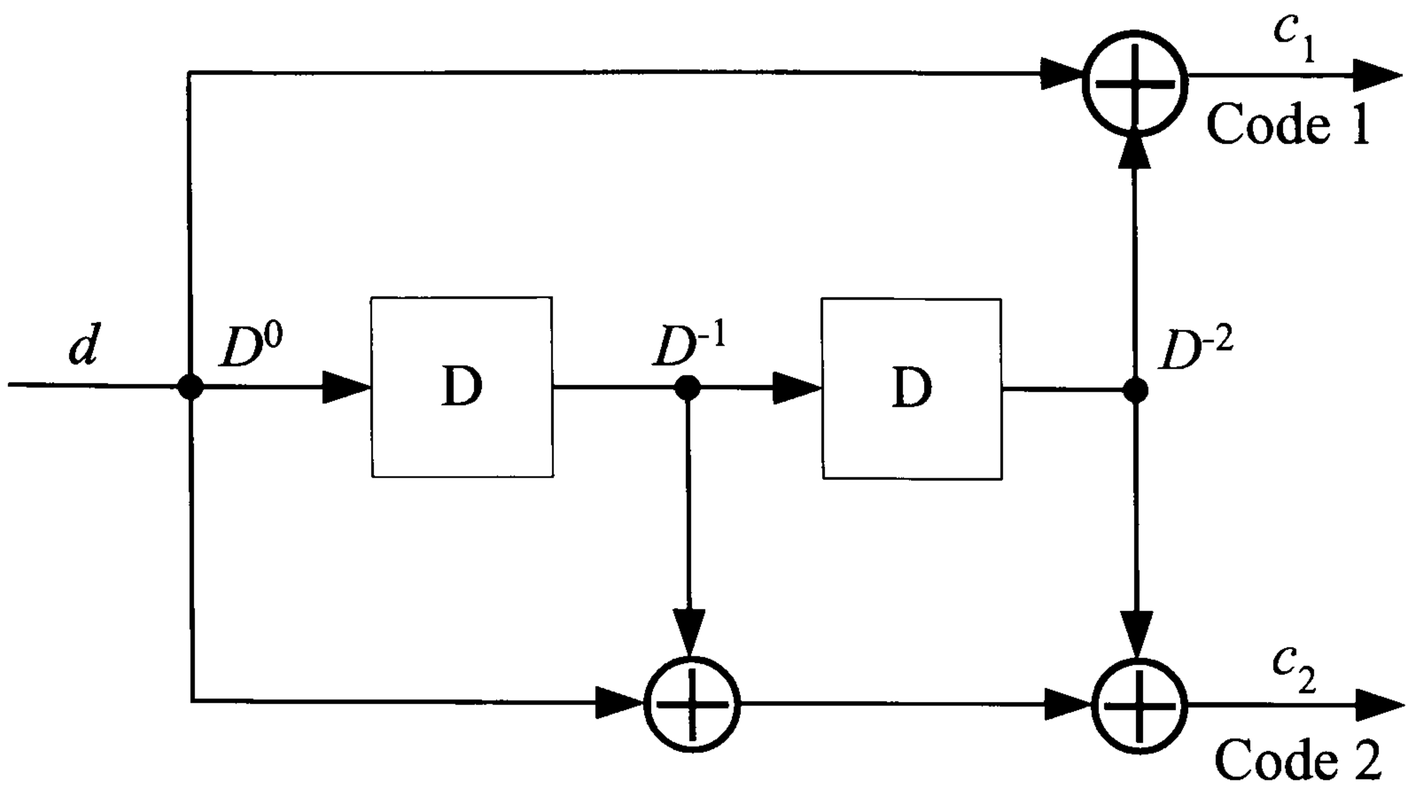


Figure 2.3: Example of convolutional encoder

Given the current states D of the shift registers:

$$\mathbf{D} = \begin{bmatrix} D^0 & D^{-1} & D^{-2} \end{bmatrix}^{\mathrm{T}} \tag{2.23}$$

the output codes C are:

$$\mathbf{C} = \begin{bmatrix} c_1 & c_2 \end{bmatrix}^{\mathrm{T}} = \mathbf{GD} \tag{2.24}$$

A convolutional code can be uniquely described by three parameters: the generator polynomial G (octal notation of the generation matrix), the code rate R and the constraint length ν. There are a number of useful tools for representing a convolutional encoder and its codes, e.g., the trellis diagram and the state diagram, the details of which can be found in [1].

Decoding is always the most difficult and computationally complex part of convolutional codes. The decoding algorithm used in this thesis is the famous Viterbi decoding [17]. Generally speaking, it is an algorithm for obtaining the decoded sequence by looking for a path, called the survivor path, through the code trellis diagram that most resembles the received code sequence. The corresponding input on the survivor path gives the decoded data sequence, while the outputs on the survivor path are parity codes form the code sequence.

There are two kinds of distances used for calculating the differences between the path and the received code sequence: Hamming distance and Euclidian distance, corresponding to the hard decision and soft decision, respectively. Soft decision decoding provides more coding gain than hard decision decoding in the equivalent case, and therefore is used in this thesis. The Viterbi decoder is usually referred to as a hard-output decoder since the decoded bits and parity bits are output in the form of zeros and ones. The parity bits from the Viterbi decoding can be used in the later iterations. The approach will be discussed in Chapter 5 and Chapter 6 of this thesis. For details of the Viterbi decoding, interested readers are referred to section 7.3.1 of [1].

As is known, convolutional coding can eliminate the deleterious effects of fading channels by providing a diversity gain potentially equal to the minimum Hamming distance of the code. In terms of the BER performance, the diversity gain behaves as an increase in the slope of the BER curve. However, the full diversity gain can only be obtained under the assumption that the bits of the code sequence fade independently. However in many cases, channels fade slowly and correlate within the coherence time. Hence the code sequence must be interleaved before transmitting. The aim of the interleaver is to separate the adjacent bits of the code sequence by at least the coherence time of the channel. Therefore after interleaving, the fading of code bits becomes independent and the full diversity gain can be obtained.

Fig. 2.4 and Fig. 2.5 show the BER performances of convolutional codes versus bit-energy-to-noise-density ratio (E_bN_0) over symbol-by-symbol time-variant flat fading channels. The noise is assumed to be additive white Gaussian noise (AWGN). The convolutional codes used in simulations have the same code rate of 1/2, but are different in strength: constraint lengths of 3,5,7 with generator polynomials (5,7), (23,35), (133,171) and minimum Hamming distances of 5,7,10, respectively. Both hard and soft decision decoding are simulated for comparison. The length of the code sequence is 256. A 16×16 block interleaver is chosen, which can separate the adjacent bits by a distance of 16 bits. Binary phase-shift keying (BPSK) modulation is used throughout the simulations.

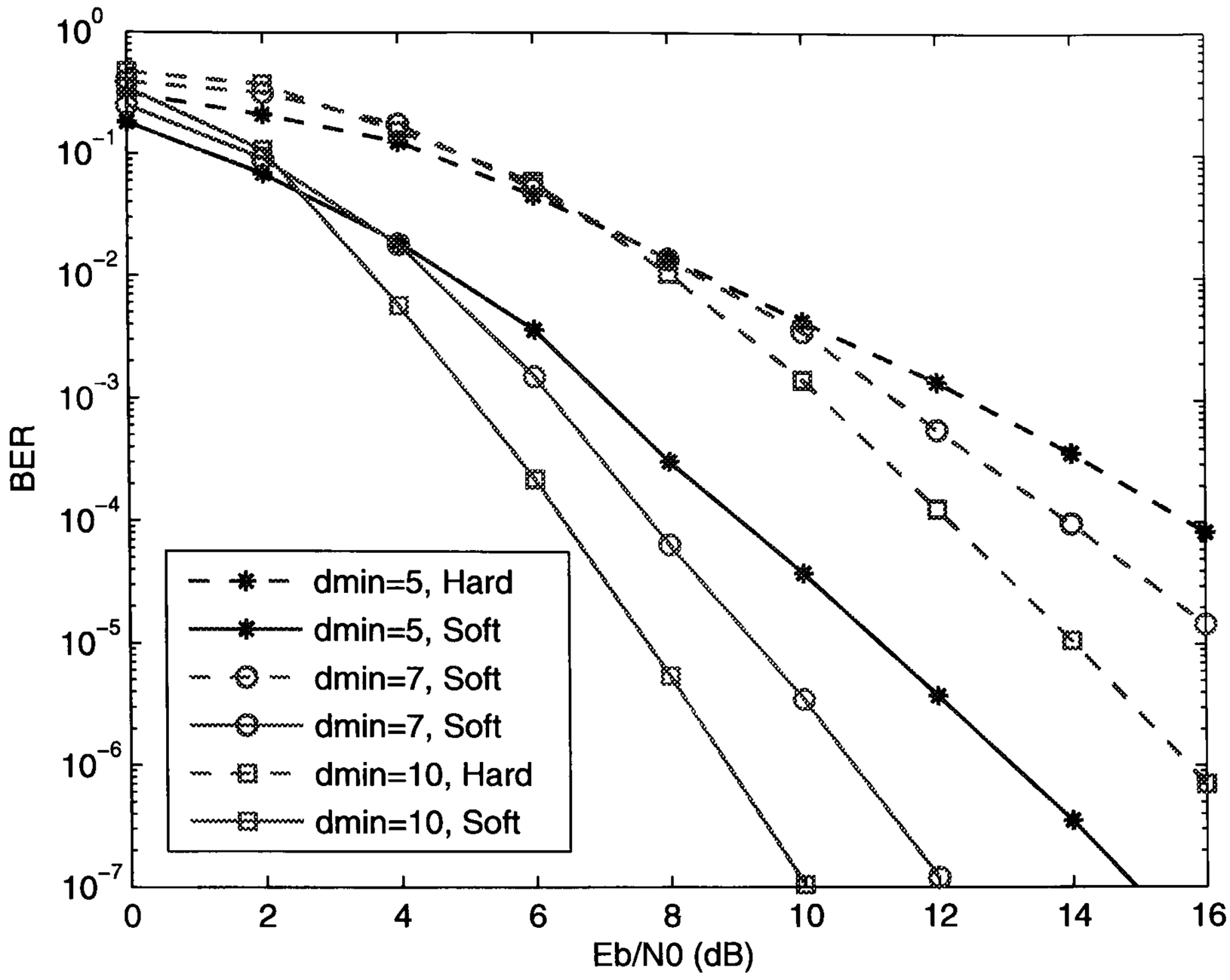


Figure 2.4: BER performance of convolutional codes over independent time-variant flat fading channels

Fig. 2.4 shows the performance of convolutional codes in independent[3] time-variant fading channels. The performance of different code strengths and decision methods are compared. It is obvious that the diversity gain increases as the code minimum distance increases. At high E_bN_0 regions, the diversity gain tends to reach the minimum Hamming distance of the corresponding code, at least for soft decision decoding. It is also observed that for the same code, soft decision can provide more gain over the hard decision, approximately 4dB at the BER value of 10^{-2}.

Fig. 2.5 gives the performance of convolutional codes in correlated[4] time-variant fading channels, with comparison to that obtained under independent channels. The codes have

[3]The channels at different symbol periods are randomly generated from a Gaussian distribution, therefore they are independent with each other.

[4]The channels at different symbol periods are sampled from a time-variant channel varying according to a fixed Doppler frequency, therefore they are correlated in time.

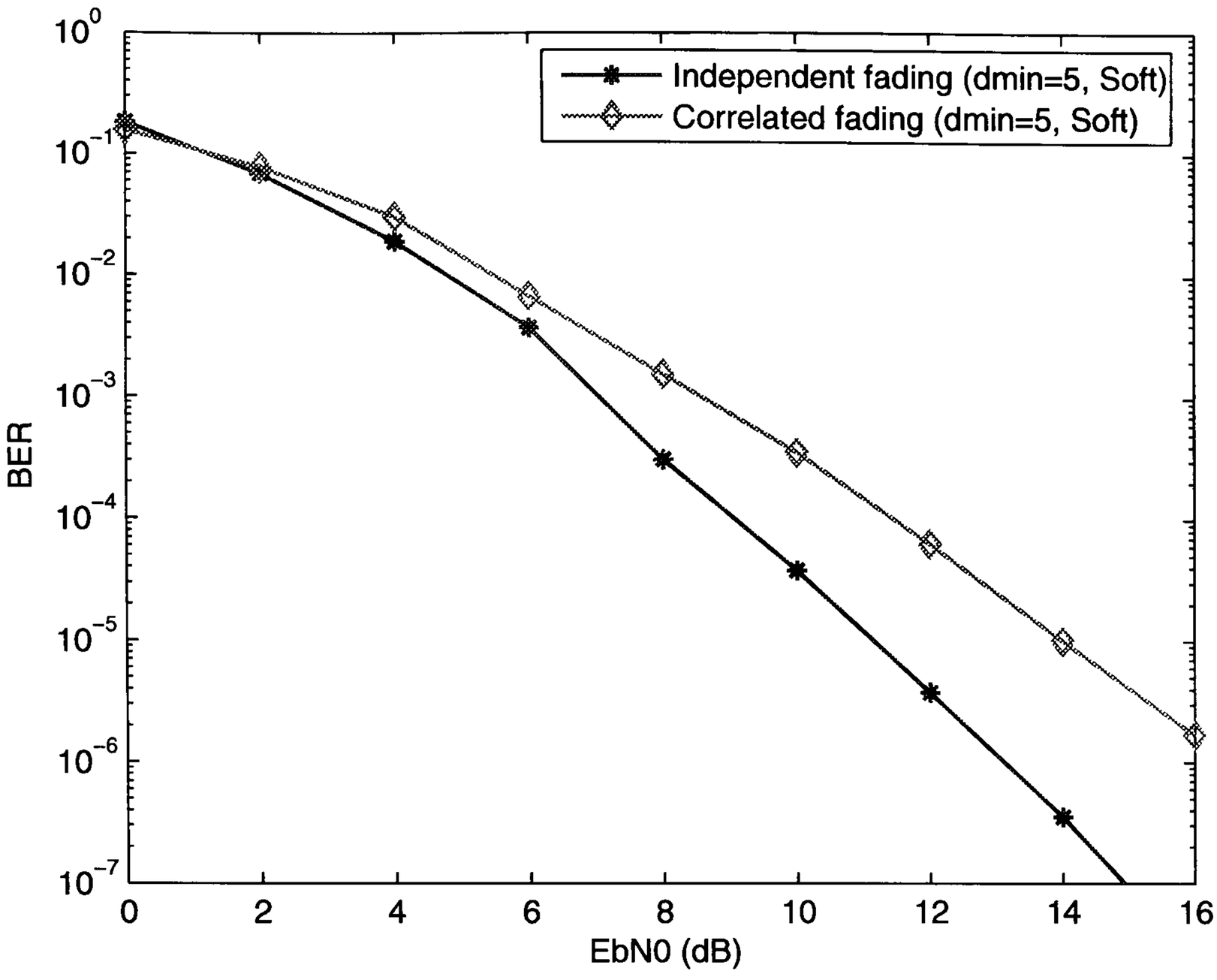


Figure 2.5: BER performance of convolutional codes over independent and correlated (fading rate 0.01) time-variant flat fading channels

the same strength (a minimum Hamming distance of 3) and decision method (soft). The fading rate of the correlated time-varying channel is 0.01, which gives a coherence time of about $1/0.01 = 100$ bit periods. Therefore, the interleaver would have to separate the adjacent bits by at least 100 bits away to create independent fading. This can hardly be achieved by a 16×16 block interleaver. Hence the available coding gain decreases in the correlated fading case.

2.2.2 Turbo Coding

As a new class of convolutional codes, turbo codes were first introduced by Berrou, Glavieux and Thitimajshima at International Conference on Communications in 1993 [18]. They claimed that a rate 1/2 turbo code can achieve a capacity of 0.7 dB away from

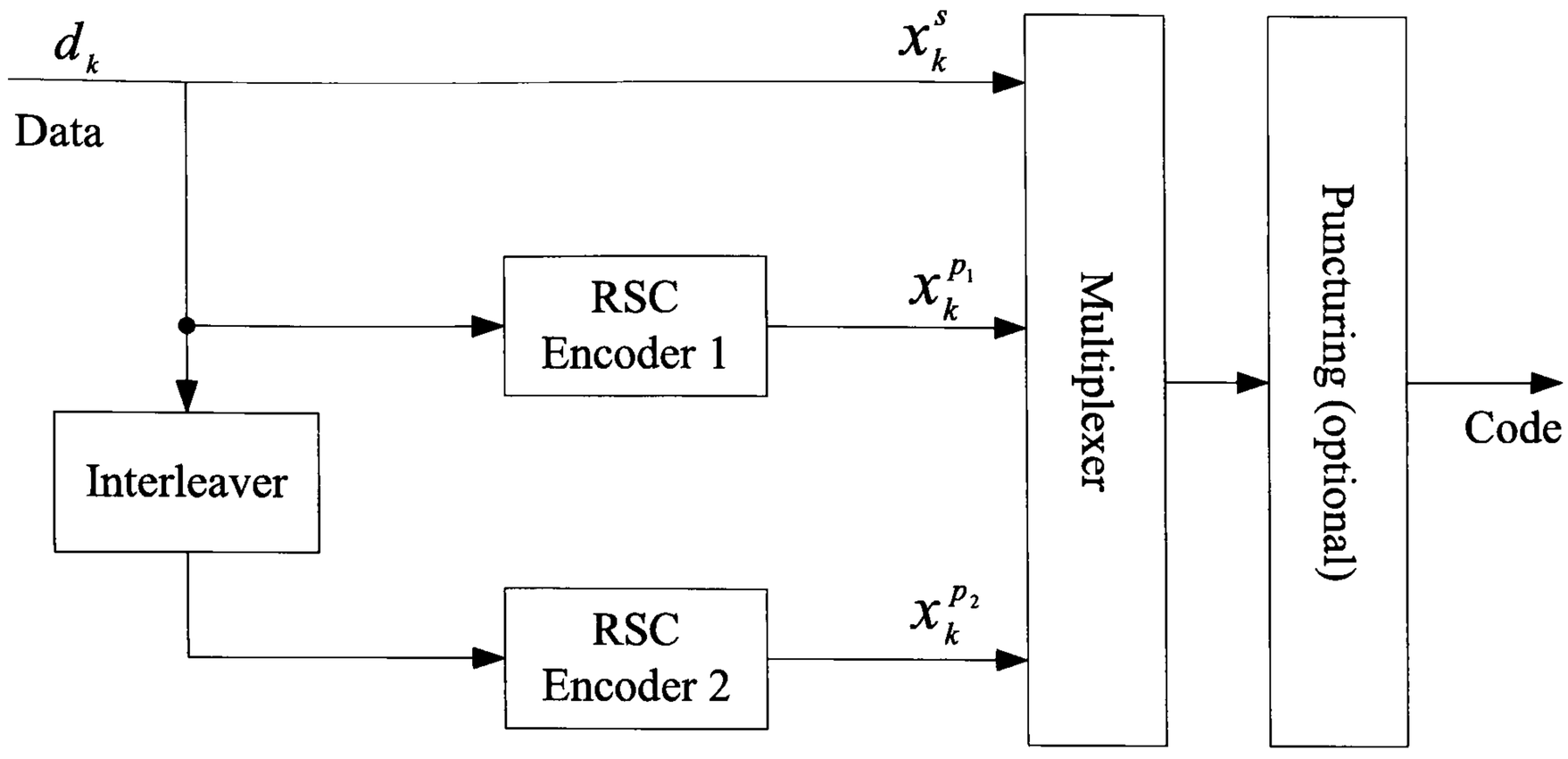


Figure 2.6: Example of the structure of turbo encoder

the Shannon capacity limit at the BER of 10^{-5} [18], which was a remarkable achievement in the existing coding world. The discovery of turbo codes puts an end to the thought that the Shannon limit can only be approached using extremely long codes with very complex decoding processes [19], but instead, by using a sub-optimal decoding method with a modest level of complexity. The powerful error-correcting capability of turbo codes have made them so popular that since the late 90s, turbo codes have been developed rapidly in practical applications [20] and also received intensive interest in the academic area.

Turbo codes are based on two fundamental concepts: concatenated coding and iterative decoding [21], the latter of which is the real 'turbo principle', since it is this method that provides remarkable performance of turbo codes. As turbo codes will be used in some parts of this thesis, this chapter will provide a brief introduction to the turbo encoding and decoding techniques in this section. For more extensive knowledge, readers are referred to [19] and [21].

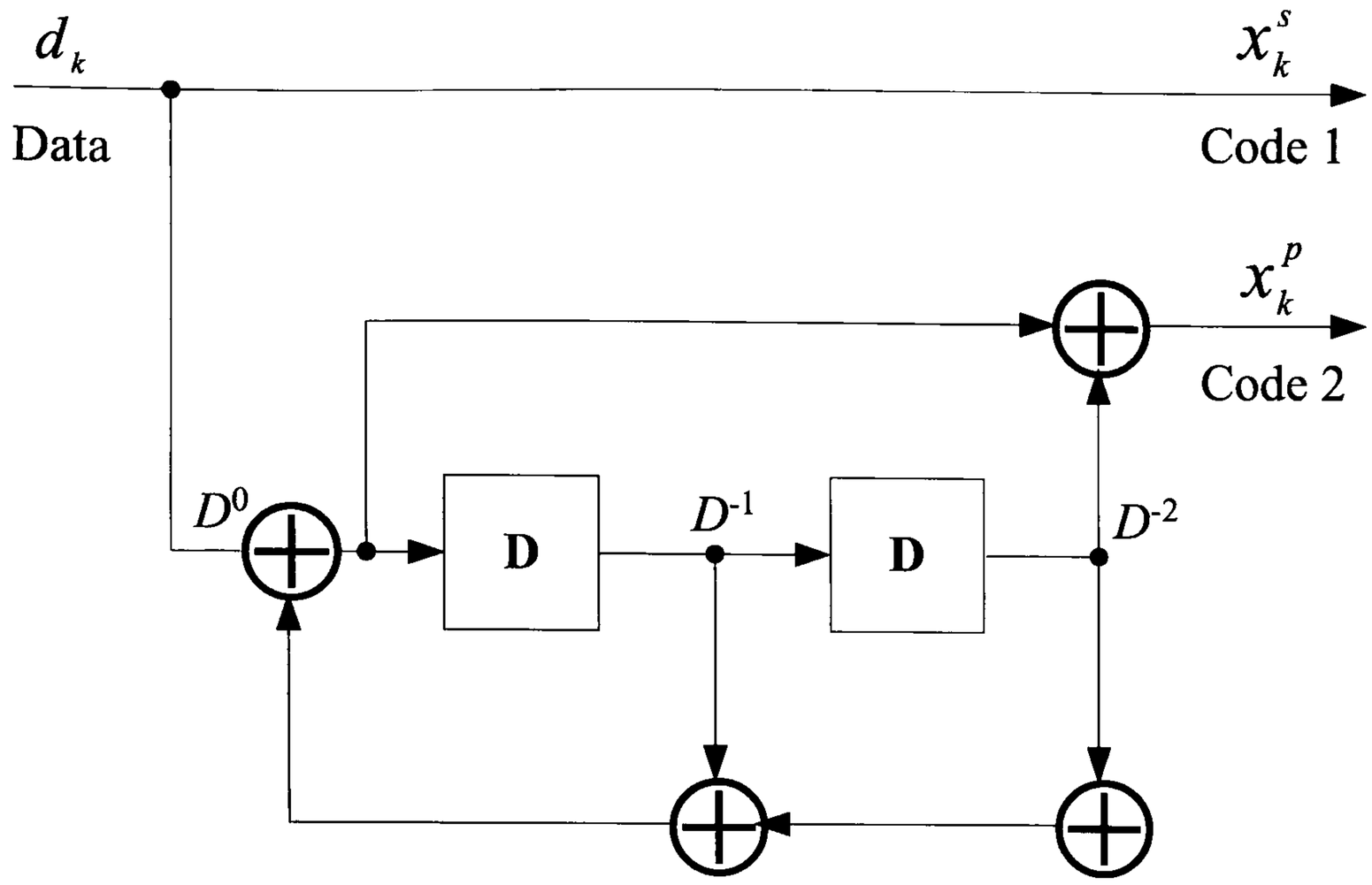


Figure 2.7: Example of a RSC code

2.2.2.1 Turbo Encoder

Fig. 2.6 gives an example of the structure of a turbo encoder. A turbo encoder is composed of two parallel concatenated recursive systematic convolutional (RSC) codes with an interleaver in between them. The turbo encoder works as follows. A length N data sequence $\mathbf{d} = [d_1, d_2, ..., d_k, ..., d_N]$ is encoded by the first RSC encoder, generating a code sequence (parity bits 1) $\mathbf{x^{p_1}} = [x_1^{p_1}, x_2^{p_1}, ..., x_k^{p_1}, ..., x_N^{p_1}]$. Then, the original data sequence $\mathbf{d}$ is interleaved and sent to the second RSC encoder, the output of which is another code sequence (parity bits 2) $\mathbf{x^{p_2}} = [x_1^{p_2}, x_2^{p_2}, ..., x_k^{p_2}, ..., x_N^{p_2}]$. Finally, $\mathbf{d}$ (also $\mathbf{x^s}$), $\mathbf{x^{p_1}}$ and $\mathbf{x^{p_2}}$ are multiplexed together and transmitted through the channel. Without puncturing, this results in a code rate of 1/3; if necessary, puncturing can be used to obtain a higher code rate.

The component convolutional codes are RSC codes. Compared with non-recursive non-systematic convolutional (NSC) codes, RSC codes use a feedback loop (thus 'recursive') and set one of the outputs equal to the input data (thus 'systematic'). Fig. 2.3 gives an

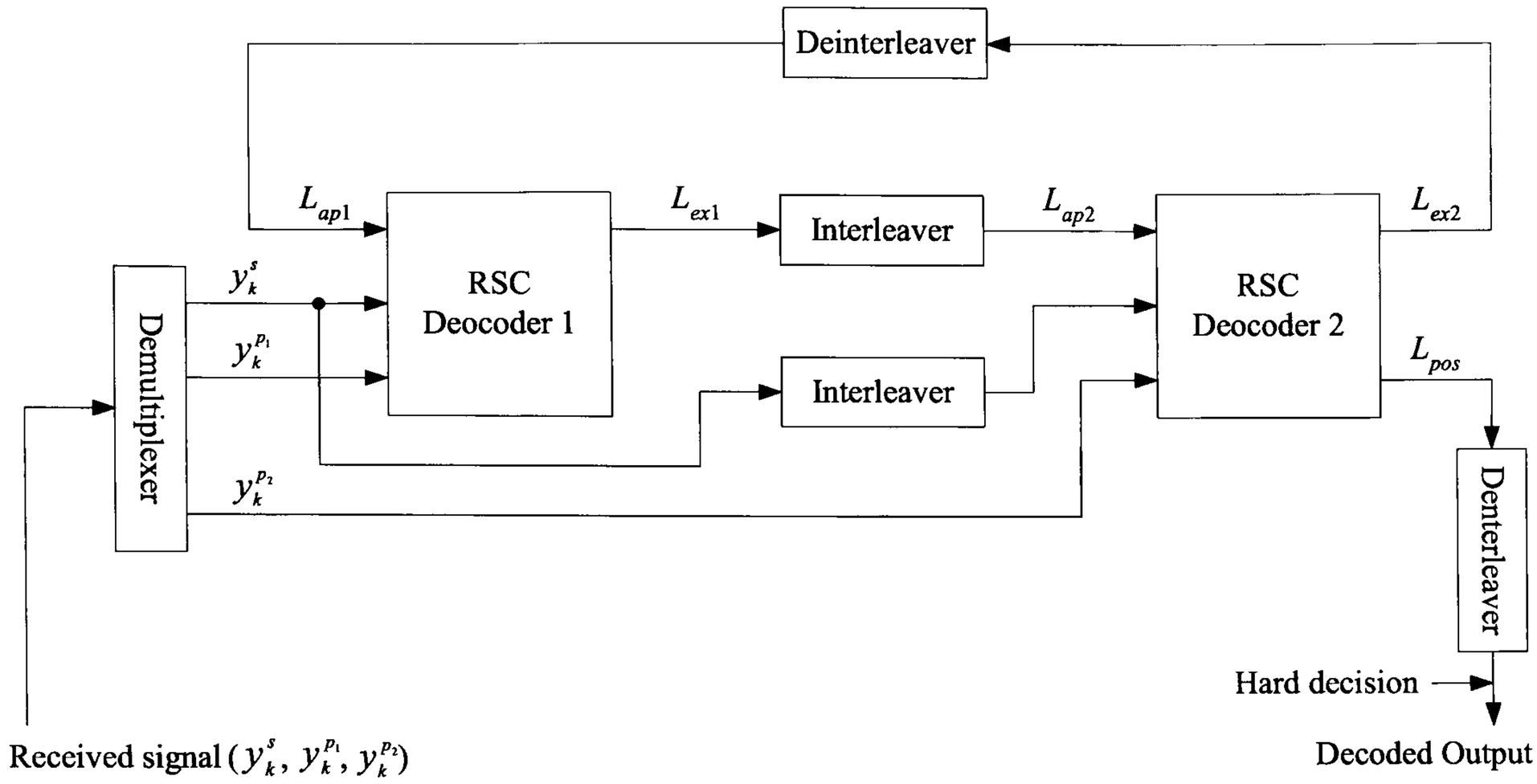


Figure 2.8: Structure of the turbo decoder for the example encoder

example of an NSC code, and Fig. 2.7 is its corresponding RSC code. The code rate and constraint length of the RSC code are 1/2 and 3, and the generator polynomials of the RSC code are (5,7), corresponding to its output and feedback connection respectively.

The interleaver between the two RSC encoders is a critical part in the turbo encoder leading to a good performance of turbo codes.The interleaver is to ensure the data bits that are close together when entering one RSC encoder are separated far apart before entering the other RSC encoder. In the turbo decoder, a corresponding deinterleaver takes the inverse operation to take the data sequence back to the original ordering. There are a number of interleavers that can be used in turbo codes, e.g., block interleavers, pseudo-random interleavers and s-random interleavers. Here, the optimum s-random [22] interleaver is chosen for its superior performance. The output of such an interleaver is generated randomly, with the condition that any two bits originally having a distance less than s bits are separated by at least s bits after interleaving.

2.2.2.2 Turbo Decoder

Fig. 2.8 illustrates the turbo decoder corresponding to the example encoder in Fig. 2.6. The decoder is composed of two serially concatenated RSC decoders that are linked by the same interleaver/deinterleaver used in the turbo encoder.

The turbo decoder works in an iterative manner and in each iteration, the two component decoders generate from the received sequence the estimates of the original uncoded data, and exchange the decoded information to help each other. Before iterations, the received signals $y_k = (y_k^s, y_k^{p1}, y_k^{p2})$ from the demodulator are multiplexed to the corresponding RSC decoder: y_k^s and y_k^{p1} to the first RSC decoder and y_k^s and y_k^{p2} to the second. When the parity bits of a given RSC encoder are punctured before transmission, the corresponding decoder inputs are set to zero at the punctured positions. In the first iteration, the first component decoder takes as its input the received systematic codes y_k^s and the received parity codes y_k^{p1}, and yields the estimate of the data bits L_{ex1}. Then the second component decoder uses its received systematic codes y_k^s and received parity codes y_k^{p2}, as well as the interleaved soft information of L_{ex1}, L_{ap2}, provided by the first component decoder, to obtain another estimate of the data bits L_{ex2}. In the subsequent iterations, besides the received signals, the first decoder also uses the additional de-interleaved information of L_{ex2}, L_{ap1}, from the second decoder in the previous iteration to generate L_{ex1}. The process is repeated iteratively until the two component decoders' estimates of the original data bits converge. Finally, another soft output L_{pos} delivered from the second decoder is de-interleaved and used for the final hard decision.

In general, each component decoder performs decoding by using its input received signals (y_k^s and $y_k^{p_i}$ $(i = 1, 2)$) and the *a priori* information (L_{api} $(i = 1, 2)$) from the other decoder, and produces the extrinsic information (L_{exi} $(i = 1, 2)$) for the other decoder. The final hard decision is made on the *a posteriori* information (L_{pos}), while the extrinsic information is only used for the exchange uses between decoders during the decoding process.

An important characteristic of the turbo component decoder is that it is a so called 'soft-in-soft-out (SISO)' decoder, which accepts soft *a priori* information L_{api} $(i = 1, 2)$ at its input from previous decoding (in addition to the input signal y_k^s and $y_k^{p_i}$ $(i = 1, 2)$), and generates soft information L_{exi} $(i = 1, 2)$ and L_{pos} at its output. Soft information means that in addition to the decoded bits, the reliability of decoding (the associated probabilities of decoded bits) is also provided, usually in the form of log-likelihood ratio (LLR). As its name implies, the LLR is the logarithm of the ratio of two probabilities in the case of binary transmission, e.g., the output *a posteriori* information (L_{pos}) is given by

$$L_{pos} = \log \frac{P(d_k = +1|\mathbf{y})}{P(d_k = -1|\mathbf{y})} \tag{2.25}$$

where the numerator and denominator are probabilities of the transmitted bit $d_k = +1$ and $d_k = -1$, provided the received sequence is $\mathbf{y}$. According to Eq. 2.25, the more positive the value of L_{pos} is, the more likely the transmitted bit was '1'; on the other hand, a more negative value of L_{pos} indicates a greater possibility that the transmitted bit was '0'.

The component SISO decoders are built based on the maximum *a posteriori* (MAP) algorithm, which was originally proposed by Bahl, Cocke, Jelinek and Raviv [23] and modified by Berrou *et al* [18]. Max-Log-MAP algorithm is a simplified version of the MAP algorithm. It drastically reduces the complexity of the MAP algorithm, but produces less accurate soft output. For computational simplicity and without losing of optimality, the Log-MAP algorithm, proposed by Robertson *et al* [24], is usually employed instead of the MAP and Max-Log-MAP algorithm. For more details about the MAP, Max-Log-MAP and Log-MAP algorithm, interested readers are referred to [1].

Fig. 2.9 is the BER performance of the turbo codes over the AWGN channel. The code employed in the simulation is a rate 1/3, 8 state turbo code. The generator polynomials of the component RSC encoders are (13,15) in octal for their feedback and output connections respectively. The length of the frame is 1024, giving a sufficient s-parameter of 20 for the s-random interleaver. The log-MAP algorithm is used for the decoders and the number of iteration trials are 1, 2, 3, 5, 8 and 10. Simulation results show that the BER performance of turbo codes improves as the number of iterations increases. The im-

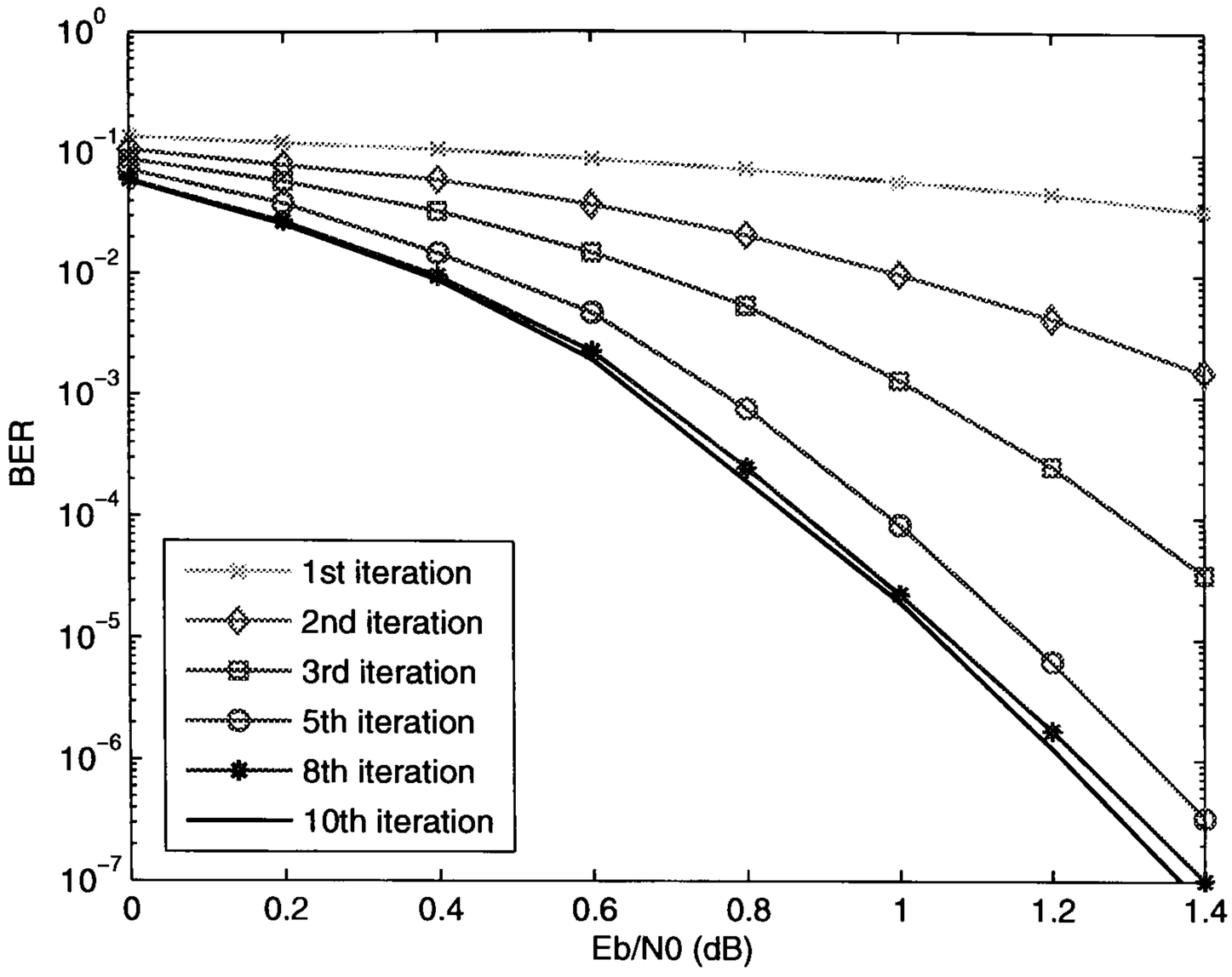


Figure 2.9: BER performance of turbo codes (rate 1/2, 8-state, 1024-bit Log-MAP) over AWGN channels

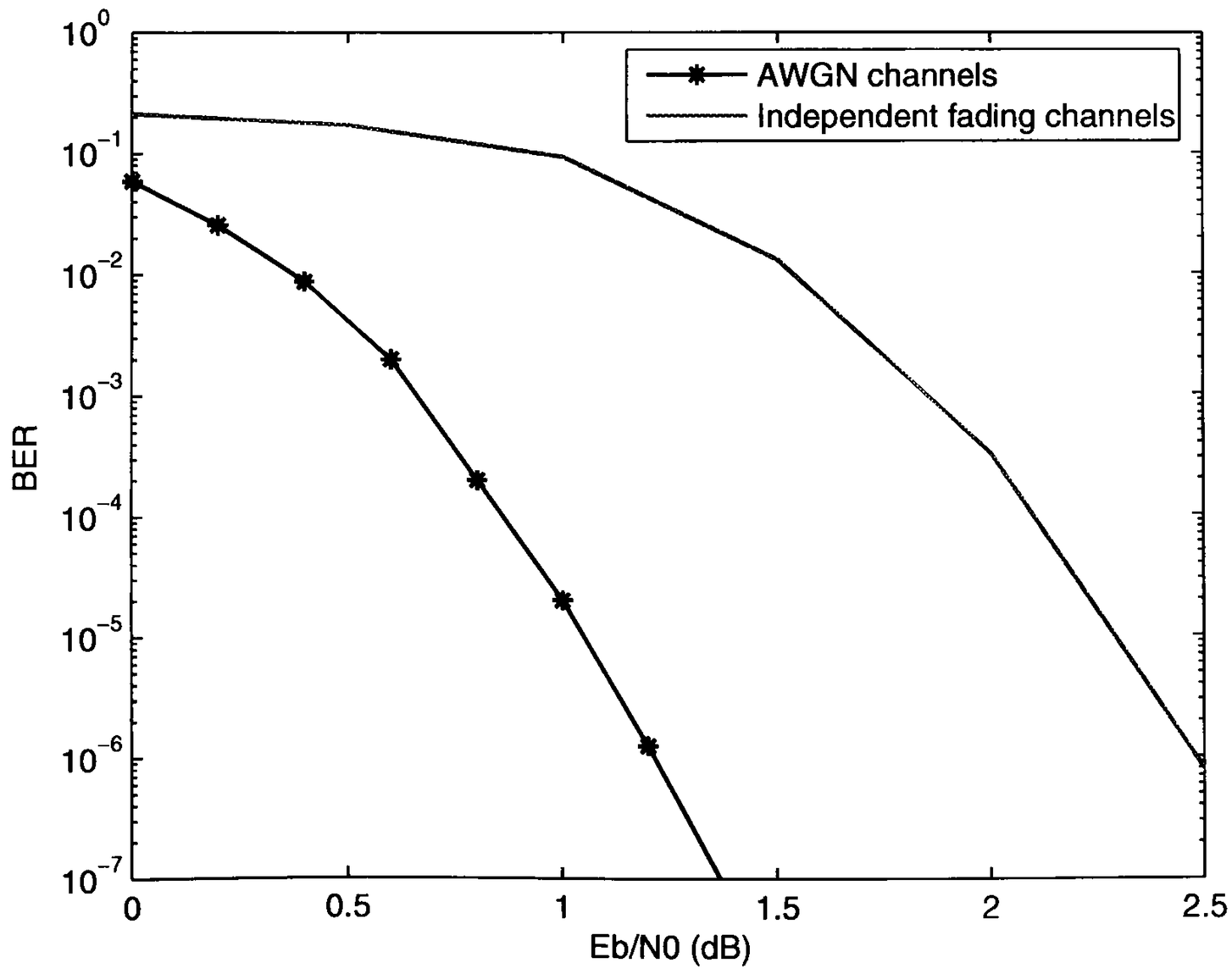


Figure 2.10: BER performance of turbo codes (rate 1/2, 8-state, 1024-bit, Log-MAP 10-iteration) over symbol-by-symbol independent time-variant fading channels

provement is great for small number of iterations, but becomes smaller as the number of iterations increases. Ten iterations is sufficient to obtain a good performance while maintaining a reasonable complexity. The BER performance of turbo codes over independent symbol-by-symbol time-variant fading channels are also given in Fig. 2.10. It is obvious that channel fading is a source of degradation in the performance of turbo codes. However, turbo codes are still able to provide a much better performance than the equivalent convolutional codes.

2.3 Summary

This chapter described the channel models and channel coding techniques that are used throughout the thesis. The tapped delay-line channel model will be applied in Chapter 3, Chapter 4 and Chapter 5 to investigate the BER performance of OFDM, orthogonal STBC and SFBC-OFDM, respectively, over quasi-static multipath (or flat) fading channels. The Jake's time-variant channel model has been used in this chapter to show the BER performance of convolutional codes over such channels. The frequency selective fading channel model will be utilized in the BER performance comparisons between space-time block coded OFDM (STBC-OFDM) and SFBC-OFDM in Chapter 5, and will appear later in simulating the performance of IEEE 802.16 OFDMA physical layer.

Convolutional codes and turbo codes are used to evaluate the BER performance of OFDM in coded systems, which will be discussed in Chapter 3. Besides, combined convolutional encoding and SISO Log-MAP decoding will be employed to provide improvement in the performance of relevant systems described in Chapter 5 and Chapter 6.

Chapter 3

OFDM

Contents

OFDM [6] is an old technique that can be dated back to as early as the 1950s. Despite that, it has attracted a lot of attention during the past a few years and been chosen for various systems such as digital audio broadcasting (DAB) [25], digital video broadcasting (DVB) [26], upstream on cable TV [27] and digital subscriber line (DSL) [28]. It has also been adopted in some of the Institute of Electrical and Electronics Engineers (IEEE) standards for wireless networks, e.g., 802.11a [29] and 802.11g [30] for wireless local area network (WLAN), and 802.16 [31] series for wireless metropolitan area network (WMAN). It was also proposed for the third-generation (3G) digital mobile radio, but

was not implemented in any of the main systems [1]. It is being regarded as a promising candidate for the fourth-generation (4G) mobile wireless systems [32].

This chapter will give a brief introduction to the basic structure of an OFDM system. The benefit of OFDM is that it eliminates the ISI introduced by the time dispersion of a multipath channel. By inserting a cyclic extension, ISI is further reduced. Equalization is required at the receiver, but the unique structure of OFDM makes it easier to implement than the conventional form. With the help of coding, the technique can also provide additional robustness due to diversity in the presence of narrow-band fading. The performances of uncoded and coded OFDM over multipath channels are investigated.

Timing and frequency synchronization in OFDM are also very important in the signal processing of an OFDM system, which will be discussed in more detail in Chapter 6.

3.1 OFDM Subcarriers

The maximum data rate on a multipath channel is limited for conventional modulation schemes due to ISI caused by the multipath dispersion [33]. ISI exists when the symbol period is less than the delay spread of the channel, when the channel is subject to frequency selective fading (Fig. 3.1). In this case, the received symbol contains delayed versions of previous symbols (Fig. 3.1). Without equalization, the highest data rate is given by the channel delay spread. On the channels frequently encountered in personal communication systems, this may limit the data rate to a few tens of k bits/s [34]. Time-domain equalization [35] is a way to eliminate the ISI so as to increase the data rate. However, the receiver complexity may increase rapidly with the product of data rate and delay spread [34].

Multi-carrier modulation is another effective way to solve the ISI problem. It was pro-

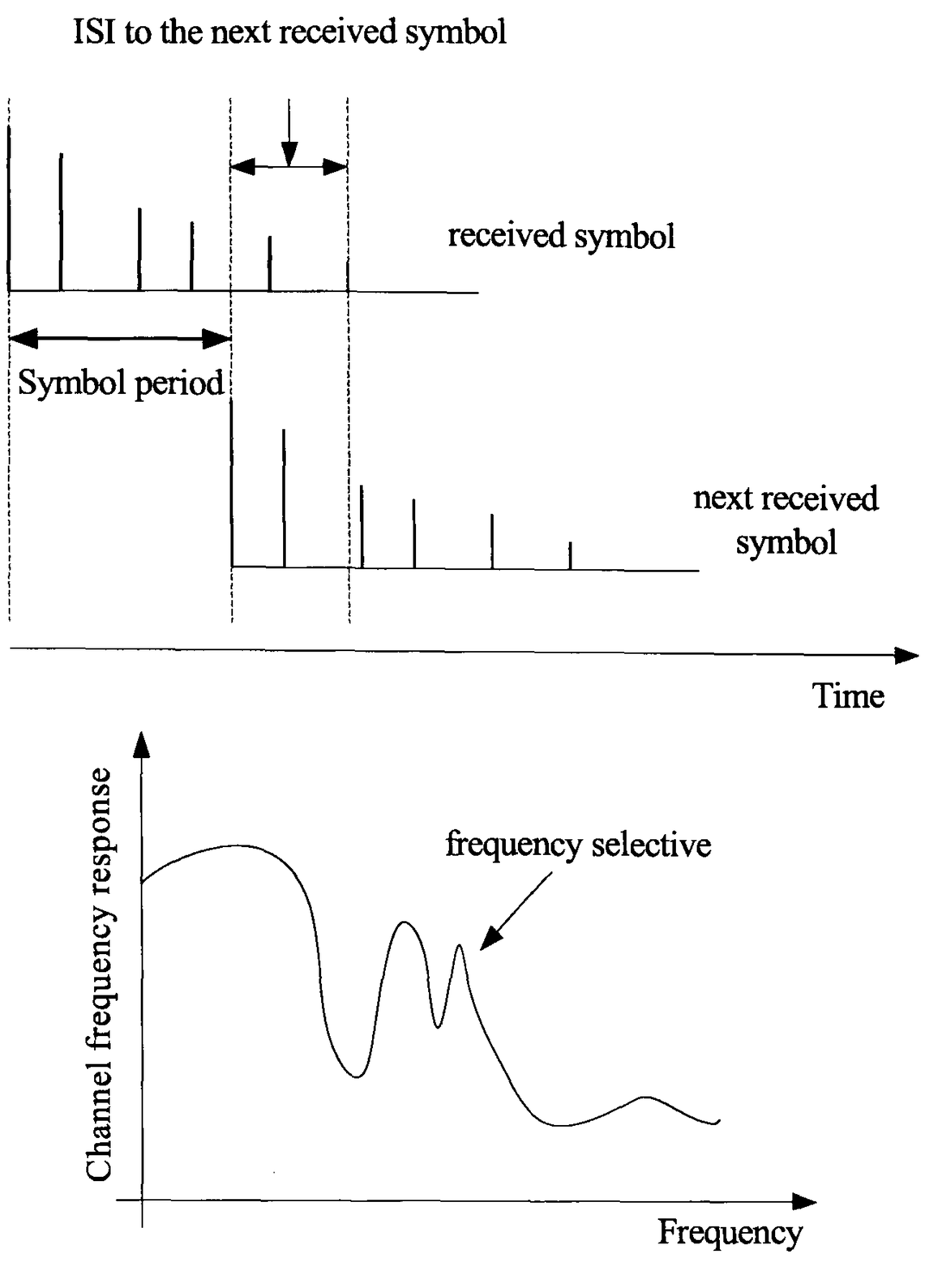


Figure 3.1: The generation of ISI on a multipath channel

posed at the early stage by Mosier and Clabaugh in 1958 [6]. In contrast to the conventional single-carrier modulation, multi-carrier modulation (Fig. 3.2 [1]) operates by dividing the available bandwidth into a number of spaced sub-channels, where the split data streams are modulated onto the corresponding carriers.

Although the total symbol rate equals that of a single-carrier system with the same bandwidth, the rate on each sub-channel is reduced relative to the delay spread of the channel, by N_s times (N_s is the number of sub-channels). Therefore, each channel is more tolerable to the time dispersion of the multipath channel. At the receiver, data on different sub-channels are separately demodulated by filters to remove side lobes. Between sub-channels, a guard band is required, so that sub-channels will not overlap significantly (Fig. 3.3(a)).

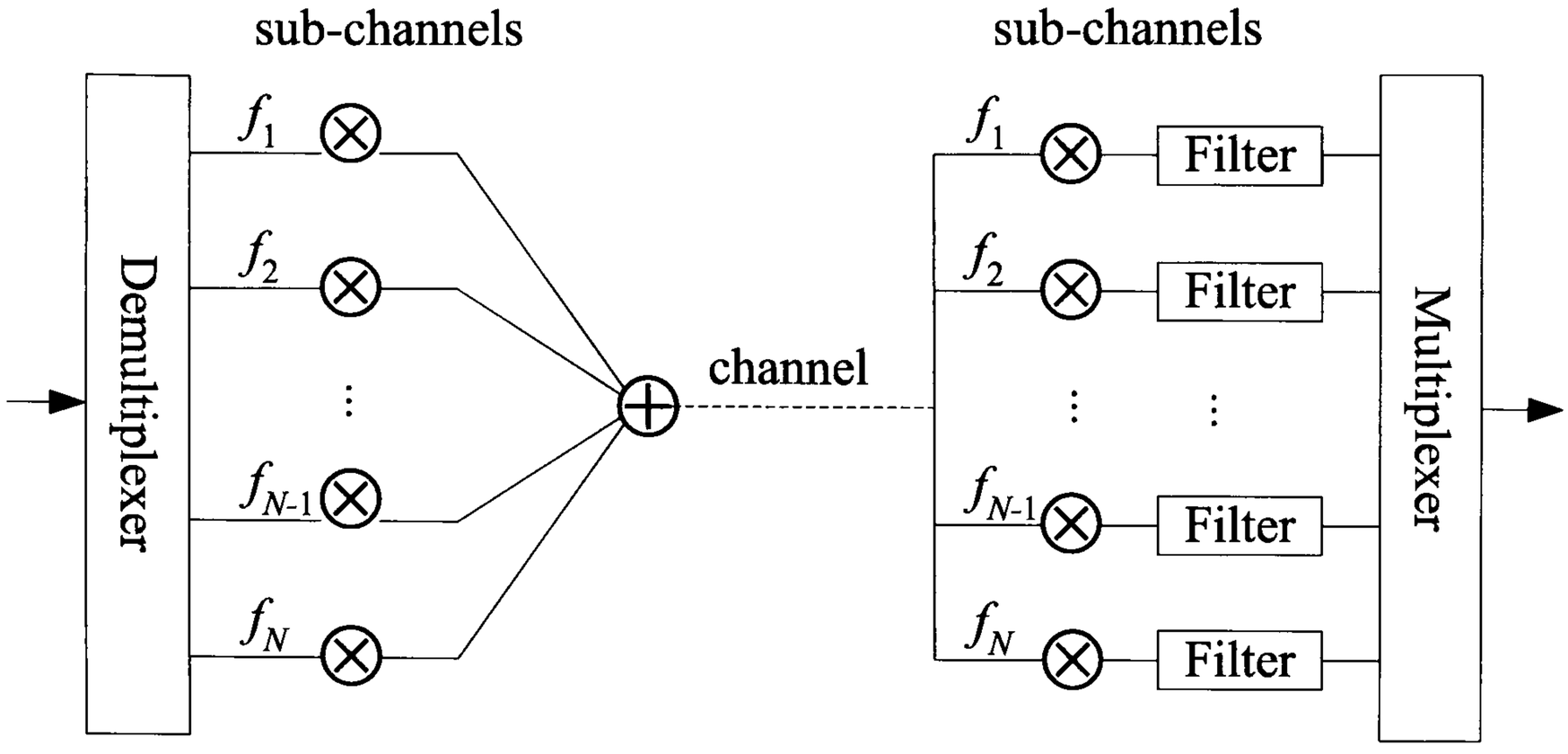


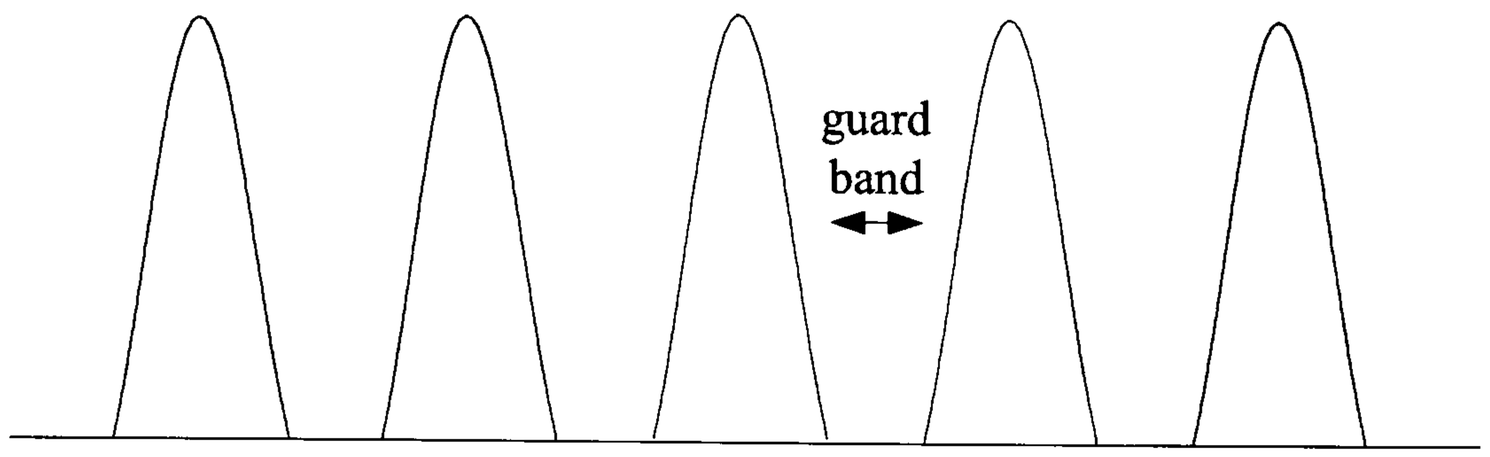
Figure 3.2: Structure of multi-carrier modulation [1]

The idea of OFDM appeared when Chang [36] and Saltzberg [37] realized that if the sub-channel spacing was equal to the OFDM symbol rate on each, the modulated signals would be orthogonal to each other, and could be simply separated by a matched filter (MF) or a correlator at the receiver [21]. In other words, each subcarrier has exactly an integer number of cycles in the OFDM symbol period, and the number of cycles of adjacent subcarriers differs exactly by one (Eq. 3.1).

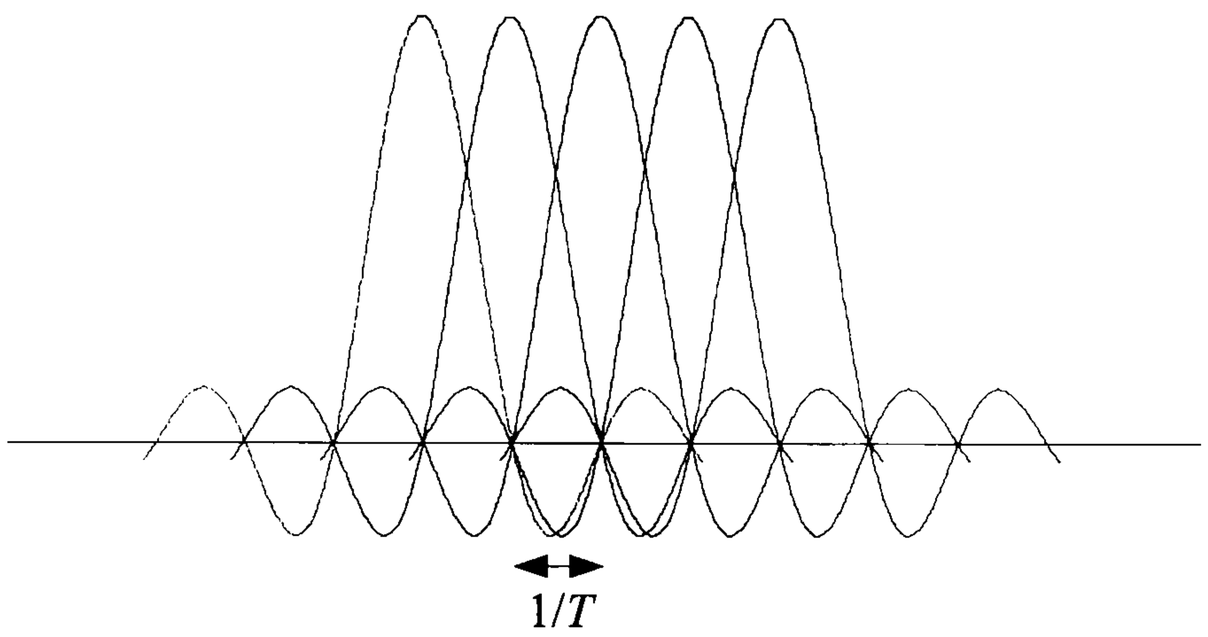
$$\Delta f = f_{i+1} - f_i = \frac{N}{T} - \frac{N+1}{T} = \frac{1}{T} \quad (i = 1, 2...N_s) \tag{3.1}$$

where f_i and f_{i+1} are subcarrier frequencies of the i-th and (i+1)-th sub-channel and Δf is the frequency spacing between them; T is the OFDM symbol period; N_s and N are the number of sub-channels and the number of cycles of the i-th carrier within the symbol period T, respectively. Fig. 3.4 shows an example of three subcarriers of one OFDM signal, supposing all subcarriers have the same phase and amplitude (may be different in practice according to the modulation scheme).

The orthogonality of OFDM subcarriers increases the bandwidth efficiency of the system significantly, since the originally spaced sub-channels in a conventional multi-carrier system can be overlapped without interference from each other by using OFDM. Fig. 3.3(b) illustrates the spectrum of individual OFDM subcarriers, with comparison to that in a conventional multi-carrier system.



(a) Conventional multi-carrier modulation



(b) OFDM

Figure 3.3: subcarrier spectrum of (a) multi-carrier modulation (b) OFDM

The final OFDM signal is the summation of the modulated signals on each subcarrier:

$$s(t) = \mathrm{Re}\{\sum_{i=-\frac{N_s}{2}}^{\frac{N_s}{2}-1} d_i \exp(j2\pi(f_c + \frac{i}{T}t)\} \quad t \in (0, T] \tag{3.2}$$

where d_i are the complex symbols modulated by phase-shift keying (PSK) or quadrature amplitude modulation (QAM) etc., and f_c is the carrier frequency. The equivalent complex baseband expression is written as

$$s(t) = \sum_{i=-\frac{N_s}{2}}^{\frac{N_s}{2}-1} d_i \exp(j2\pi\frac{i}{T}t) \quad t \in (0, T] \tag{3.3}$$

After sampling, the time-discrete baseband OFDM symbols are obtained:

$$s(k) = \sum_{i=-\frac{N_s}{2}}^{\frac{N_s}{2}-1} d_i \exp(j2\pi\frac{ik}{N_s}) \quad (k = 1, 2 \ldots N_s) \tag{3.4}$$

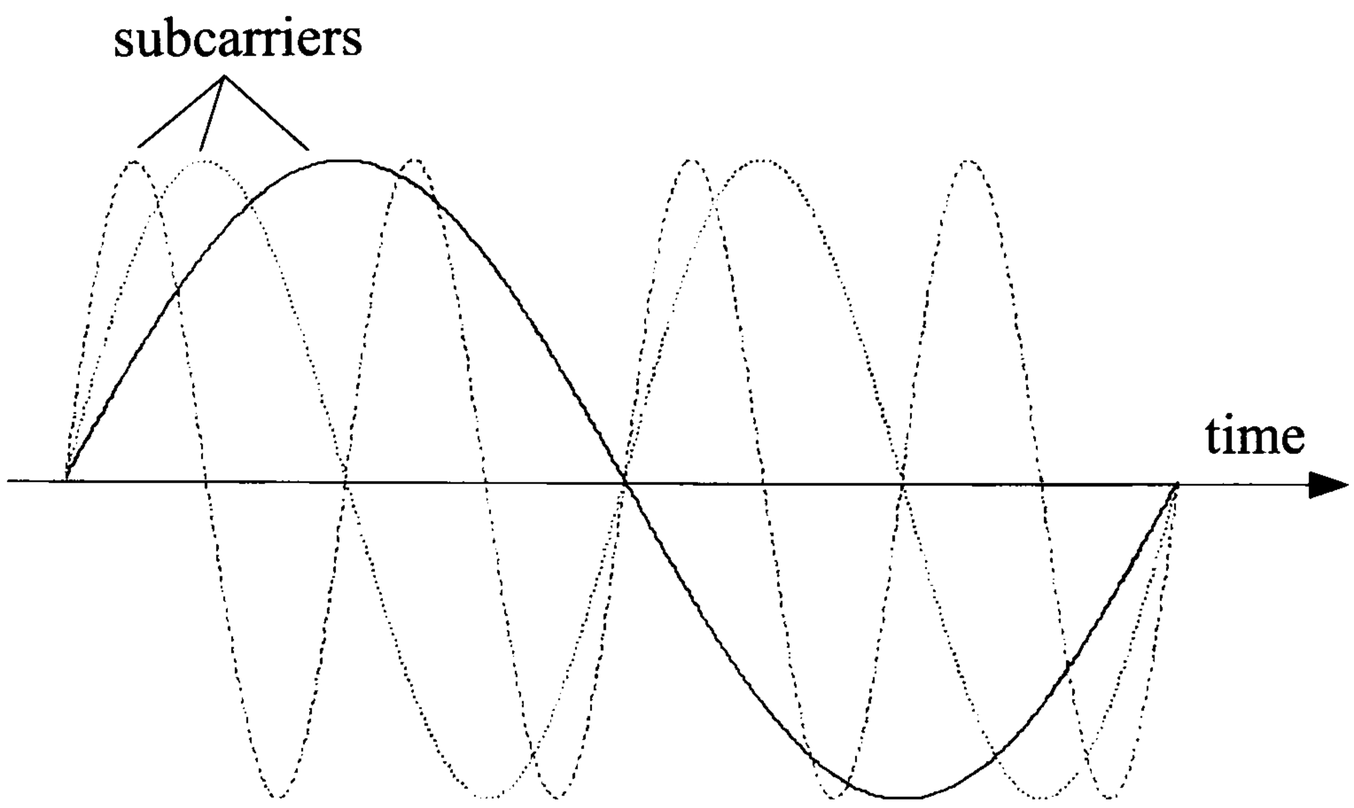


Figure 3.4: Example of three subcarriers in one OFDM symbol

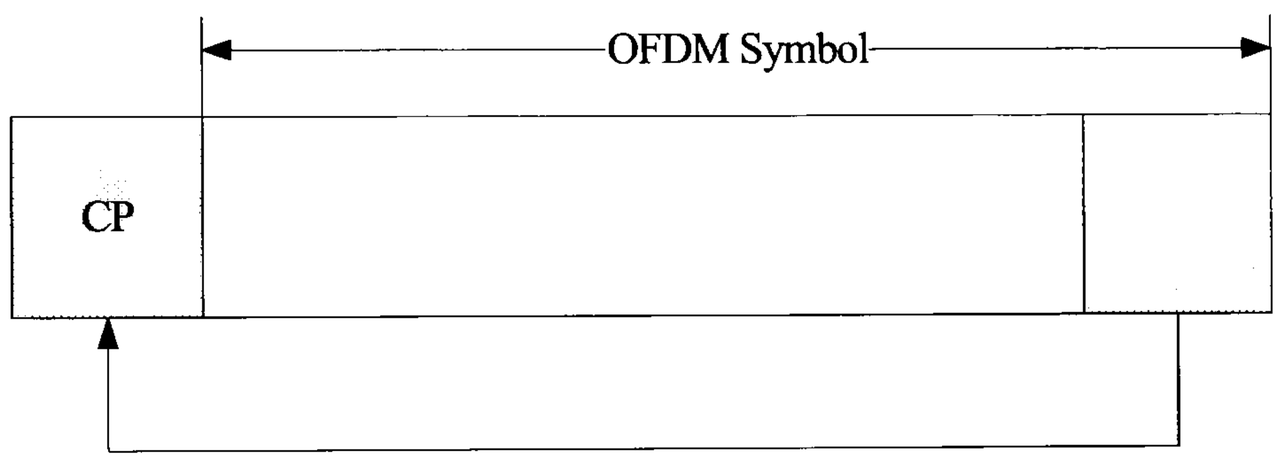


Figure 3.5: Inserting CP in an OFDM symbol

where k is the sampling index. Eq. 3.4 can be effectively realized by performing an inverse discrete Fourier transform (IDFT). Correspondingly, a discrete Fourier transform (DFT) is applied to demodulate at the receiver. Usually, the inverse fast Fourier transform and fast Fourier transform (IFFT/FFT) are used instead to reach a drastically reduced computational complexity. Thanks to the development in digital signal processing (DSP) devices, these transformations can be rapidly and readily implemented in practice [38]. In fact, this is the main reason that triggers the renewed popularity of OFDM.

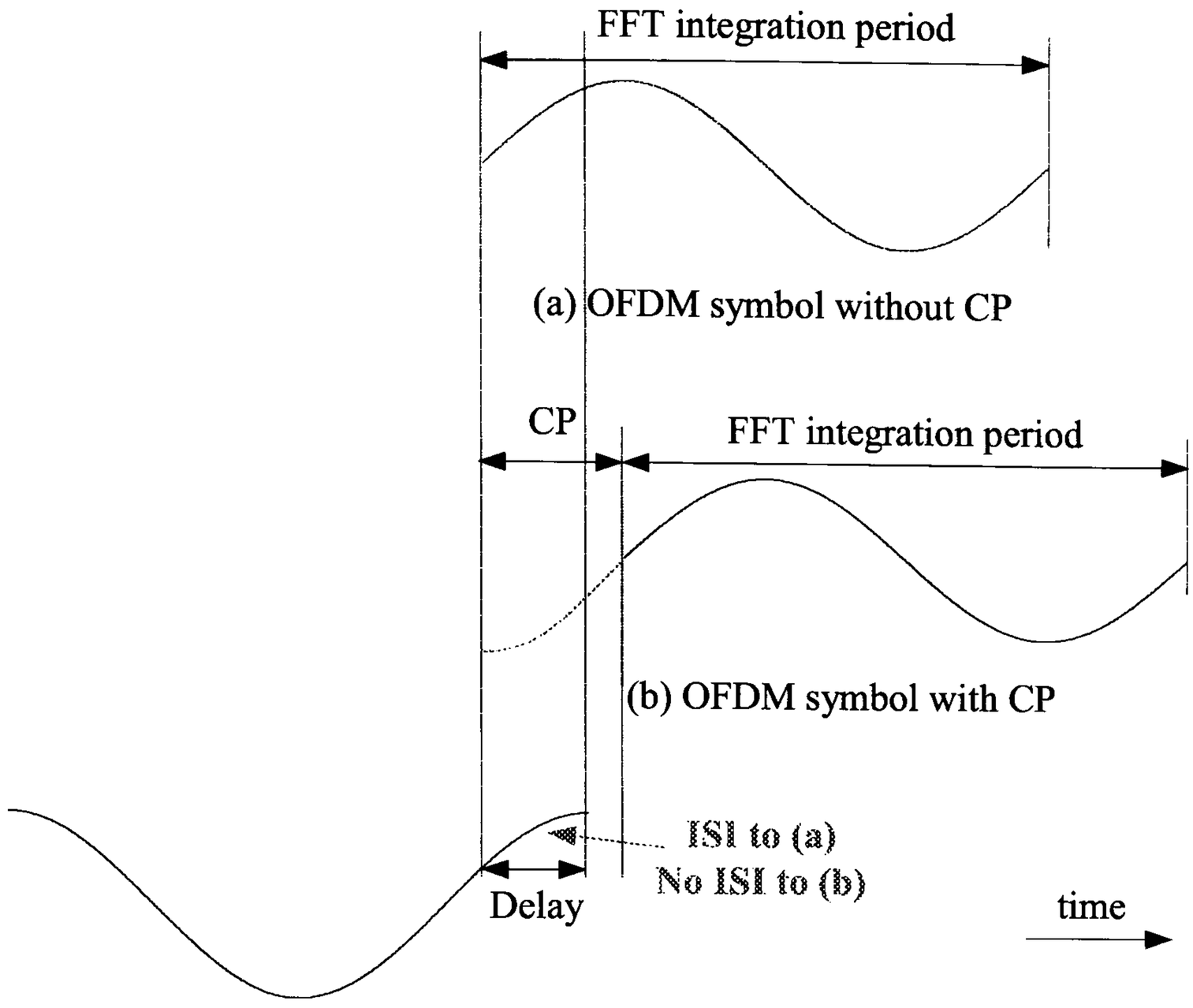


Figure 3.6: Elimination of ISI by adding CP to the OFDM symbol

3.2 Insertion of CP

The most attractive property of OFDM is its high resistance to the multipath delay spread. However, with the scheme so far described there is still some residual ISI existing between OFDM symbols. As shown in Fig. 3.6 (a) and (c) (assuming one sub-channel and representing the OFDM symbol by a sine wave), the end of the delayed version of the last symbol introduces interference to the current symbol when integrating over the FFT period, producing the ISI. In addition, the dispersion also destroys the orthogonality between subcarriers and gives rise to the interference between sub-channels, named ICI, the effect of which is illustrated in Fig. 3.7 (a1) and (a2).

In this example, 'ubcarrier 1' and delayed 'subcarrier 2' are shown separately during one OFDM symbol interval (an OFDM symbol is the sum of several subcarriers). Assume

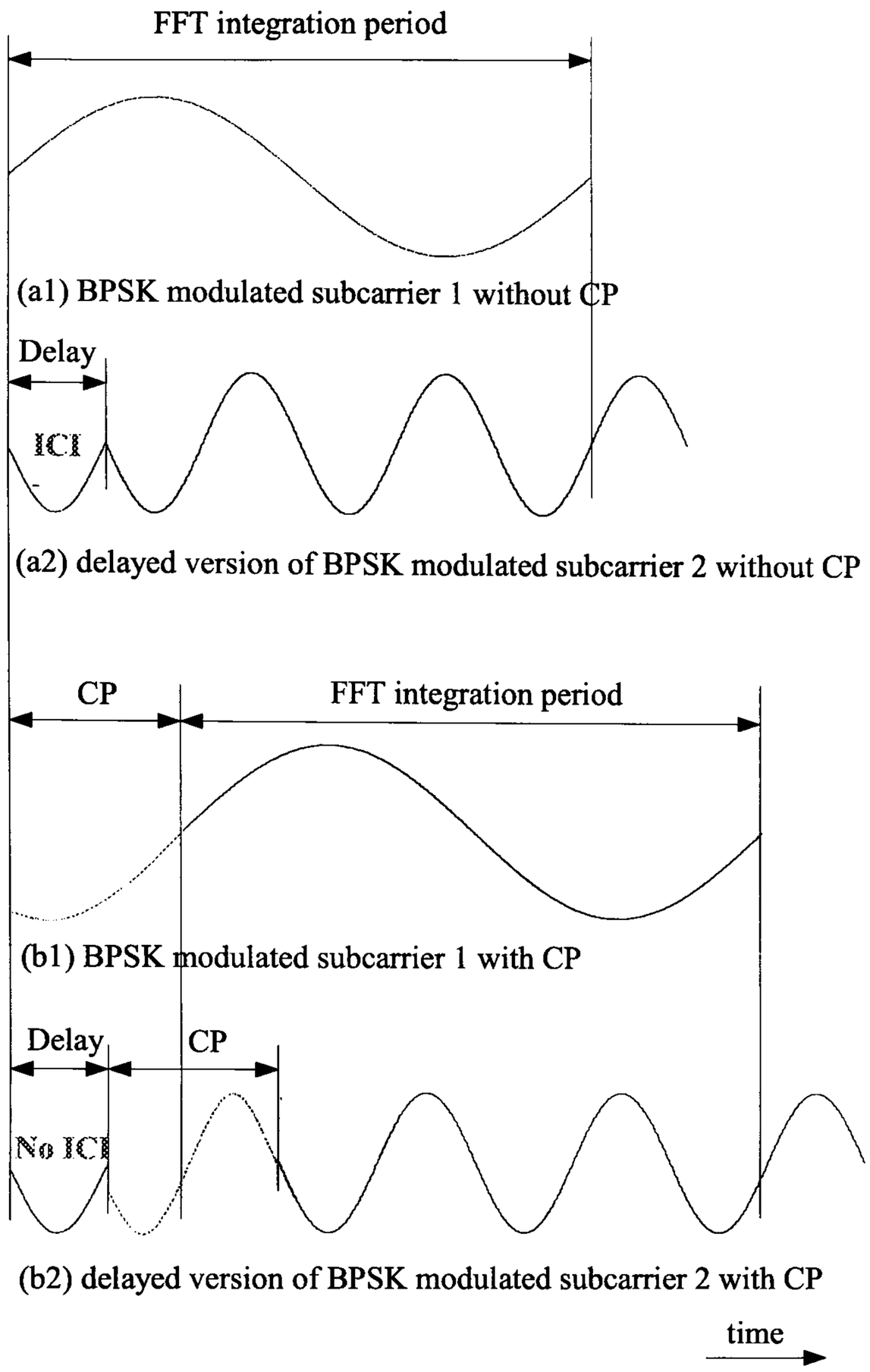


Figure 3.7: Elimination of ICI by adding CP to the OFDM symbol

each subcarrier is BPSK modulated, which means there may be a 2π-degree phase change at the OFDM symbol boundaries (Fig. 3.7 (a2)). If delay exists, the phase 'switching point' of the delayed subcarrier will fall within the FFT window. In this case, the difference in the number of cycles of subcarriers is not an integer number; therefore subcarriers are no longer orthogonal to each other. When de-modulating (FFT), the phase jump in some subcarriers produces a certain level of interference to other subcarriers.

The effect of ISI and ICI can be eliminated by introducing a time-domain cyclic prefix (CP) (Fig. 3.5) after the modulation (IFFT), and discarding it before the de-modulation (FFT). Fig. 3.6 (b) and (c) show how the introduction of CP eliminates ISI. As long as the

length of CP is longer than the multipath delay spread, the delayed part of the previous OFDM symbol will fall within the CP period and be deleted when the CP is removed. The previous OFDM symbol dies away before the current symbol arrives, therefore the current OFDM symbol will not be affected when integrating over the FFT interval. To remove ICI, CP cannot be any other value but the exact duplicate of the last samples[1] of the data portion of a OFDM symbol. This is to ensures that the delayed signals on subcarriers have an integer number of cycles within the FFT interval (Fig. 3.7 (b1) and (b2)). Again, provided the delay is smaller than the length of CP, the phase 'switching point' disappears after the CP is deleted. As a result, ICI is eliminated.

The insertion of CP certainly introduces overhead, which costs a reduction in bandwidth efficiency depending on how long the CP is relative to the symbol period. A typical overhead of CP is 25% of the OFDM symbol period, e.g., for a FFT size of 64, the CP length will be 16. The impact of CP is similar to the 'roll-off factor' in raised-cosine filtered single-carrier systems. Since OFDM has a wide spectrum, a large fraction of the allocated bandwidth can be utilized for data transmission, which helps to moderate the loss in efficiency due to CP.

3.3 Equalization

Although OFDM is robust to multipath dispersion, it still encounters different amplitude attenuations and phase shifts between sub-channels, due to the unevenness of the channel frequency responses. This can be corrected by means of equalization, which performs the inverse filtering effect of the channel. In particular, equalization can be simply performed in OFDM by multiplying the received signal on each sub-channel by the complex conjugate of the sub-channel frequency response separately, so that the original signal is re-created before demodulation. In fact, the simple equalization in frequency domain is

[1]One OFDM symbol contains N_s samples, where N_s is the number of OFDM subcarriers

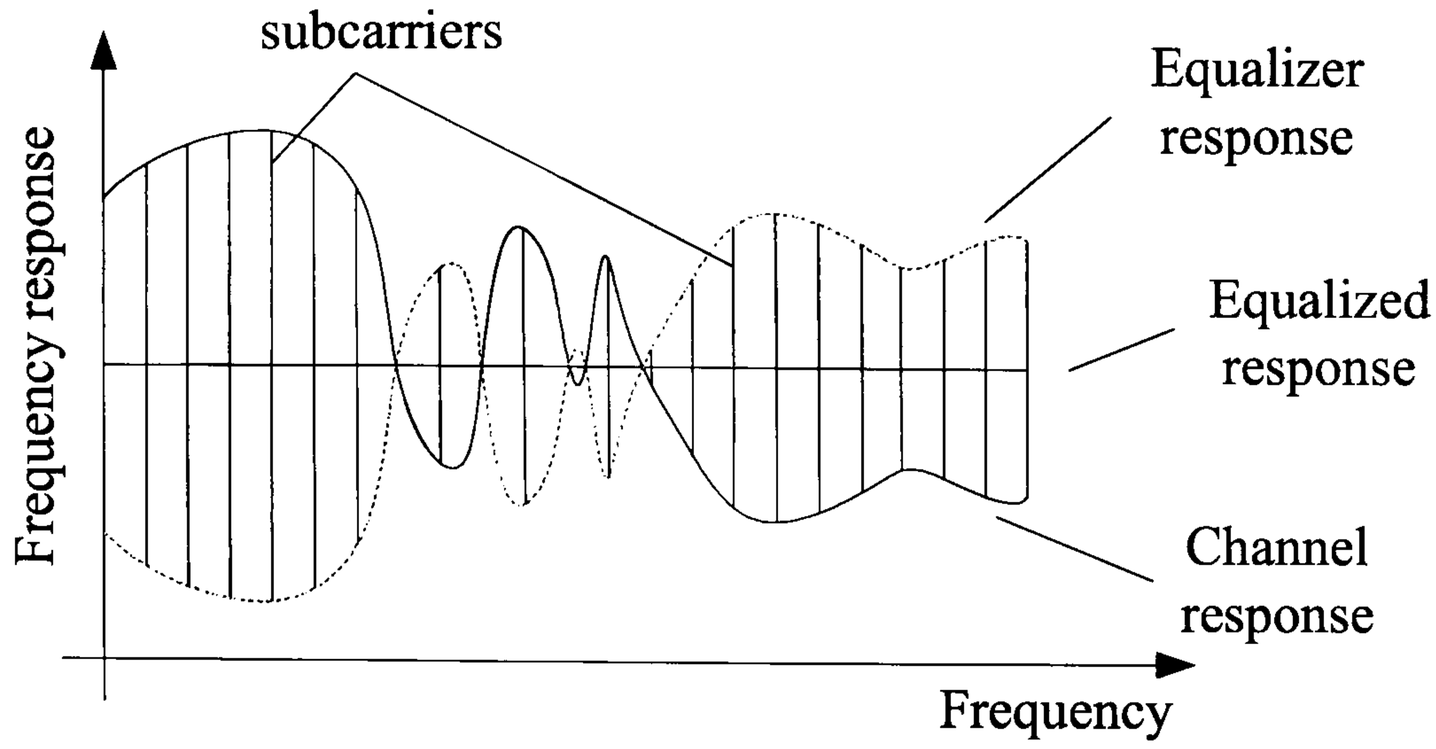


Figure 3.8: Frequency-domain equalization in OFDM

one of the advantages of OFDM. Fig. 3.8 describes the rationale behind the frequency-domain equalization in OFDM. For an individual sub-channel, the equalization is more or less a single-tap filter.

For coherent modulation, it is necessary to know the frequency responses of sub-channels at the receiver in order to equalize. This is normally provided by channel estimation using the pilot symbols inserted on some sub-channels at the beginning of the transmission.

Fig. 3.9 (without equalization) and Fig. 3.10 (with equalization) show how the multipath channel changes the amplitude and phase of the original signal and how the equalization works to correct it. To show the effect of equalization more clearly, receiver noise is set to be zero in the simulations. The number of subcarriers is 64 and for a channel order of 3, the length of CP is chosen to be 4. The modulation scheme used is BPSK. It is obvious from the results that equalization is an effective and simple way to correct the amplitude variations and phase shift in OFDM caused by the multipath channel.

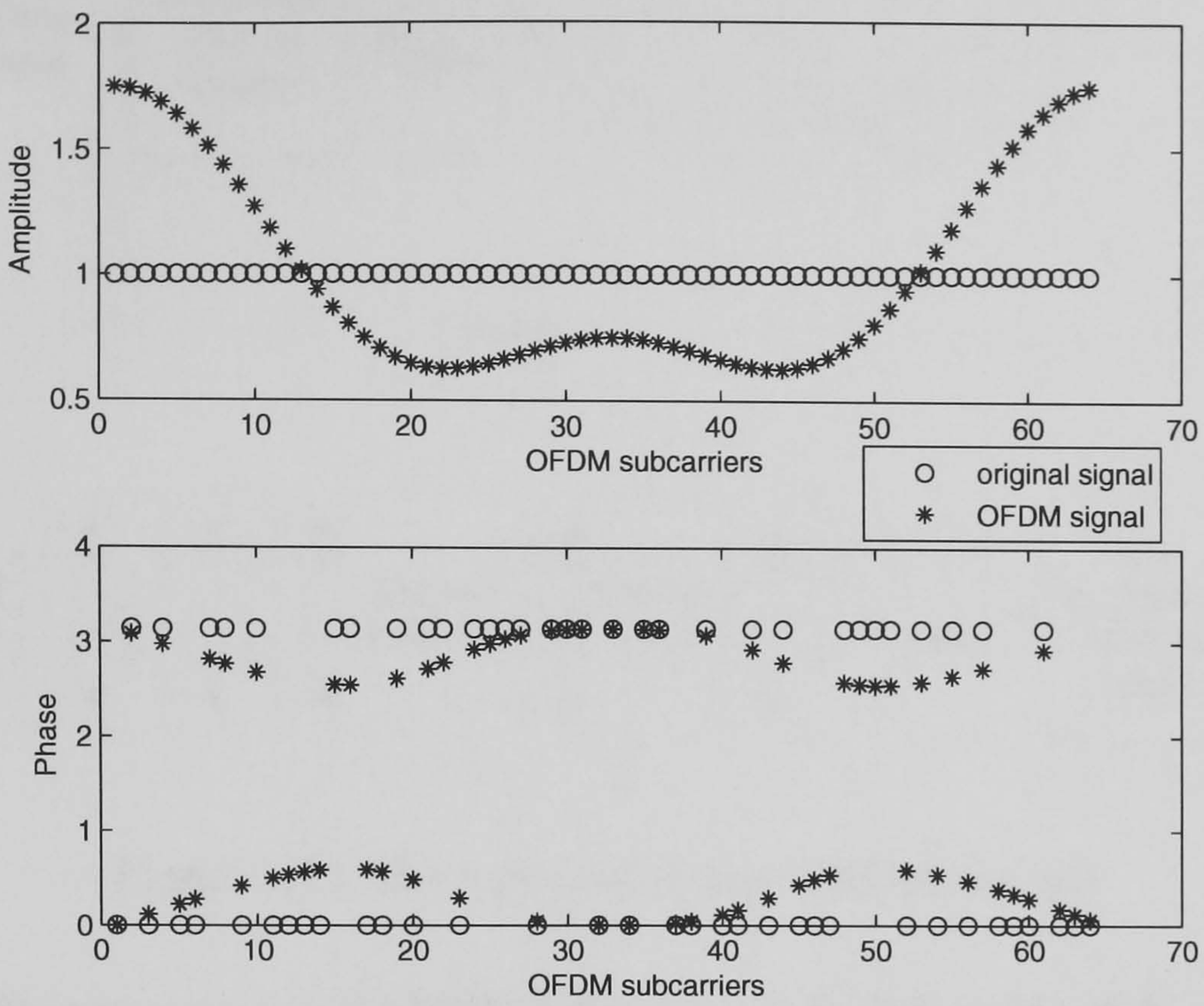


Figure 3.9: Amplitude and phase changes in OFDM sub-channels without equalization

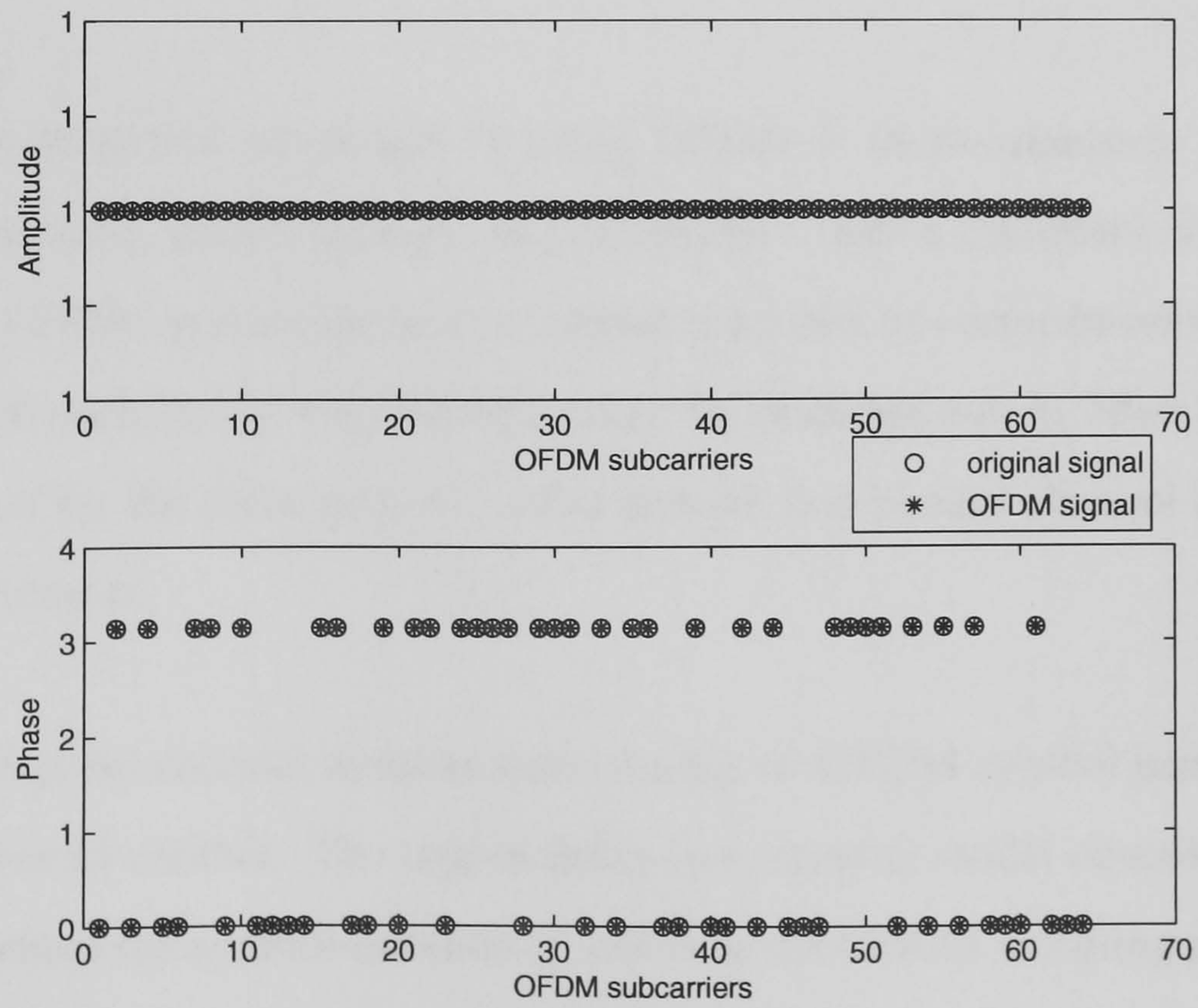


Figure 3.10: Amplitude and phase recovery for OFDM sub-channels by equalization

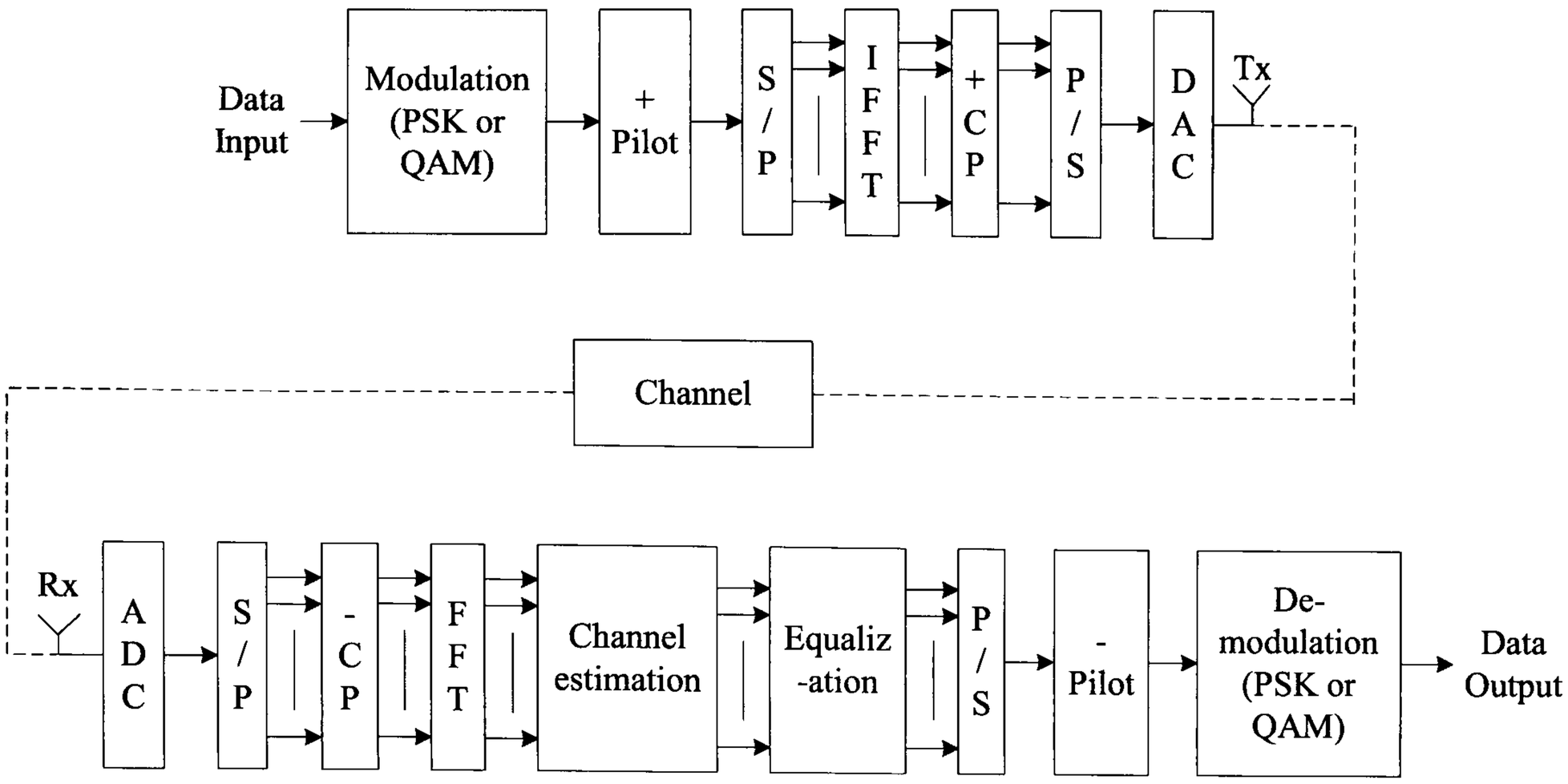


Figure 3.11: Block diagram of basic OFDM structure

3.4 Performance of OFDM over Multipath Channels

With all the schemes described above, the basic structure of OFDM is illustrated in Fig. 3.11.

Since the most attractive advantage of using OFDM is its resistance to the multipath effect of the channel, in this section, its performance over a multipath channel will be discussed. An OFDM system can be considered as a block or vector transmission system. consideration is restricted to one symbol, since the transmission in other blocks can be readily analyzed by the same process. Also assume that perfect channel information is known at the receiver.

It is assumed that the channel remains static during an OFDM symbol period and varies from one symbol to another. The tapped delay-line channel model described in Section 2.1.1 is used, where the symbol duration T_s refers to the OFDM sampling period T_s. The channel taps are equally spaced by one sample and have a maximum delay of $(L-1)$

samples. The vector form of the channel impulse response is given by

$$\mathbf{h} = \begin{bmatrix} h_1 & h_2 & ... & h_L \end{bmatrix} \tag{3.5}$$

where h_i $(i = 1, 2...L)$ represents the value of the i-th channel tap. The vector of input data symbols is written as $\mathbf{S} = [S_1\ S_2\ ...\ S_{N_s}]^{\mathrm{T}}$. These symbols are to be mapped in parallel on to N_s sub-channels, and $S_k(k = 1, 2...N_s)$ denotes the data symbol on the k-th sub-channel. After the N_s-point IFFT, the symbol is extended by a length-N_{cp} CP ($N_{cp} \geq (L-1)$ to avoid ISI/ICI), creating a transmission vector

$$\mathbf{s} = \begin{bmatrix} s_{N_s-N_{cp}+1} & s_{N_s-N_{cp}+2} & \cdots & s_{N_s} & s_1 & s_2 & \cdots & s_{N_s} \end{bmatrix}^{\mathrm{T}} \tag{3.6}$$

which is then transmitted serially through the channel. The received signal is simply the convolution of the transmitted symbol and the channel impulse response:

$$\mathbf{r} = \mathbf{s} * \mathbf{h} + \mathbf{n} \tag{3.7}$$

where $\mathbf{n}$ is a zero-mean AWGN vector. The received signal is then CP removed and FFT demodulated. Note that with the CP extension, the convolution in Eq. 3.7 is a cyclic convolution. A well-known property of FFT is that cyclic convolution in the time domain results in multiplication in the frequency domain. Therefore, it is easy to rewrite Eq. 3.7 in its equivalent frequency-domain expression:

$$\mathbf{R} = \mathbf{HS} + \mathbf{N} \tag{3.8}$$

where $\mathbf{R}$, $\mathbf{S}$ and $\mathbf{N}$ are $N_s \times 1$ vectors whose elements are N_s-point FFT of $\mathbf{r}$, $\mathbf{s}$ and $\mathbf{n}$, respectively; $\mathbf{H}$ is a diagonal matrix whose diagonal elements are the N_s-point FFT of $\mathbf{h}$ and non-diagonal elements are zeros. $\mathbf{H}_{k,k}$ is the complex channel gain (frequency response) of the k-th $(k = 1, 2...N_s)$ sub-channel. $\mathbf{R}$ is then equalized using the estimated channel frequency responses. From Eq. 3.8, it is obvious to see that OFDM is able to convert a frequency selective fading channel into N_s perfectly flat fading sub-channels. This provides another way to understand why OFDM eliminates ISI/ICI.

Fig. 3.12 gives the performance of OFDM with 256 sub-channels over multipath channels. The power delay profile of the channel is uniformly distributed and the channel

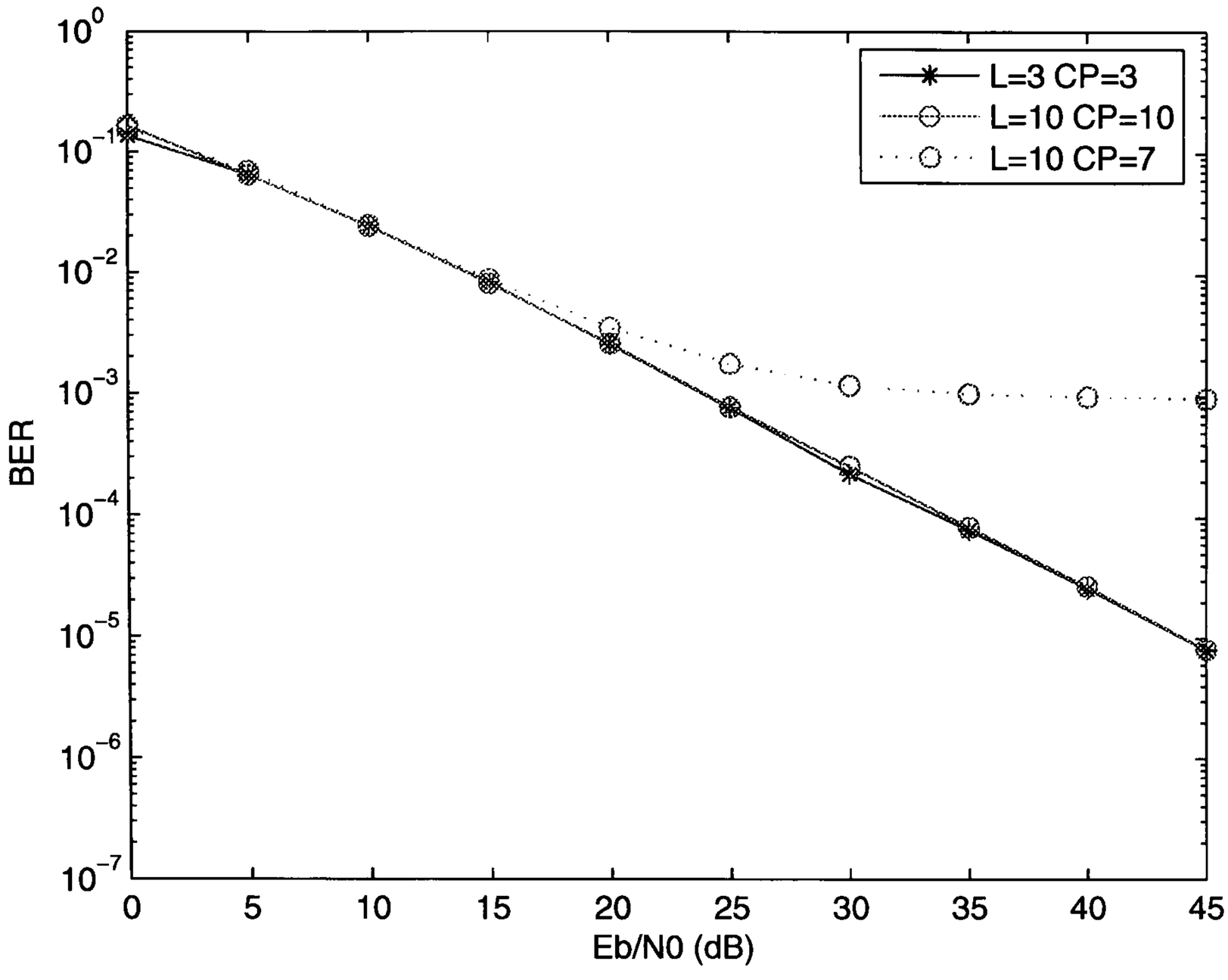


Figure 3.12: Performance of OFDM in multipath channels

orders are 3 and 10 respectively. The figure demonstrates that as long as the delay spread of the channel is shorter than the CP, OFDM offers an efficient way to overcome the deleterious effect of multipath. Otherwise, the existing ISI/ICI introduces an irreducible error floor in the high E_b/N_0 region, as in the case when the CP is shorter than the channel delay ('$L = 10, CP = 7$'). However, it is also noticed that the diversity does not increase as the number of channel paths increases, which indicates that OFDM cannot offer frequency diversity alone.

3.5 Diversity Realization by FEC Coding

As is known, the use of OFDM does not increase the total bandwidth of a system, and therefore the bandwidth efficiency is not reduced. The bandwidth of an OFDM system is

close to[2] that of a single-carrier system carrying the same data rate. The overall bandwidth is just divided between sub-channels. However on the other side, this indicates that no additional diversity advantage is available on fading channels through the use of OFDM alone [1]. This can be understood in another way that OFDM transfers a frequency selective fading channel into a number of flat fading channels, therefore the diversity offered by the channel is lost in sub-channels.

Since the overall BER of the system is dominated by that of the most severely faded sub-channels [1], the performance of such a system could be rather poor on a fading channel. This is because in non-adaptive OFDM, the power and bit rate of each sub-channel are fixed, and some of sub-channels might be severely faded by the frequency-selective fading channels. Fortunately, this problem can be solved by employing FEC coding, which is famously known as coded orthogonal frequency division multiplexing (COFDM). Coding and interleaving are essential for OFDM to compensate for fading sub-channels and yields a significant diversity improvement. Note that FEC coding must be powerful enough to correct the sampled errors lying in the frequency selectivity null regions. Also, the interleaver should be able to distribute the coded bits across all the sub-channels and separate the adjacent bits by more than the coherence bandwidth of the channel. To realize this, the total bandwidth should be much greater than the coherence bandwidth.

Similar to the convolutional diversity gain over the time-variant fading channel, the maximum diversity gain provided by COFDM over a frequency selective fading channel is determined by the minimum Hamming distance d_{min} of the code, provided the interleaver is effective enough. It is also true that the channel response depends on L independently fading taps of the multipath channel, which shows that the maximum diversity order is also given by L. Limited by each other, the maximum diversity is the minimum of the

[2]It is not identical because OFDM efficiency is reduced by the CP and the side lobes at the edge of the multiplex, while the single carrier efficiency is reduced by the roll-off factor of the Nyquist filtering.

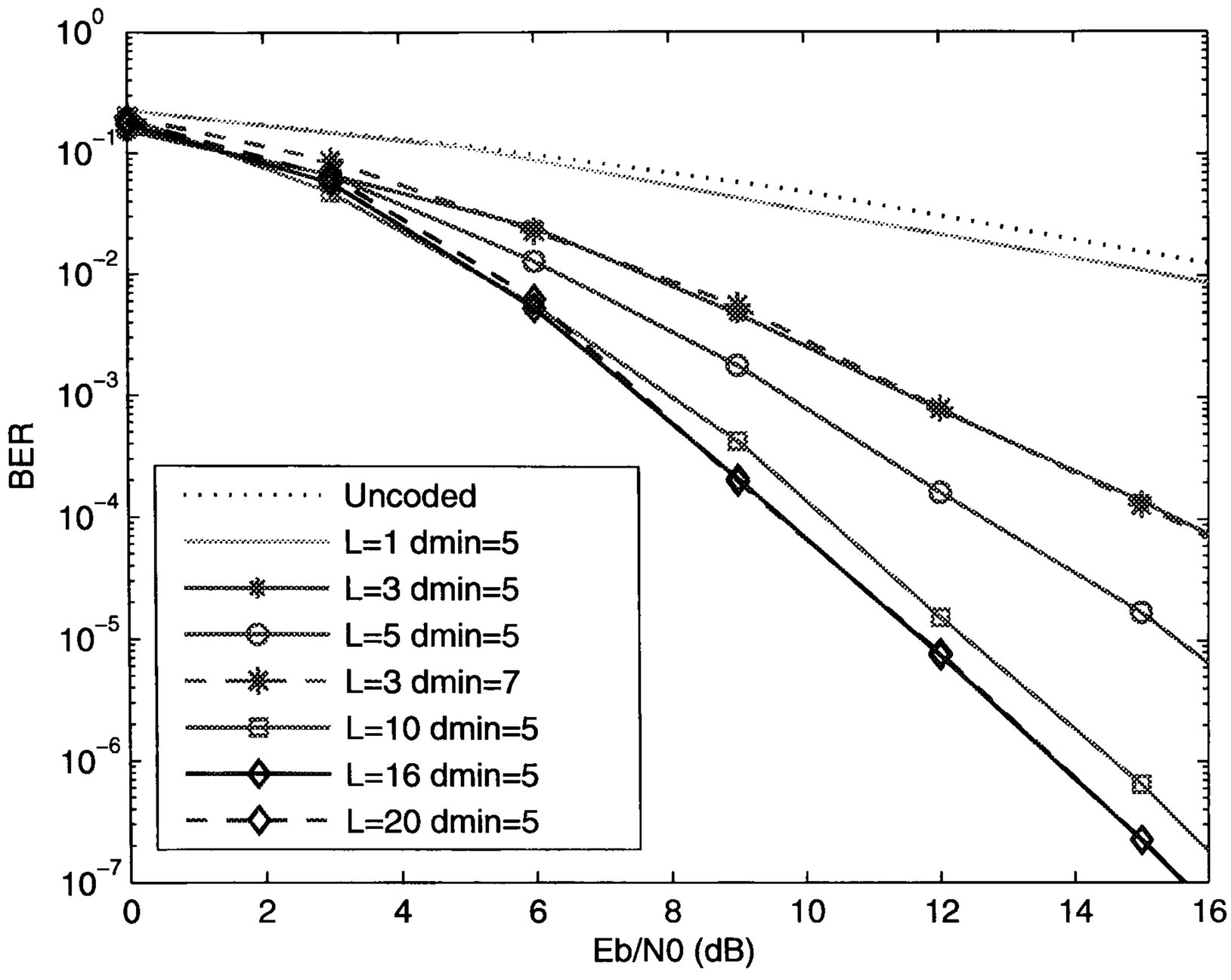


Figure 3.13: Performance of convolutional coded OFDM over multipath channels

two.

$$\text{diversity} = \min(d_{min}, L) \tag{3.9}$$

Nevertheless, even with interleaving, the full extent of the diversity advantage indicated above may not be achieved in practice. The resulting correlations between coded symbols will affect the performance of coded modulation schemes in the form of a decrease in diversity gain.

To verify this, Fig. 3.13 gives the performance of convolutional coded OFDM with different code strengths and channel orders. The codes used here has a minimum Hamming distance of 5 and 7, obtained from rate 1/2, constraint length 3 and 5 codes respectively. The number of OFDM sub-channels is 256, and a 16×16 block interleaver[3] is chosen.

[3]For an optimum block interleaver (read in rows and read out in columns), the length of the row should be longer than three times constraint length of convolutional codes, while that of the column should be longer than the channel coherence bandwidth

The taps of multipath channel have an uniform distributed power delay profile, and are assumed to be sample-spaced with an order of L. The length of CP is chosen to be the same as the channel order L, in order to minimize the overhead due to CP. Suppose the OFDM symbol period is T, therefore the sub-channel spacing is $1/T$. The delay spread of the channel can be roughly calculated by $\Delta=\frac{T}{256}L$. The coherence bandwidth is approximately the inverse of the delay spread, resulting in $\frac{256}{TL}$. Hence, the coherence bandwidth equals to the bandwidth of $(\frac{256}{TL})/(\frac{1}{T}) = \frac{256}{L}$ sub-channels. The 16×16 interleaver is able to spread the adjacent bits to be more than 16 sub-channels apart. Therefore, to ensure that the interleaver works most effectively, $\frac{256}{L}$ should be less than 16, giving in a channel order $L \geq 16$.

It is observed from Fig. 3.13 that in the case $L \leq d_{min}$ (d_{min}=5, L=1,3,5 in the figure), the diversity is determined by L and increases as L gets longer. Even if d_{min} increases, the available diversity still remains the same, as long as L is less than d_{min}. Examples of d_{min}=5 and d_{min}=7 are compared in the figure, where in both cases L=3. It is seen that the performances of the two are nearly identical. However, the diversity will be limited by d_{min} when $L \geq d_{min}$ (d_{min}=5, L=10,16,20 in the figure). It is true that there is an diversity increase from L=10 to L=16. This is because when $L < 16$, the interleaver is not totally effective. There are still some remanent correlations between coded bits. Any increase in L will make the interleaver perform better and result in a diversity improvement. However, this improvement disappears when the interleaver reaches its highest performance ($L \geq 16$), since in this case the coherence bandwidth ($\frac{256}{L}$) is less than the interleaving depth (16). The maximum diversity will be truly dominated by the minimum Hamming distance of the code.

Fig. 3.14 also gives the performance of turbo coded OFDM over multipath channels. The turbo encoder used in the simulation includes two 8-state RSC encoders, hence has a rate of 1/3. The generator polynomials are (13,15) for its feedback and output connections respectively. The length of the frame is 640, and an s-parameter of 16 is chosen for the s-random interleaver. The OFDM symbol has 1932 bits (46×42 block interleaver), which

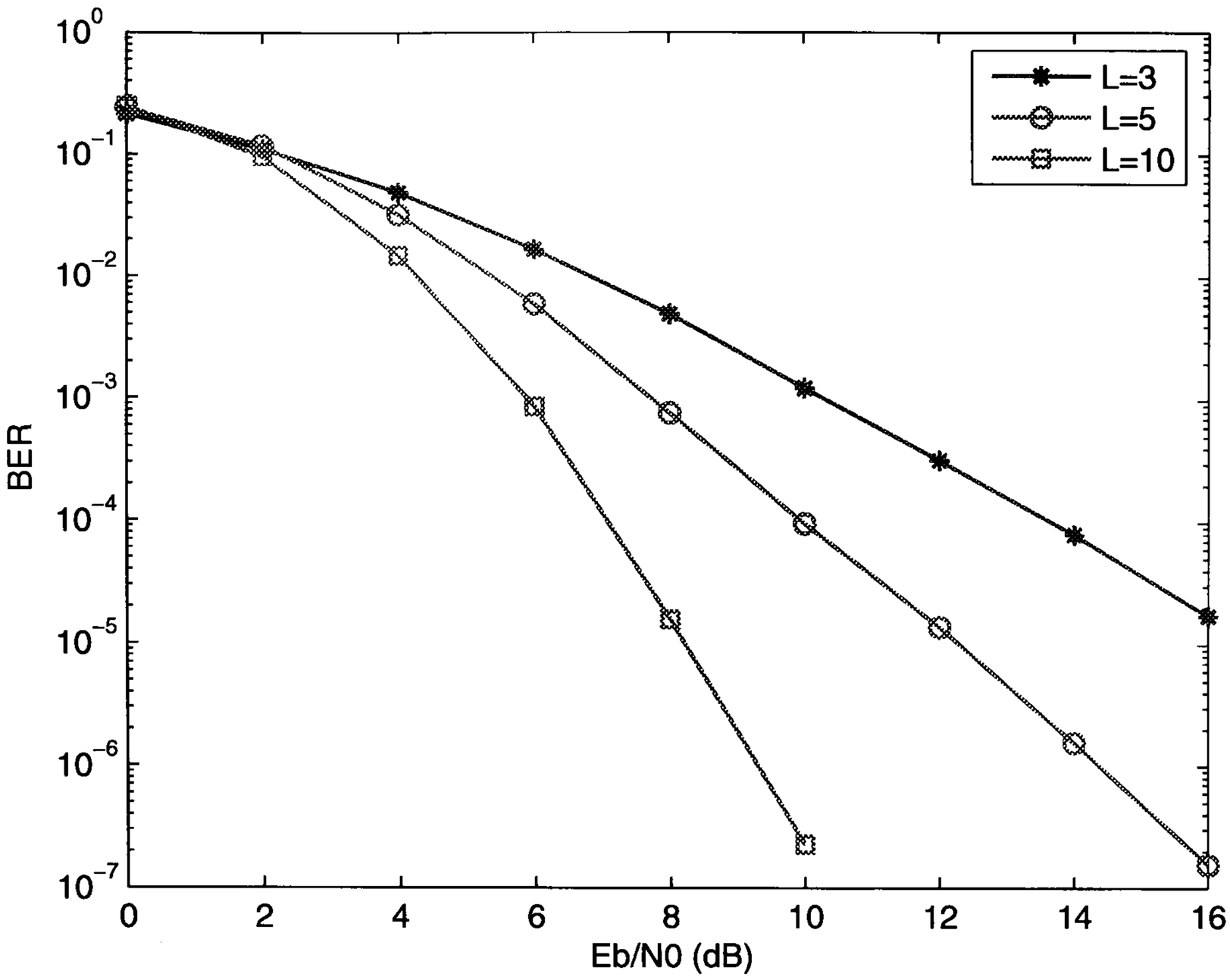


Figure 3.14: Performance of turbo coded OFDM over multipath channels

results in an OFDM with 966 sub-channels by QPSK modulation. The decoder uses the log-MAP algorithm and the number of iterations is 10. Again, the multipath channel has an uniformly distributed power delay profile and the channel orders are chosen to be 3, 5 and 10 for comparisons. From the results it is clear that since turbo codes are strong, the diversity is mainly determined by the number of taps of the channel and increases as the number of taps increases.

3.6 Summary

This chapter has given some background information about OFDM, including the application of OFDM, the generation of OFDM subcarriers, the importance of adding CP and the frequency-domain equalization. Famous for its resistance to the multipath effect, the

performance OFDM over multipath channels has been evaluated. It has been concluded that as long as the length of CP is longer than the delay spread of the channel, the BER performance degradation due to multiple channel delays can be eliminated, although at the cost of the overhead in CP. Furthermore, since coding is necessary for OFDM systems to realize the frequency diversity, the BER performance of FEC coded OFDM systems has also been investigated. It has been clear that with convolutional coding, the maximum diversity that can be realized in such systems is the minimum between the channel delay spread and the minimum Hamming distance of the code, provided that the interleaver is absolutely efficient. However, because turbo codes are much more powerful than the convolutional codes, the diversity provided by turbo coded OFDM mainly depends on the delay spread of the channel.

Chapter 4

Orthogonal STBC

Contents

This chapter gives a short introduction to orthogonal STBC [7], with emphasis on its decoding techniques.

4.1 MIMO Techniques

The major challenges in future wireless communication systems are the increasing requirements on data rate and quality of service [8]. Since wireless transmissions are impaired by fading and interference, new techniques are called for to increase the spectral

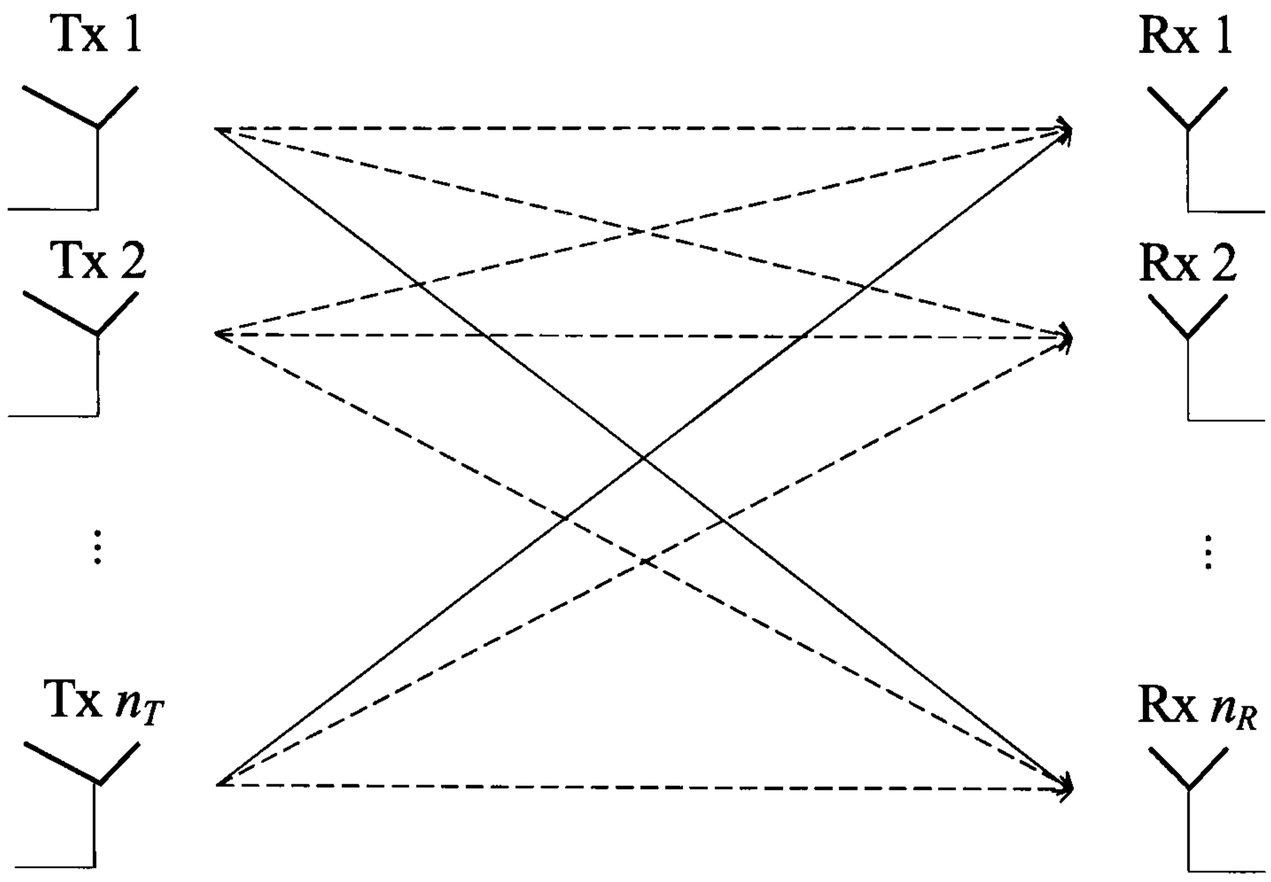


Figure 4.1: Structure of MIMO

efficiency and improve the link reliability. MIMO [8], which uses multiple antennas at both ends of a wireless link (Fig. 4.1), is regarded as one of such techniques.

Diversity in wireless systems is an effective way to combat fading caused by multipath effects. The rationale behind this is to receive several replicas of the information signal through independent fading channels, with the hope that at least one of these links will not fade, so that the original signal can be recovered successfully. There are three main forms of diversity: time diversity, frequency diversity and spatial diversity. Among these, spatial diversity has attracted a great deal of attention in recent years, and it can be realized by using one of the MIMO techniques: space-time coding [39]. The basic idea here is to send the information signal with different processing (coding) before transmitted from different antennas at different time slots, so that the receiver can combine these signals to obtain diversity. Channel information is not provided at the transmitter, but need to be estimated at the receiver in order to decode. Assuming the number of transmit and receive antennas are n_T and n_R respectively, the maximum diversity obtained by the MIMO system is $n_T \times n_R$, provided that $n_T \times n_R$ links fade independently (by separating the antennas more than the coherence distance [8]).

Theoretically, both ends can use multiple antennas to gain transmit diversity as well as

receive diversity. However in the downlink transmission, receive diversity is not preferred, due to the size, weight and battery power consumption of the handset. Therefore, transmit diversity at the base station has become an active area of research in the last few years [7, 10, 11, 39–41].

In general, space-time codes are classified into space-time trellis codes (STTC) [39] and STBC, while STBC includes orthogonal STBC and quasi-orthogonal STBC [42]. In particular, orthogonal STBC are attractive for practical application, since they provide full transmit diversity (due to the block-constant flat fading channel assumption) while ML decoding can be replaced by low-complexity processing (due to the orthogonality of the codes) at the receiver. Therefore, this thesis is mostly concerned with the orthogonal STBC transmit diversity scheme.

4.2 Encoding Structure of Orthogonal STBC

Eq. 4.1 gives the coding matrix of the famous Alamouti scheme for a 2-transmit-antenna (2Tx) MIMO system [10] ($*$ is the conjugate).

$$\mathbf{G}_2 = \begin{bmatrix} s_1 & s_2 \\ -s_2^* & s_1^* \end{bmatrix} \tag{4.1}$$

At a given symbol period, two PSK or QAM modulated symbols are simultaneously transmitted from two antennas. The signals transmitted from one antenna at the first time instant is s_1 , followed by $-s_2^*$ in the next instant. Meanwhile, s_2 is transmitted from the other antenna firstly, after which s_1^* is sent out. This can be regarded as a block transmission, during which two symbols are transmitted in two time slots, giving a coding rate of unity.

Alamouti's simple 2-Tx coding scheme has been extended by Tarokh [11] to higher-order transmit antenna schemes. The used in this thesis is the half rate orthogonal STBC for

four transmit antennas. The coding structure is given by

$$\mathbf{G}_4 = \begin{bmatrix} s_1 & s_2 & s_3 & s_4 \\ -s_2 & s_1 & -s_4 & s_3 \\ -s_3 & s_4 & s_1 & -s_2 \\ -s_4 & -s_3 & s_2 & s_1 \\ s_1^* & s_2^* & s_3^* & s_4^* \\ -s_2^* & s_1^* & -s_4^* & s_3^* \\ -s_3^* & s_4^* & s_1^* & -s_2^* \\ -s_4^* & -s_3^* & s_2^* & s_1^* \end{bmatrix} \tag{4.2}$$

During this block transmission, four symbols are sent out from four transmit antennas within a total of eight time instants, giving a code rate of 1/2.

The orthogonal property of STBC refers to the relationship between the columns of the code matrix $\mathbf{G}_i$, where i are the numbers of transmit antennas, which satisfies

$$\mathbf{G}_i^{\mathrm{H}}\mathbf{G}_i = \sum_{i=1}^{n_T} |s_i|^2 \mathbf{I} \tag{4.3}$$

Here, I denotes an identity matrix. This indicates that signal transmitted from different antennas are designed orthogonal to each other. The column-orthogonality of STBC code matrix allows a simple linear decoding scheme at the receiver.

4.3 Decoding Algorithm for Orthogonal STBC over Block-constant Flat Fading Channels

Assume that the channel impulse response is constant during the block period, and there is no time dispersion (i.e the impulse response contains one component). The decoding process of the Alamouti scheme shall be described, which can be readily extended to that of the $\mathbf{G}_4$ code.

Considering one receive antenna, the received signal r_1 at the first time interval and r_2 at the second time interval are given by Eq. 4.4.

$$\begin{aligned} r_1 &= h_{11}s_1 + h_{21}s_2 + n_1 \\ r_2 &= -h_{12}s_2^* + h_{22}s_1^* + n_2 \end{aligned} \tag{4.4}$$

where h_{it} $(i = 1, 2;\ t = 1, 2)$ are the impulse responses of the flat fading channel from the i-th transmit antenna to the receive antenna at the t-th time instant; n_i $(i = 1, 2)$ are AWGN noise samples.

The received signal in Eq. 4.4 can be written in a vector-matrix form as

$$\mathbf{r} = \mathbf{Hs} + \mathbf{n} \tag{4.5}$$

where $\mathbf{r} = [r_1\ r_2^*]^{\mathrm{T}}$, $\mathbf{s} = [s_1\ s_2]^{\mathrm{T}}$ and $\mathbf{n} = [n_1\ n_2^*]^{\mathrm{T}}$ are the received signal vector, transmitted signal vector and noise vector respectively (note that r_2 and n_2 are conjugated). The equivalent channel matrix is

$$\mathbf{H} = \begin{bmatrix} h_{11} & h_{21} \\ h_{22}^* & -h_{12}^* \end{bmatrix} \tag{4.6}$$

Assuming the perfect channel knowledge (PCK) can be acquired at the receiver, ML decoding is performed to detect symbols s_1 and s_2:

$$\hat{\mathbf{s}} = \arg \min_{s_m \in M} \{||\mathbf{r} - \mathbf{Hs}_m||^2\} \tag{4.7}$$

where $\mathbf{s}_m$ are vectors containing all the possible combination constellation values of s_1 and s_2. If the constellation size is $|M|$ (e.g.,$|M| = 4$ for QPSK and $|M| = 16$ for 16-QAM), the ML decoder will perform $|M|^{n_T}$ enumerations before finding the minimum. The computational complexity of ML decoder is high (growing exponentially with the number of transmit antennas), sometimes prohibitive in practice, although the ML decoder yields the best performance in terms of error rate.

Because of the orthogonality of the codes, the above ML decoding is equivalent to a simple MF decoding and a least square (LS) detection. First of all, the receive vector is

left-multiplied by the conjugate transpose of the channel matrix (MF decoding):

$$\hat{\mathbf{s}} = \mathbf{H}^{\mathrm{H}}\mathbf{r} = \mathbf{\Psi}\mathbf{s} + \mathbf{H}^{\mathrm{H}}\mathbf{n} \tag{4.8}$$

where the matrix $\mathbf{\Psi}$ is defined as

$$\mathbf{\Psi} = \mathbf{H}^{\mathrm{H}}\mathbf{H} = \begin{bmatrix} |h_{11}|^2 + |h_{22}|^2 & h_{11}^* h_{21} - h_{22} h_{12}^* \\ h_{21}^* h_{11} - h_{12} h_{22}^* & |h_{21}|^2 + |h_{12}|^2 \end{bmatrix} \tag{4.9}$$

Then, decomposing Eq. 4.8, the estimates of symbols s_1 and s_2 are separately given by

$$\begin{aligned} \hat{s}_1 &= (|h_{11}|^2 + |h_{22}|^2)s_1 + (h_{11}^* h_{21} - h_{22} h_{12}^*)s_2 + \tilde{n}_1 \\ \hat{s}_2 &= (h_{21}^* h_{11} - h_{12} h_{22}^*)s_1 + (|h_{21}|^2 + |h_{12}|^2)s_2 + \tilde{n}_2 \end{aligned} \tag{4.10}$$

where $\tilde{n}_1 = h_{11}^* n_1 + h_{22} n_2^*$ and $\tilde{n}_2 = h_{21}^* n_1 - h_{12} n_2^*$.

It is also assumed that channel impulse responses are constant over one block period (e.g., 2 symbol periods for Alamouti scheme) and may vary from one block to another. Therefore, $h_{11} = h_{12}$ and $h_{21} = h_{22}$. Eq. 4.10 can be simplified to

$$\begin{aligned} \tilde{s}_1 &= (|h_{11}|^2 + |h_{22}|^2)s_1 + \tilde{n}_1 \\ \tilde{s}_2 &= (|h_{21}|^2 + |h_{12}|^2)s_2 + \tilde{n}_2 \end{aligned} \tag{4.11}$$

Note that the subtractions in non-diagonal elements of $\mathbf{\Psi}$ result in zeros (Eq. 4.12), which indicates that under the assumption of a block-constant channel, no ISI exists during the decoding process.

$$\mathbf{\Psi} = \begin{bmatrix} |h_{11}|^2 + |h_{22}|^2 & 0 \\ 0 & |h_{21}|^2 + |h_{12}|^2 \end{bmatrix} \tag{4.12}$$

Finally, the symbols $\tilde{s}_1$ and $\tilde{s}_2$ are detected independently by a LS approach:

$$\hat{s}_i = \arg \min_{s_m \in M} \{|\tilde{s}_i - \psi_{ii} \cdot s_m|^2\} \quad (i = 1, 2) \tag{4.13}$$

where ψ_{ii} $(i = 1, 2)$ are the diagonal elements of $\mathbf{\Psi}$. The complexity of the above MF decoding includes a matrix multiplication and two separate LS detections with $n_T|M|$ enumerations. Compared to that of ML decoding, the complexity of MF decoding is drastically reduced, because under the assumption of block-constant flat fading channels,

the vector detection problems (Eq. 4.7) is decomposed into independent scalar detection problems (Eq. 4.13).

It is clear from Eq. 4.12 that if the fading between different antenna pairs are independent and neither $h_{11}(h_{12})$ nor $h_{21}(h_{22})$ is faded, a second order diversity is obtained. This process performs similarly to the maximum-ratio combining (MRC) at the transmitter, except that STBC does not need to know the channel information before transmitting. Note that the diagonal elements of Ψ indicate the diversity gain provided, while the non-diagonal elements of Ψ gives the information of interference from other symbols in the same decoding block.

The decoding process for the half-rate $\mathbf{G}_4$ codes can be similarly obtained. Here, the channel matrix $\mathbf{H}$ in Eq. 4.5 can be expressed as

$$\mathbf{H} = \begin{bmatrix} h_{11} & h_{21} & h_{31} & h_{41} \\ h_{22} & -h_{12} & h_{42} & -h_{32} \\ h_{33} & -h_{43} & -h_{13} & h_{23} \\ h_{44} & h_{34} & -h_{24} & -h_{14} \\ h_{15}^* & h_{25}^* & h_{35}^* & h_{45}^* \\ h_{26}^* & -h_{16}^* & h_{46}^* & -h_{36}^* \\ h_{37}^* & -h_{47}^* & -h_{17}^* & h_{27}^* \\ h_{48}^* & h_{38}^* & -h_{28}^* & -h_{18}^* \end{bmatrix} \tag{4.14}$$

while the corresponding receive vector, transmitted signal vector and noise vector are $\mathbf{r} = [r_1\ r_2\ r_3\ r_4\ r_5^*\ r_6^*\ r_7^*\ r_8^*]^{\mathrm{T}}$, $\mathbf{s} = [s_1\ s_2\ s_3\ s_4]^{\mathrm{T}}$ and $\mathbf{n} = [n_1\ n_2\ n_3\ n_4\ n_5^*\ n_6^*\ n_7^*\ n_8^*]^{\mathrm{T}}$.

Again, assuming the flat fading channel remains constant during one block period, which includes eight symbol periods for $\mathbf{G}_4$ codes ($h_{i1} = h_{i2} = \ldots = h_{i8}$ $(i = 1, 2 \ldots 4)$), the

matrix $\mathbf{\Psi}$ can be simplified to

$$\mathbf{\Psi} = \begin{bmatrix} c & 0 & 0 & 0 \\ 0 & c & 0 & 0 \\ 0 & 0 & c & 0 \\ 0 & 0 & 0 & c \end{bmatrix} \tag{4.15}$$

Here, $c = 1/R_{\mathbf{G}_4}(|h_{11}|^2 + |h_{21}|^2 + |h_{31}|^2 + |h_{41}|^2)$ is a constant value ($R_{\mathbf{G}_4}$=1/2 is the rate of G_4 codes). It is clear from the non-diagonal elements of $\mathbf{\Psi}$ that no ISI is generated during the decoding process. Finally, s_1, s_2, s_3 and s_4 can be detected separately by a simple LS approach.

Note that if more than one receive antenna is employed, the MF decoding is carried out individually on each receive antenna, and a LS detection for each symbol based on received signals from all receive antennas is performed, realizing a full spatial diversity of $n_T \times n_R$ (Eq. 4.16).

$$\hat{s}_i = \arg \min_{s_m \in M} \{ | \sum_{j=1}^{n_R} \tilde{s}_i^j - \sum_{j=1}^{n_R} \psi_{ii}^j \cdot s_m |^2 \} \quad (i = 1, 2...n_T) \tag{4.16}$$

where j are the receive antenna indices ($j = 1, 2...n_R$). This process is similar to the MRC of the outputs of the detectors for each receive antenna.

Fig. 4.2 shows the BER performance versus E_b/N_0 of orthogonal STBC over block-constant flat Rayleigh fading channels. A full-rate Alamouti code and a half-rate $\mathbf{G}_4$ code are employed for the 2Tx system and the 4Tx system respectively. The corresponding modulation schemes are BPSK and QPSK, reaching a total rate of unity (1 bit/s/Hz). The MIMO channels are assumed to be independent flat fading and remain static during one block period. The average energy of each channel is normalized according to

$$\sigma_h^2 = \mathrm{E}\{|h_{ij}|^2\} = \frac{1}{n_R} \tag{4.17}$$

where h_{ij} are channel impulse responses from the i-th transmit antenna to the j-th receive antenna. The normalization is to ensure that the receive power is equal to the transmit

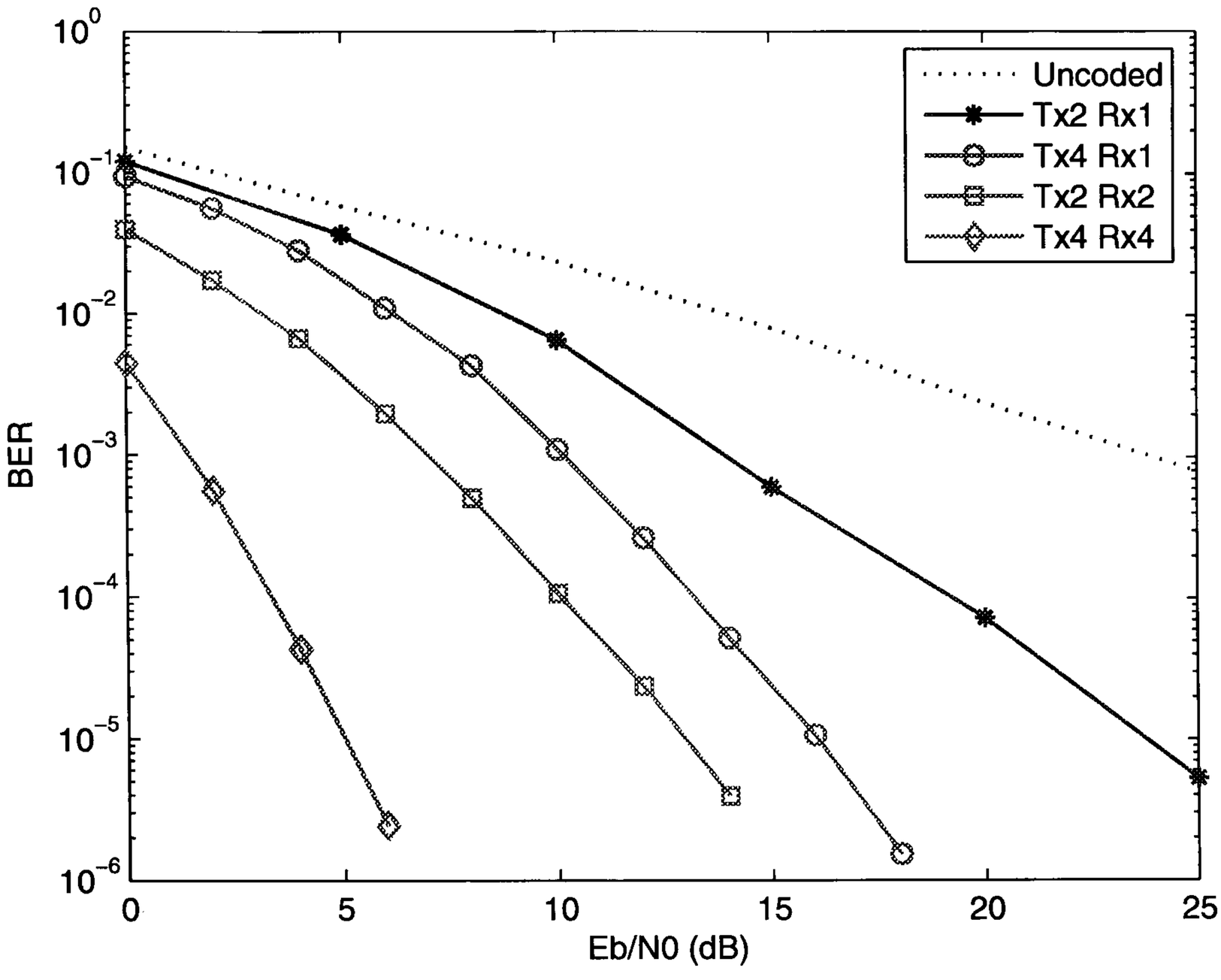


Figure 4.2: BER performance of orthogonal STBC over block-constant flat fading channels

power, so that the receive power does not increase as the number of receive antennas increases. It is assumed that PCK is known at the receiver. The results demonstrate that compared to the 'uncoded' case, where no diversity gain is provided, STBC can offer a diversity order equal to the multiplication of the number of transmit antennas and the number of receive antennas. Note that when there are multiple receive antennas ('Tx2Rx2' and 'Tx4Rx4'), an additional array gain is obtained due to the coherent combination of the signals on multiple receive antennas. The E_b/N_0 improvement by this array gain is in proportion to the number of receive antennas, e.g., 3dB for 2-Rx case and 6dB for 4-Rx case at the BER of 10^{-4}. Note that the array gain is not provided by multiple transmit antennas, since the channel information is unknown at the transmitter.

4.4 Summary

In this chapter, one of the MIMO techniques, orthogonal STBC, has been introduced. The encoding structure, in particular, in cases of 2 and 4 transmit antennas has been described. The focus has been put on the corresponding decoding algorithm over block-constant flat fading channels, and the BER performance evaluation of orthogonal STBC coded systems with various numbers of transmit and receive antennas over such channels. It has been concluded that because of the orthogonality of the codes, the decoding procedure can be simplified to a MF decoding and a LS detection. Furthermore, under the assumption of block-constant flat fading channels, the full transmit and receive diversity of orthogonal STBC can be realized.

Chapter 5

Interference Cancelation in SFBC-OFDM over Quasi-static Frequency Selective Fading Channels

Contents

As discussed in Chapter 4, orthogonal STBC at the transmitter side of a MIMO system is able to increase the link reliability by providing a transmit diversity equal to the number of transmit antennas. However, flat fading channels are only true for a narrowband wireless system. Frequency selective fading channels are more common in a wideband system. Since OFDM is resistant to the multipath effect, it has been suggested to com-

bine the orthogonal STBC with OFDM, so that the orthogonal STBC can perform well in a frequency selective fading environment. There are two combining schemes depending on how space-time coded signals are mapped onto OFDM subcarriers and symbols: STBC-OFDM and SFBC-OFDM.

In Chapter 4, a simple MF decoder for orthogonal STBC over flat fading channels has been introduced, which yields the ML performance under the assumption that channels remain static during a block period. Similarly, the MF decoding approach can be readily applied to the equivalent SFBC-OFDM over frequency selective fading channels. As long as the channel frequency response is approximately constant within a block, the ML performance can be reached. Meanwhile, spatial diversity can be obtained, which is identical to the one offered by STBC over block-static flat fading channels. However, if the channel frequency response varies rapidly within a block, which usually happens in a severe multipath environment, extra detection errors will occur in addition to those caused by the noise, giving rise to the ICI. In such a case, the BER performance will degrade at high E_b/N_0 regions, where an irreducible error floor exists. The amount of degradation depends on the channel delay spread as well as the number of subcarriers.

A zero-forcing (ZF) decoder has been proposed to eliminate the ICI in a 2-antenna SFBC-OFDM system using Alamouti codes [9]. Simulation results will show that the ZF decoder is capable of eliminating the ICI in such a system. However, the ZF decoding algorithm for the 4-Tx SFBC-OFDM system using $\mathbf{G}_4$ codes is too complicated to realize. Therefore instead, an iterative PIC scheme, which is originally used in the multi-user detection for code division multiple access (CDMA) system [43], is applied based on the MF decoding to mitigate the effect of ICI.

Inspired by the iterative principle of turbo decoding [18], soft bits from the log-MAP decoder are fed back as the input to the inner PIC processor to enhance its performance of interference cancelation. The outer iterations, working together with the inner MF-based PIC iterations, form an effective way of reducing ICI in the $\mathbf{G}_4$ SFBC-OFDM

over multipath channels, and can be readily applied to other SFBC-OFDM systems with higher-order antennas.

Finally, in all simulations in this chapter, it is assumed that fading between two antenna pairs is uncorrelated and PCK can be obtained at the receiver.

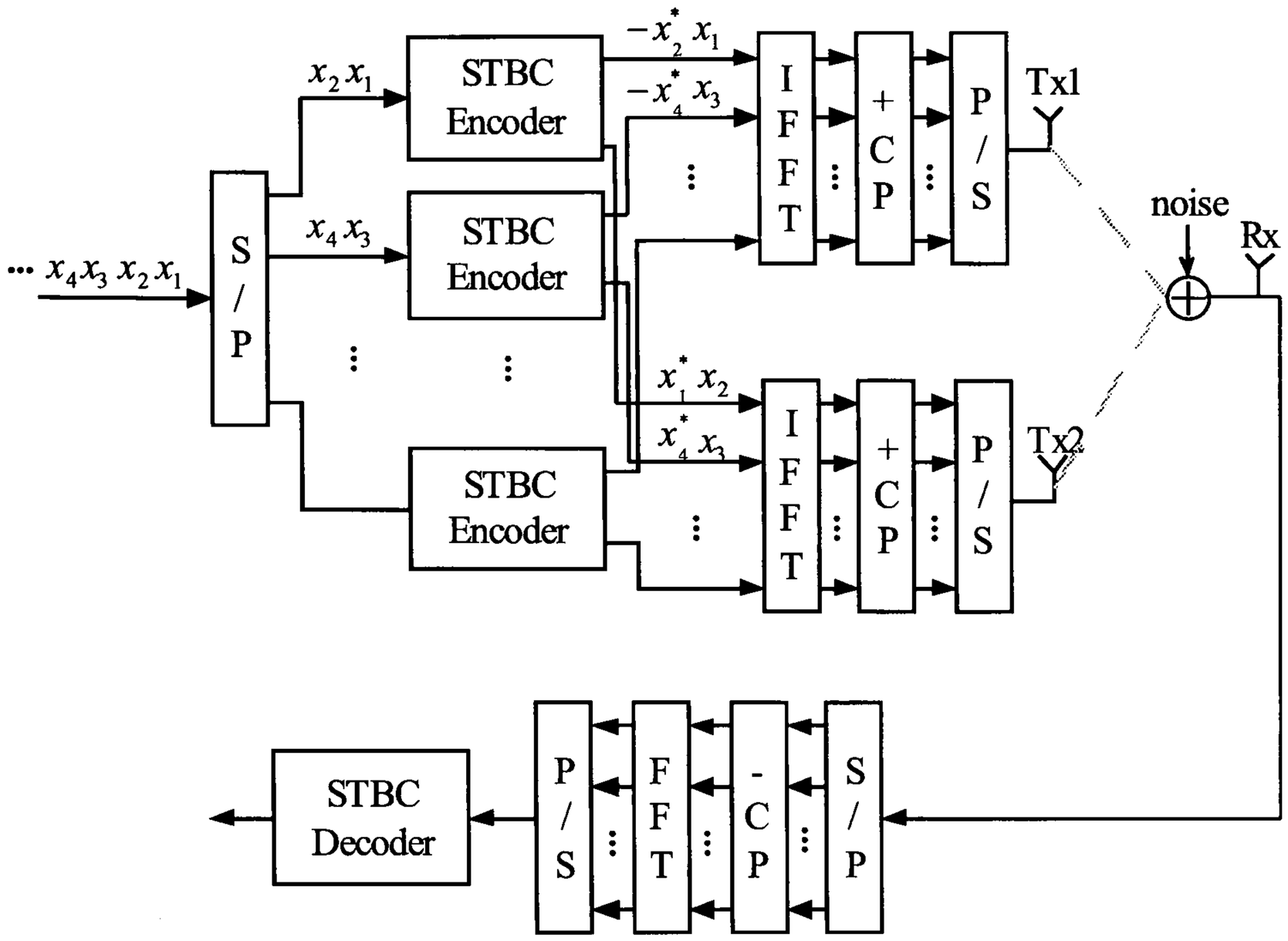
5.1 STBC-OFDM and SFBC-OFDM

By inserting a CP, OFDM transfers a frequency selective fading channel into a number of flat fading channels for its subcarriers. Therefore orthogonal STBC, which performs ideally on flat fading channels, are well suited to be applied to OFDM. At each transmit antenna, the symbols in a STBC code matrix can either be mapped onto one of the OFDM subcarriers in successive OFDM symbols (STBC-OFDM), or onto several adjacent OFDM sub-channels in parallel within a symbol. The latter is called space-frequency block coded OFDM (SFBC-OFDM). Fig. 5.1(a) and Fig. 5.1(b) illustrate the structure of 2-antenna STBC-OFDM and SFBC-OFDM systems using Alamouti codes.

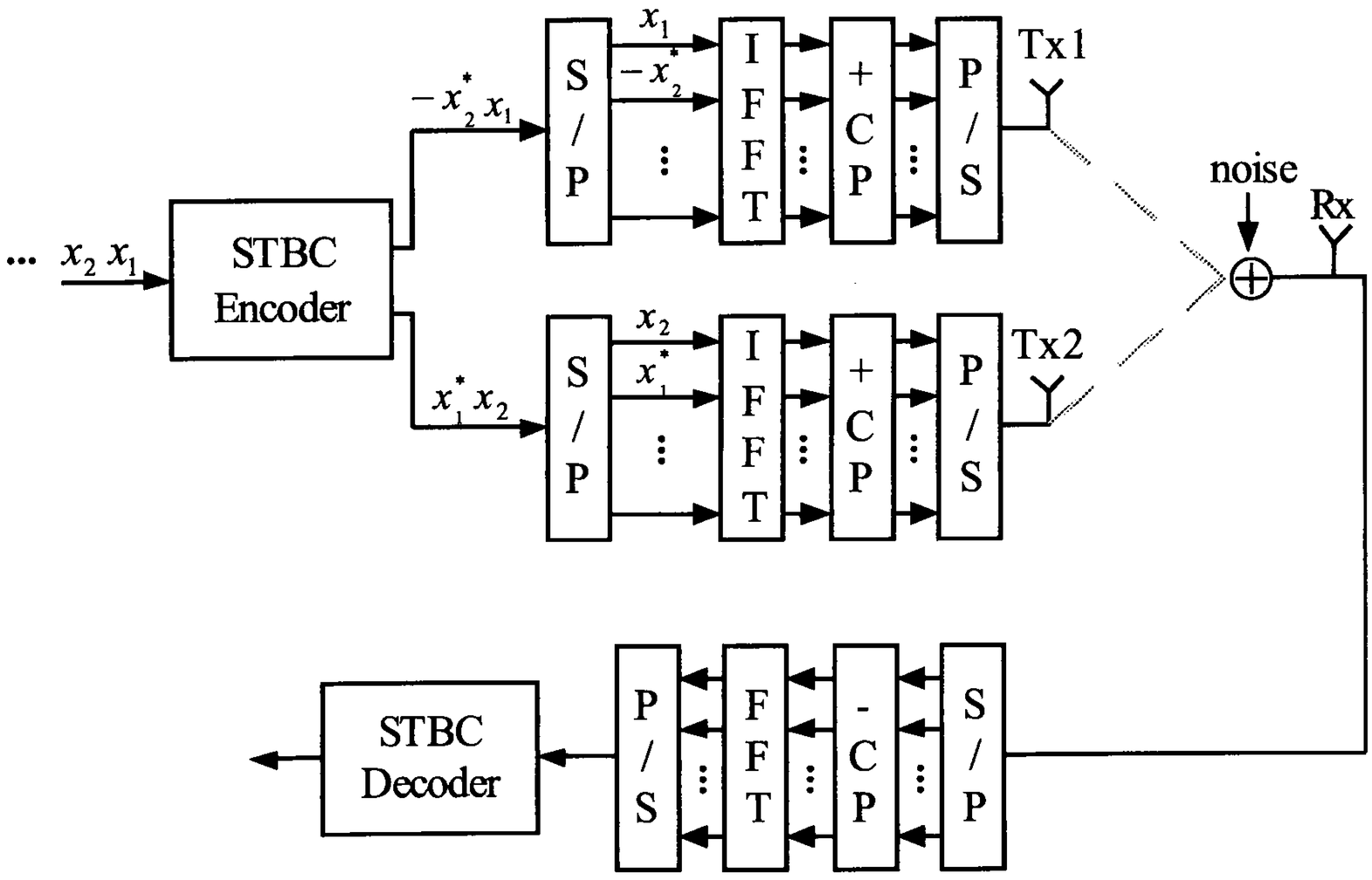
The MF decoder for orthogonal STBC, introduced in Chapter 4, can be easily applied to the decoding procedure of STBC-OFDM and SFBC-OFDM. To obtain independent detections for each symbol without ICI, a similar assumption needs to be made in order to make the non-diagonal elements of matrix Ψ zeros. The assumptions for STBC-OFDM and SFBC-OFDM are different due to their different mapping manners:

- **STBC-OFDM:** Channel impulse responses are assumed to be constant within a block period of two OFDM symbol periods for Alamouti codes or eight OFDM symbol periods for G_4 codes.

- **SFBC-OFDM:** Channel impulse responses are assumed to remain static during one OFDM symbol period and channel frequency response needs to be flat within



(a)STBC-OFDM



(b)SFBC-OFDM

Figure 5.1: Structure of STBC-OFDM and SFBC-OFDM systems

a block, which is two adjacent subcarriers for Alamouti codes or eight adjacent subcarriers for $\mathbf{G}_4$ codes.

It is obvious that STBC-OFDM is more sensitive to channel time variations between OFDM symbols but may have a good tolerance to frequency response variations between subcarriers, while SFBC-OFDM is vulnerable by both time variations between OFDM samples and frequency response fluctuations between subcarriers. The performance of Alamouti coded STBC-OFDM and SFBC-OFDM will be compared in various channel scenarios in the following paragraphs.

Suppose the number of OFDM subcarriers is N_s. Firstly, the performance of STBC-OFDM over frequency selective fading channels that remain static during one OFDM symbol period is investigated. According to Fig. 5.1(a), $2N_s$ symbols are transmitted on N_s subcarriers in two successive OFDM symbols. The frequency-domain received signals on the k-th subcarrier ($k = 1, 2...N_s$) are

$$\begin{aligned} R_1(k) &= H_{11}(k)S_{2k-1} + H_{21}(k)S_{2k} + N_1(k) \\ R_2(k) &= -H_{12}(k)S^*_{2k} + H_{22}(k)S^*_{2k-1} + N_2(k) \end{aligned} \tag{5.1}$$

where S_{2k-1} and S_{2k} are symbols transmitted on the k-th ($k = 1, 2...N_s$) subcarrier; $H_{it}(k)$ are frequency responses of the k-th ($k = 1, 2...N_s$) sub-channel during the t-th ($t = 1, 2$) OFDM symbol period from the i-th ($i = 1, 2$) transmit antenna to the receive antenna; $R_t(k)$ and $N_t(k)$ are frequency-domain received signals and receiver noise on the k-th ($k = 1, 2...N_s$) sub-channel during the t-th ($t = 1, 2$) OFDM symbol period. The equivalent vector-matrix form can be expressed as

$$\mathbf{R} = \mathbf{HS} + \mathbf{N} \tag{5.2}$$

where $\mathbf{R} = [R_1(k)\ R_2^*(k)]^{\mathrm{T}}$, $\mathbf{S} = [S_{2k-1}\ S_{2k}]^{\mathrm{T}}$ and $\mathbf{N} = [N_1(k)\ N_2^*(k)]^{\mathrm{T}}$. The channel matrix $\mathbf{H}$ is

$$\mathbf{H} = \begin{bmatrix} H_{11}(k) & H_{21}(k) \\ H^*_{22}(k) & -H^*_{12}(k) \end{bmatrix} \tag{5.3}$$

The output of the MF decoder can be obtained by

$$\tilde{\mathbf{S}} = \mathbf{\Psi}\mathbf{S} + \mathbf{H}^{\mathrm{H}}\mathbf{N} \tag{5.4}$$

where the matrix $\mathbf{\Psi}$ is

$$\mathbf{\Psi} = \mathbf{H}^{\mathrm{H}}\mathbf{H} = \begin{bmatrix} |H_{11}(k)|^2 + |H_{22}(k)|^2 & H_{11}^*(k)H_{21}(k) - H_{22}(k)H_{12}^*(k) \\ H_{21}^*(k)H_{11}(k) - H_{12}(k)H_{22}^*(k) & |H_{21}(k)|^2 + |H_{12}(k)|^2 \end{bmatrix} \tag{5.5}$$

Finally, a LS detection approach is implemented:

$$\hat{S}_i = \arg\{\min_{S_m \in M} |\tilde{S}_i - \psi_{ii} \cdot S_m|^2\} \quad (i = 1, 2) \tag{5.6}$$

where $\tilde{S}_i$ are the i-th ($i = 1, 2...4$) elements of $\tilde{\mathbf{S}}$ in Eq. 5.4; ψ_{ii} are the i-th ($i = 1, 2...4$) diagonal elements of Ψ and S_m are constellation values. Under the assumption of $H_{11}(k) = H_{12}(k)$ and $H_{21}(k) = H_{22}(k)$, the non-diagonal elements of $\mathbf{\Psi}$ add up zeros, and hence no ISI exists in the decoding process. In addition, spatial diversity is obtained given the diagonal elements of Ψ. Note that since symbols in one block are transmitted over one subcarrier from two transmit antennas, the variations in the channel frequency response will not affect the decoding process of STBC-OFDM.

Fig. 5.2 gives the BER performance of Alamouti coded STBC-OFDM over frequency selective fading channels that are varying from symbol to symbol. The number of subcarriers is 256 and the channel order is 3. The channel is generated by the time-varying multipath channel model described in Chapter 2. The channel impulse response is varying from one OFDM symbol to another at a fading rate of $f_d T_s$, where f_d is the Doppler frequency and T_s is the OFDM symbol period. It is clear from the results that diversity decreases due to ISI caused by channel time variations between two symbols. An irreducible error floor occurs at high E_b/N_0 regions and tends to get higher in a faster time-varying channel.

Then, the performance of SFBC-OFDM over frequency selective fading channels that are varying between symbols is considered. Fig. 5.1(b) shows that in SFBC-OFDM, N_s

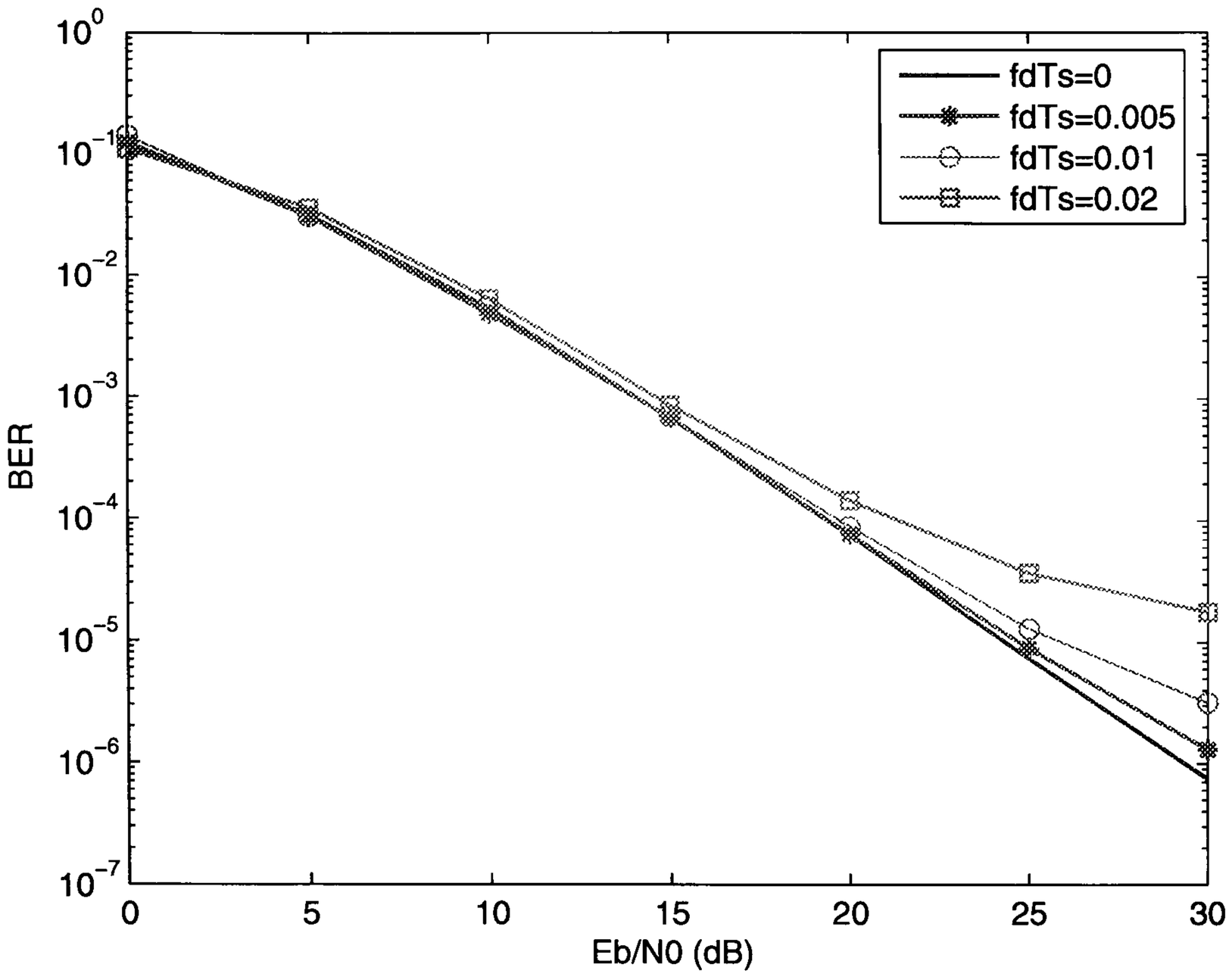


Figure 5.2: BER performance of Alamouti coded STBC-OFDM over frequency selective fading channels varying between two OFDM symbols

STBC coded symbols are mapped onto N_s subcarriers within one OFDM symbol. The received signal is given by

$$\begin{aligned} R_{2k-1} &= H_{1(2k-1)}S_{2k-1} + H_{2(2k-1)}S_{2k} + N_{2k-1} \\ R_{2k} &= -H_{1(2k)}S^*_{2k} + H_{2(2k)}S^*_{2k-1} + N_{2k} \end{aligned} \tag{5.7}$$

where the subscripts '$2k$-1' and '$2k$' represent the $(2k$-1)-th and $(2k)$-th ($k = 1, 2...N_s/2$) subcarriers; $H_{i(2k-1)}$ and $H_{i(2k)}$ are channel frequency responses from the i-th (i=1,2) transmit antenna to the receive antenna; S_{2k-1} and S_{2k} are transmitted symbols; R_{2k-1}, R_{2k} and N_{2k-1}, N_{2k} are received symbols and receiver noise in the frequency domain.

Similarly, Eq. 5.7 can be written in a vector-matrix form as in Eq. 5.2. Omitting the derivative process, the matrix $\mathbf{\Psi}$ can be obtained:

$$\mathbf{\Psi} = \begin{bmatrix} |H_{1(2k-1)}|^2 + |H_{2(2k)}|^2 & H^*_{1(2k-1)}H_{2(2k-1)} - H_{2(2k)}H^*_{1(2k)} \\ H^*_{2(2k-1)}H_{1(2k-1)} - H_{1(2k)}H^*_{2(2k)} & |H_{2(2k-1)}|^2 + |H_{1(2k)}|^2 \end{bmatrix} \tag{5.8}$$

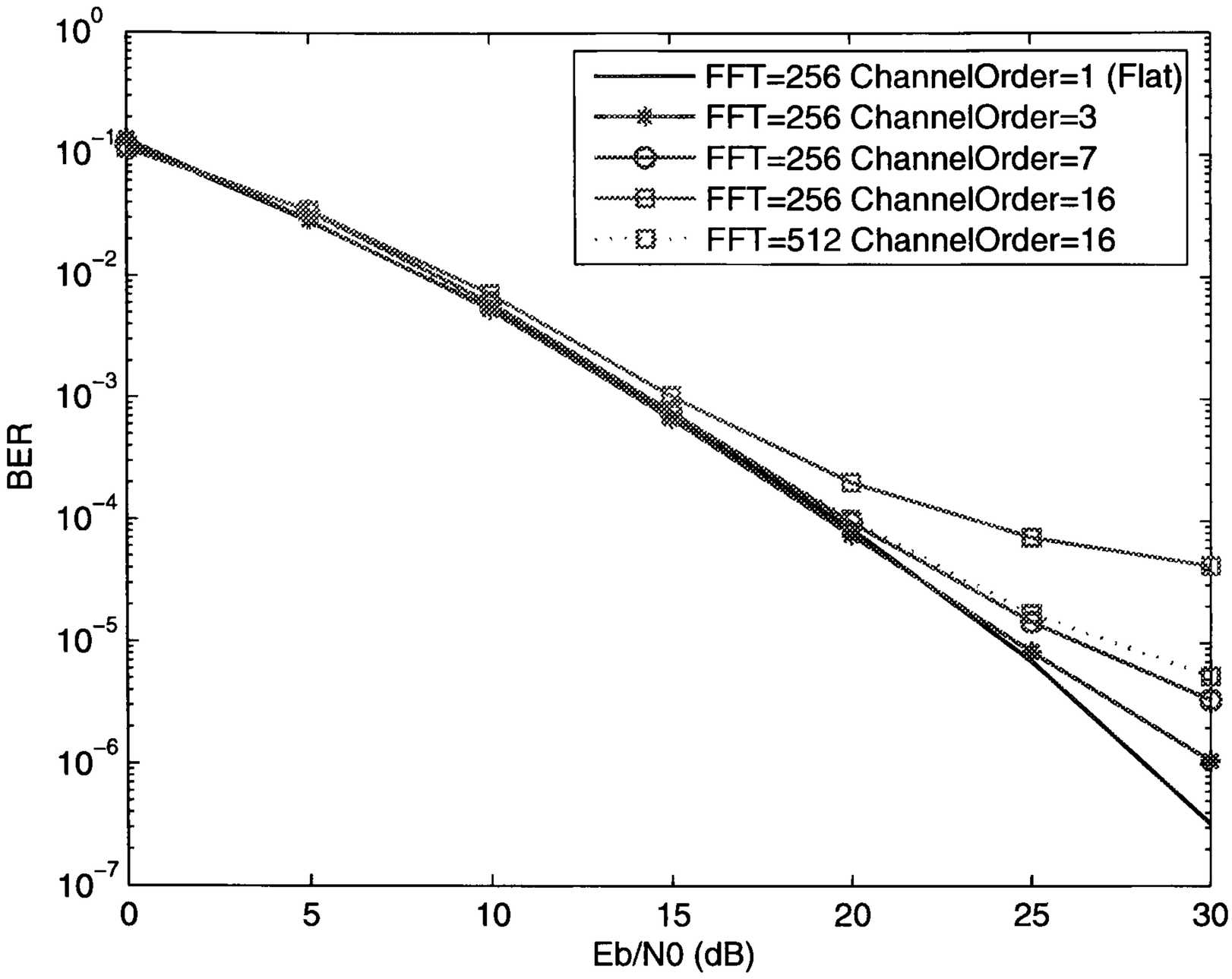


Figure 5.3: BER performance of Alamouti coded SFBC-OFDM over quasi-static (within 1 OFDM symbol) frequency selective fading channels

If channel frequency response is approximately constant within a block, which means $H_{1(2k-1)} \approx H_{1(2k)}$ and $H_{2(2k-1)} \approx H_{2(2k)}$, ICI can be eliminated without reducing the diversity order.

Fig. 5.3 shows the BER performance of Alamouti coded SFBC-OFDM over frequency selective fading channels that are static during one OFDM symbol. Channels are created by using the tapped delay-line model described in Chapter 2. The number of subcarriers is also 256. The performance of the system in flat fading channels ('ChannelOrder=1') and multipath channels ('ChannelOrder> 1') are compared. It is observed that frequency response variations between subcarriers give rise to ICI and introduce an irreducible error floor at high E_b/N_0 regions. The error floor appears higher if channel frequency response varies more rapidly between subcarriers. That is, for a fixed bandwidth, the channel delay spread increases or the number of subcarriers decreases. For the latter case, it is shown

that with the same channel order ('ChannelOrder=16'), the system with 512 subcarriers presents lower error floor than the one with 256 subcarriers. The time variations between OFDM symbols does not affect the decoding process of SFBC-OFDM.

The performance of STBC-OFDM and SFBC-OFDM are also compared in frequency selective fading channels that are varying sample by sample, as given by Eq. 2.14. The problem is that in this case, channel frequency responses are also changing with time (sample by sample), which makes it hard to find the matrix $\mathbf{H}$ in the MF decoding process. A sub-optimal approximation is that, for each channel tap, take the average value over all time slots (samples) in an OFDM symbol. In this way, a 'quasi-static' channel whose channel taps are presented by these average values can be generated. The decoding can be processed based on its frequency-domain representation.

The time-domain received signals from the i-th ($i = 1, 2$) transmit antenna are

$$\mathbf{r}^i = \begin{bmatrix} r^i_{N_1+1} \\ \vdots \\ r^i_{N_s} \\ r^i_1 \\ \vdots \\ r^i_{N_s} \end{bmatrix} = \begin{bmatrix} h^i_{11} & & & & & \\ h^i_{22} & h^i_{21} & & & 0 & \\ \vdots & & \ddots & & & \\ h^i_{LL} & \cdots & h^i_{L2} & h^i_{L1} & & \\ & \ddots & \cdots & \cdots & \ddots & \\ & 0 & h^i_{N_2L} & \cdots & h^i_{N_22} & h^i_{N_21} \end{bmatrix} \cdot \begin{bmatrix} s^i_{N_1+1} \\ \vdots \\ s^i_{N_s} \\ s^i_1 \\ \vdots \\ s^i_{N_s} \end{bmatrix} \tag{5.9}$$

where s^i_t and r^i_t are respectively the t-th sample ($t = N_s - N_{cp} + 1 \ldots N_s, 1 \ldots N_s$) of the $(N_s + N_{cp})$ transmitted samples from the CP extended signal and the corresponding received samples (N_s are the number of subcarriers and N_{cp} are the length of CP); h^i_{tl} represent the l-th ($l = 1, 2 \ldots L$) channel tap at the t-th sampling time, which compose an $(N_s+N_{cp})\times(N_s+N_{cp})$ channel matrix. For notational simplicity, denote $N_1 = (N_s - N_{cp})$ and $N_2 = (N_s + N_{cp})$. The received signal can be expressed as

$$\mathbf{r} = \sum_{i=1}^{2} \mathbf{r}^i + \mathbf{n} \tag{5.10}$$

where $\mathbf{r}$ and $\mathbf{n}$ are vectors containing $(N_s + N_{cp})$ received samples and AWGN noise samples.

The channel impulse response after averaging over all samples in a OFDM symbol from the i-th ($i = 1, 2$) transmit antenna is

$$\bar{\mathbf{h}}^i = \begin{bmatrix} \bar{h}_L^i & \cdots & \bar{h}_2^i & \bar{h}_1^i \end{bmatrix} \tag{5.11}$$

Therefore, Eq. 5.9 is equivalent to

$$\bar{\mathbf{r}}^i = \begin{bmatrix} r_{N_1+1}^i \\ \vdots \\ r_{N_s}^i \\ r_1^i \\ \vdots \\ r_{N_s}^i \end{bmatrix} = \begin{bmatrix} \bar{h}_1^i & & & & & \\ \bar{h}_2^i & \bar{h}_1^i & & & 0 & \\ \vdots & & \ddots & & & \\ \bar{h}_L^i & \cdots & \bar{h}_2^i & \bar{h}_1^i & & \\ & \ddots & \cdots & \cdots & \ddots & \\ & 0 & \bar{h}_L^i & \cdots & \bar{h}_2^i & \bar{h}_1^i \end{bmatrix} \cdot \begin{bmatrix} s_{N_1+1}^i \\ \vdots \\ s_{N_s}^i \\ s_1^i \\ \vdots \\ s_{N_s}^i \end{bmatrix} \tag{5.12}$$

Eq. 5.12 is exactly the cyclic convolution of the averaged channel impulse response $\bar{\mathbf{h}}^i$ ($i = 1, 2$) and the transmitted samples, therefore,

$$\bar{\mathbf{r}}^i = \bar{\mathbf{h}}^i * \mathbf{s}^i \tag{5.13}$$

Therefore, the frequency-domain received signal can be written in the same form as in Eq. 5.2, where the frequency responses of sub-channels are obtained from the N_s point FFT of $\bar{\mathbf{h}}^i$ ($i = 1, 2$).

Note that although averaging the values of channel taps over time greatly reduces the decoding complexity, sample-by-sample channel time variations do introduce ICI in the decoding process. The coexistence of time variations and frequency response fluctuations may make the BER performance of SFBC-OFDM or STBC-OFDM even worse.

Fig. 5.4 and Fig. 5.5 shows the BER performance of STBC-OFDM and SFBC-OFDM over sample-by-sample time variant and frequency selective fading channels. The fading rate of the channel is $f_d t_s$, where f_d is the Doppler frequency and t_s is the OFDM sampling period. Again, the number of subcarriers is 256. The performance under different channel orders ('ChannelOrder=3' and 'ChannelOrder=16') and fading rates ('$f_d t_s = 0.01\%$' and '$f_d t_s = 0.02\%$') are compared. It is shown that under the same channel conditions, SFBC-OFDM performs better than STBC-OFDM in general. The error floor in STBC-OFDM

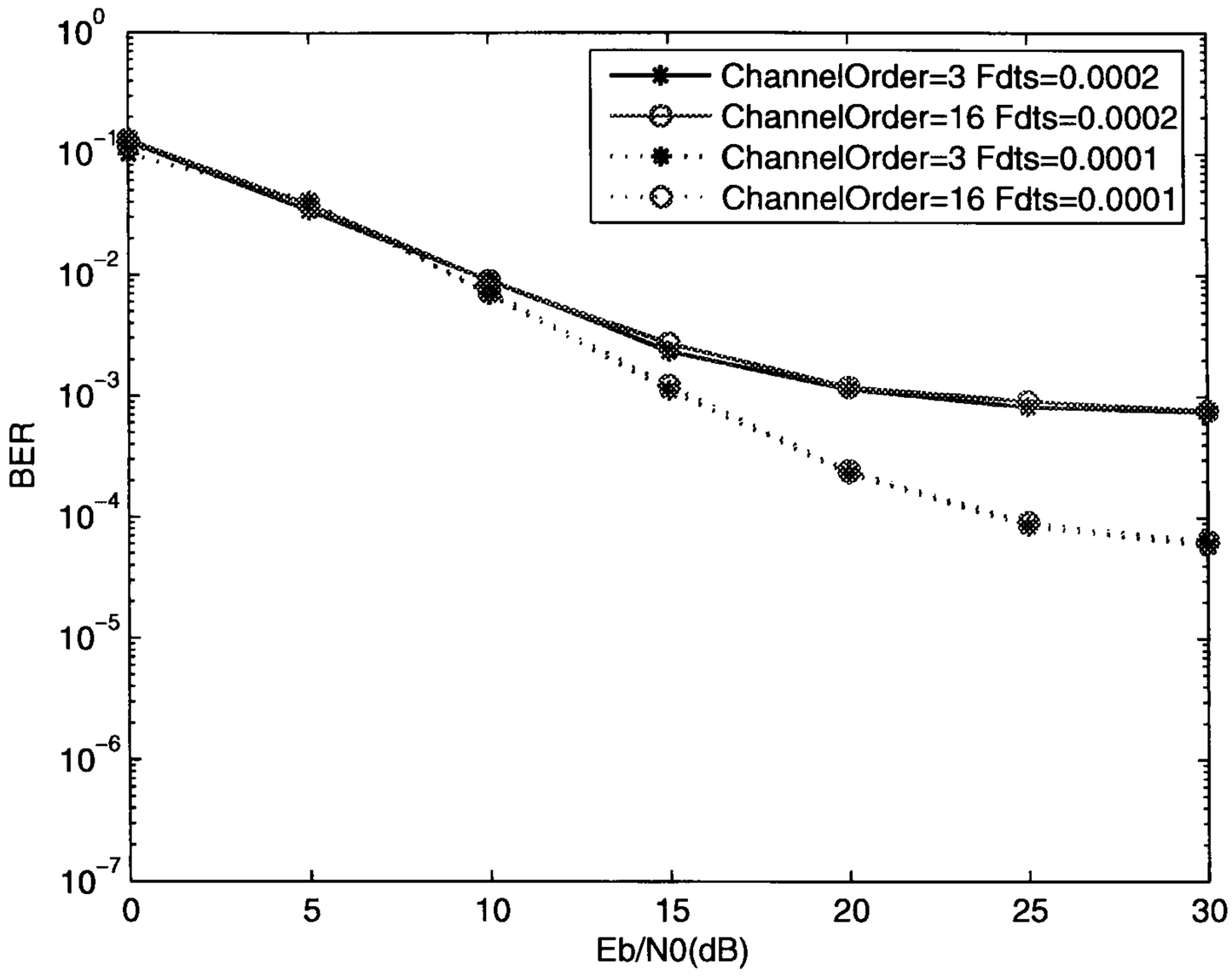


Figure 5.4: BER performance of Alamouti coded STBC-OFDM over sample-by-sample time-varying and frequency selective fading channels

only results from the time variation between OFDM samples, while both frequency response and time variations between samples affect the performance of SFBC-OFDM. It is obvious that the error floors in both systems will become higher when the channels are subject to a higher fading rate or longer delay spread.

5.2 ZF Decoding for Alamouti SFBC-OFDM

This section will discuss the interference cancelation schemes in Alamouti ($\mathbf{G}_2$) and $\mathbf{G}_4$ SFBC-OFDM based on the MF decoding. Assume that the channel impulse response remains static during one OFDM symbol period, so that the ICI only results from frequency response fluctuations between subcarriers within a block. Since the decoding is performed within one block, for simplicity while without losing generality, our consideration is restricted in the first block.

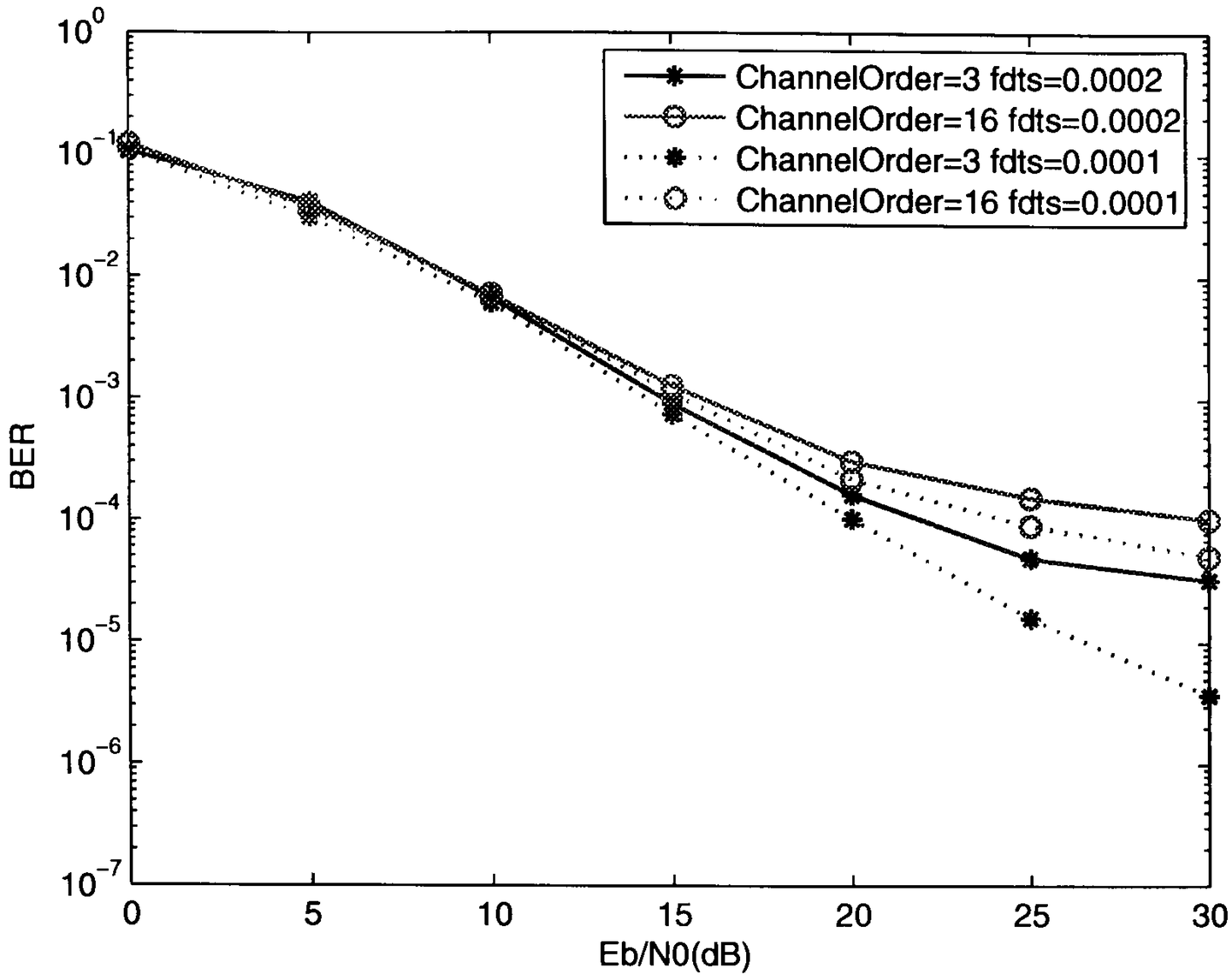


Figure 5.5: BER performance of Alamouti coded SFBC-OFDM over sample-by-sample time-varying and frequency selective fading channels

In this section, a ZF decoder [9] that is able to eliminate ICI in the Alamouti SFBC-OFDM system will be introduced. From Eq. 5.7, the frequency-domain received signal in the vector-matrix form can be written as

$$\begin{bmatrix} R_1 \\ R_2^* \end{bmatrix} = \mathbf{H} \begin{bmatrix} S_1 \\ S_2 \end{bmatrix} + \begin{bmatrix} N_1 \\ N_2^* \end{bmatrix} \tag{5.14}$$

where

$$\mathbf{H} = \begin{bmatrix} H_{11} & H_{21} \\ H_{22}^* & -H_{12}^* \end{bmatrix} \tag{5.15}$$

After the MF decoding, detections are taken separately for different symbols:

$$\begin{bmatrix} \hat{S}_1 \\ \hat{S}_2 \end{bmatrix} = \mathbf{\Psi} \begin{bmatrix} S_1 \\ S_2 \end{bmatrix} + \begin{bmatrix} \tilde{N}_1 \\ \tilde{N}_2 \end{bmatrix} \tag{5.16}$$

where $\tilde{N}_1$ and $\tilde{N}_2$ are still Gaussian values: $\tilde{N}_1 = H_{11}^* N_1 + H_{22} N_2^*$, $\tilde{N}_2 = H_{21}^* N_1 - H_{12} N_2^*$, and

$$\mathbf{\Psi} = \mathbf{H}^{\mathrm{H}}\mathbf{H} = \begin{bmatrix} |H_{11}|^2 + |H_{22}|^2 & H_{11}^* H_{21} - H_{22} H_{12}^* \\ H_{21}^* H_{11} - H_{12} H_{22}^* & |H_{21}|^2 + |H_{12}|^2 \end{bmatrix} \tag{5.17}$$

It is clear from the diagonal elements of Ψ that the MF decoding is able to achieve the maximum spatial diversity. However, in case $H_{11} \neq H_{12}$ and $H_{21} \neq H_{22}$, ICI will be introduced in the decoding process, as shown by the non-diagonal elements of $\mathbf{\Psi}$.

Intuitively, eliminating ICI equals to find a diagonal matrix $\mathbf{\Psi}$ where non-diagonal elements are zeros, and diagonal elements are maximized to reach the most of the available diversity, which can be simply represented by

$$\mathbf{\Psi} = \widetilde{\mathbf{H}}\mathbf{H} = \begin{bmatrix} d_1 & 0 \\ 0 & d_2 \end{bmatrix} \tag{5.18}$$

The following shows the derivation of $\widetilde{\mathbf{H}}$ in the ZF decoding. Firstly, denote $\widetilde{\mathbf{H}}$ as

$$\widetilde{\mathbf{H}} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \tag{5.19}$$

The equations for the interference cancelation are

$$a_{11}H_{21} - a_{12}H_{12}^* = 0, \quad a_{21}H_{11} + a_{22}H_{22}^* = 0 \tag{5.20}$$

and the diversity gain is expressed as

$$d_1 = a_{11}H_{11} + a_{12}H_{22}^*, \quad d_2 = a_{21}H_{21} - a_{22}H_{12}^* \tag{5.21}$$

By using Eq. 5.20, a_{12} and a_{21} can be expressed in terms of a_{11} and a_{22}:

$$a_{12} = a_{11}H_{21}/H_{12}^*, \quad a_{21} = -a_{22}H_{22}^*/H_{11} \tag{5.22}$$

Substituting Eq. 5.22 into Eq. 5.21, the diversity equations can be written as

$$d_1 = a_{11}(H_{11} + H_{22}^*H_{21}/H_{12}^*), \quad d_2 = a_{22}(-H_{12}^* - H_{21}H_{22}^*/H_{11}) \tag{5.23}$$

Let $a_{11} = H_{11}^*$ and $a_{22} = -H_{12}$, to reach the maximum diversity from the two signals transmitted on the first antenna. Then substitute them into the diversity equations. The diversity equation can be expressed as:

$$\begin{aligned} d_1 &= |H_{11}|^2 + |H_{22}|^2 \frac{H_{11}^*H_{21}}{H_{12}^*H_{22}} \\ d_2 &= |H_{12}|^2 + |H_{21}|^2 \frac{H_{22}^*H_{12}}{H_{21}^*H_{11}} \end{aligned} \tag{5.24}$$

where $\frac{H_{11}^* H_{21}}{H_{12}^* H_{22}}$ and $\frac{H_{22}^* H_{12}}{H_{21}^* H_{11}}$ are the amplitude and phase differences between channel frequency responses of adjacent subcarriers. In the extreme cases, $\frac{H_{11}^* H_{21}}{H_{12}^* H_{22}}$ or $\frac{H_{22}^* H_{12}}{H_{21}^* H_{11}}$ can either be zero or infinite, which means d_1 or d_2 will be dominated either by $|H_{11}|^2$ ($|H_{12}|^2$) or $|H_{22}|^2$ ($|H_{21}|^2$). This indicates that the diversity gain can decrease to 'slope 1' in the worst cases. In other words, the advantage of interference cancelation by ZF decoding is obtained in the sacrifice of the diversity.

Finally, $\widetilde{\mathbf{H}}$ takes place of $\mathbf{H}^{\mathrm{H}}$ in the MF decoding:

$$\widetilde{\mathbf{H}} = \begin{bmatrix} H_{11}^* & H_{11}^* H_{21}/H_{12}^* \\ H_{12} H_{22}^*/H_{11} & -H_{12} \end{bmatrix} \tag{5.25}$$

and the matrix $\mathbf{\Psi}$ becomes

$$\mathbf{\Psi} = \widetilde{\mathbf{H}}\mathbf{H} = \begin{bmatrix} |H_{11}|^2 + |H_{22}|^2 \frac{H_{11}^* H_{21}}{H_{12}^* H_{22}} & 0 \\ 0 & |H_{12}|^2 + |H_{21}|^2 \frac{H_{22}^* H_{12}}{H_{21}^* H_{11}} \end{bmatrix} \tag{5.26}$$

The 'ZF' decoder is named since it eliminates ICI by forcing the non-diagonal elements of $\mathbf{\Psi}$ to zeros. It is clear from Eq. 5.25 and Eq. 5.26 that the complexity of ZF decoder is higher than that of the conventional MF decoder. However, given the advantage of interference cancelation, the extra computations in the ZF decoder is tolerable, compared with the full ML search over all two symbols in the $\mathbf{G}_2$ case, whose complexity is $|M|^2$ ($|M|$ is the signalling constellation size, e.g., $|M| = 4$ for QPSK).

The performance of the ZF decoder, compared with that of the MF decoder, in Alamouti coded SFBC-OFDM over frequency selective fading channels is demonstrated in Fig. 5.6. In the 'ChannelOrder=16' case, the ZF decoder is able to eliminate the error floor caused by the frequency response variations between adjacent subcarriers. However, this benefit is gained at the cost of a decrease in the spatial diversity gain, which can be fully achieved by the MF decoding in a frequency flat case ('ChannelOrder=1').

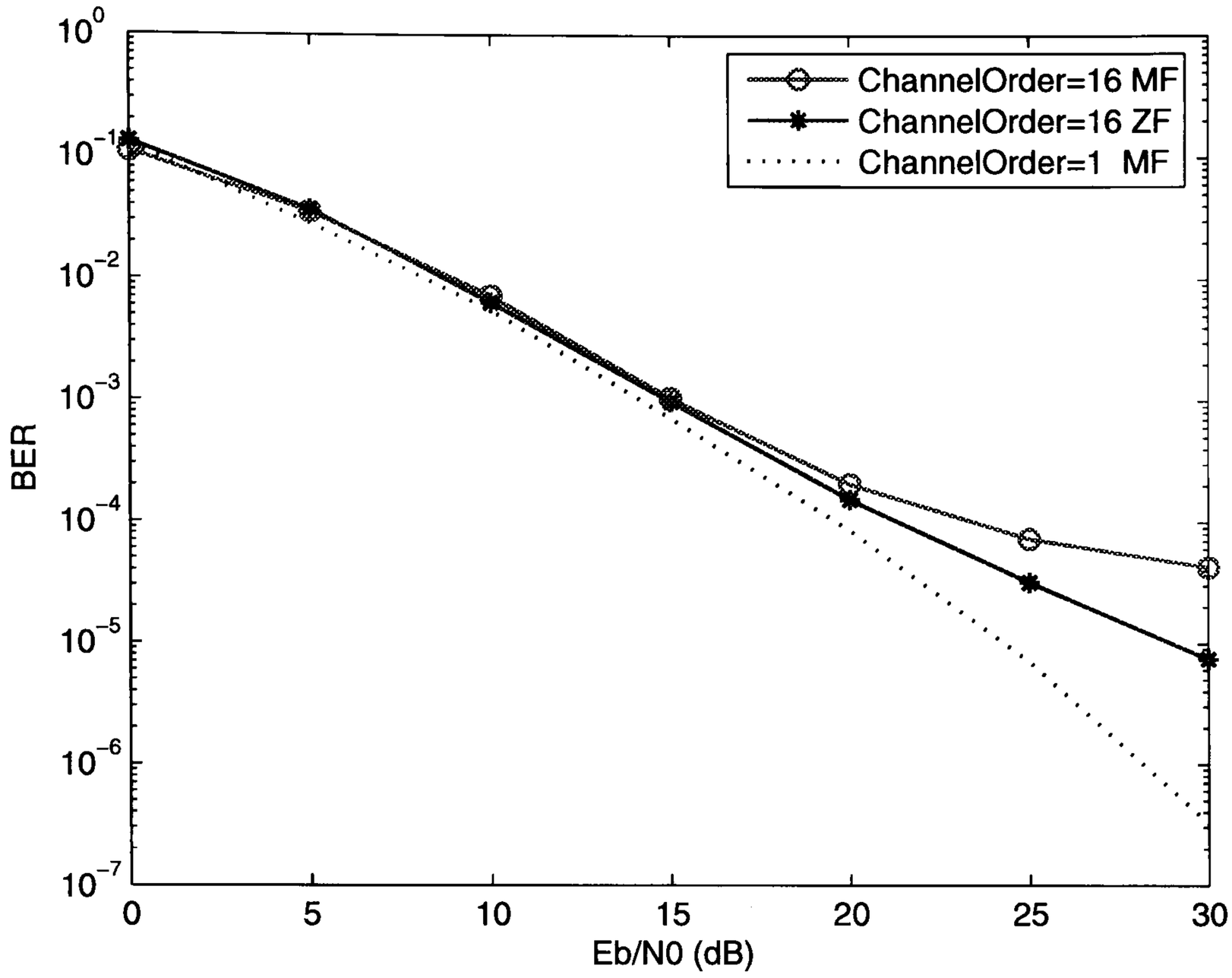


Figure 5.6: BER performance of MF and ZF decoding in Alamouti coded SFBC-OFDM over frequency selective fading channels

5.3 MF-based PIC Detection for $\mathbf{G}_4$ SFBC-OFDM

It is clear that $\mathbf{G}_4$ SFBC-OFDM presents more sensitivity to variations of channel frequency responses than $\mathbf{G}_2$ SFBC-OFDM, since the block length is four times as long. The ZF decoding has been shown to be able to eliminate the ICI effect in $\mathbf{G}_2$ SFBC-OFDM. It is straightforward to find a ZF algorithm for the $\mathbf{G}_4$ system.

Again, take the example of the first transmission block for notational simplicity. In the $\mathbf{G}_4$ case, one block contains eight adjacent subcarriers. The frequency-domain received signals and its conjugates in a vector-matrix form can be written as Eq. 5.2. Here, $\mathbf{R} = [R_1\ R_2\ R_3\ R_4\ R_5^*\ R_6^*\ R_7^*\ R_8^*]^{\mathrm{T}}$, $\mathbf{S} = [S_1\ S_2\ S_3\ S_4\ S_5\ S_6\ S_7\ S_8]^{\mathrm{T}}$ are received and transmitted signals and $\mathbf{N} = [N_1\ N_2\ N_3\ N_4\ N_5^*\ N_6^*\ N_7^*\ N_8^*]^{\mathrm{T}}$ are AWGN samples in the frequency

domain. The 8×4 channel matrix $\mathbf{H}$ is expressed as

$$\mathbf{H} = \begin{bmatrix} \mathbf{H}_{12}^{1} & \mathbf{H}_{34}^{1} \\ \acute{\mathbf{H}}_{34}^{3} & -\acute{\mathbf{H}}_{12}^{3} \\ \mathbf{H}_{12}^{5\,*} & \mathbf{H}_{34}^{5\,*} \\ \acute{\mathbf{H}}_{34}^{7\,*} & -\acute{\mathbf{H}}_{12}^{7\,*} \end{bmatrix} \tag{5.27}$$

where $\mathbf{H}_{i_1 i_2}^{k}$ and $\acute{\mathbf{H}}_{i_1 i_2}^{k}$ are matrixes representing

$$\mathbf{H}_{i_1 i_2}^{k} = \begin{bmatrix} H_{i_1 k} & H_{i_2 k} \\ H_{i_2 (k+1)} & -H_{i_1 (k+1)} \end{bmatrix}, \quad \acute{\mathbf{H}}_{i_1 i_2}^{k} = \begin{bmatrix} H_{i_1 k} & -H_{i_2 k} \\ H_{i_2 (k+1)} & H_{i_1 (k+1)} \end{bmatrix} \tag{5.28}$$

$H_{i_1 k}$, $H_{i_1 (k+1)}$, $H_{i_2 k}$ and $H_{i_2 (k+1)}$ are channel frequency responses on the k-th or $(k+1)$-th ($k = 1, 3, 5, 7$) subcarriers from the i_1-th or i_2-th ($i_1, i_2 = 1, 2...4$) transmit antennas.

The ZF decoding is to find a matrix $\widetilde{\mathbf{H}}$, which can force the non-diagonal elements of $\widetilde{\mathbf{H}}\mathbf{H}$ zeros. In the $\mathbf{G}_4$ case, $\widetilde{\mathbf{H}}$ is a 4×8 matrix, which means 32 unknown parameters need to be obtained from 12 interference cancelation equations and 4 diversity maximization polynomials. The computational complexity of ZF algorithm for $\mathbf{G}_4$ SFBC-OFDM is too high.

As is well known, PIC is an effective yet simple way for multi-user detection in CDMA [43] systems. In SFBC-OFDM, symbols in one block are detected in parallel in the MF decoding, which makes PIC approach suited to mitigate the ICI in parallel in such systems. This approach is called 'MF-based PIC detection'.

First of all, the received signals are decoded by the conventional MF decoder, as expressed in Eq. 5.4-5.5. Here, the channel transmission matrix $\mathbf{H}$ is given by Eq. 5.27-5.28 and the output vector $\tilde{\mathbf{S}}$ is composed of $\tilde{S}_i$ ($i = 1, 2...4$). PIC works in an iterative manner by feeding back the refreshed data from the previous iteration to the current one, while the PIC algorithm remains the same. The refreshed data from each iteration can be 'hard' or 'soft', depending on whether a LS decision is applied or not. They are called 'hard/soft PIC' respectively.

The 'hard PIC' is processed in the following steps.

1. Define a matrix $\mathbf{\Psi}_D$ with only the diagonal elements of $\mathbf{\Psi}$ on its diagonal positions. Hence, a matrix $\mathbf{\Psi}_{ND}$ containing the non-diagonal elements of $\mathbf{\Psi}$ can be obtained by

$$\mathbf{\Psi}_{ND} = \mathbf{\Psi} - \mathbf{\Psi}_D \tag{5.29}$$

2. Carry out the LS decision on the MF decoded symbols $\tilde{S}_i$ $(i = 1, 2...4)$ according to Eq. 5.6. The LS detected symbols are denoted as $\hat{S}_i$ $(i = 1, 2...4)$. Set the iteration number $iter = 0$. The initial values $\hat{\mathbf{S}}^{(0)}$ for the iteration are obtained by

$$\hat{\mathbf{S}}^{(0)} = [\hat{S}_1\ \hat{S}_2\ \hat{S}_3\ \hat{S}_4]^{\mathrm{T}} \tag{5.30}$$

   Repeat for $iter = 1, 2...I$ in the following two steps.

3. For the $iter$-th iteration ($iter > 0$), subtract interference from the initial value using the estimated data from the ($iter$-1)-th iteration.

$$\tilde{\mathbf{S}}^{(iter)} = \tilde{\mathbf{S}} - \mathbf{\Psi}_{ND}\hat{\mathbf{S}}^{(iter-1)} \tag{5.31}$$

4. Again, apply the LS detection to the interference-canceled data from Eq. 5.31 for the current iteration.

$$\hat{S}_i^{(iter)} = \arg\{\min_{S_m \in M} |\tilde{S}_i^{(iter)} - \psi_{ii} \cdot S_m|^2\} \quad (i = 1, 2, 3, 4) \tag{5.32}$$

   where ψ_{ii} $(i = 1, 2...4)$ are diagonal elements of the matrix $\mathbf{\Psi}$; M denotes the signalling constellation and $\tilde{S}_i^{(iter)}$ is the i-th $(i = 1, 2...4)$ element of the vector $\tilde{\mathbf{S}}^{(iter)}$ in Eq. 5.31. If the iteration ends ($iter = I$), Eq. 5.32 gives the output, which will then be de-modulated (effectively by separating the real and imaginary parts in the case of QPSK) and interleavered before fed to a soft decision decoder for the outer FEC code. If not ($iter < I$), let

$$\hat{\mathbf{S}}^{(iter)} = [\hat{S}_1^{(iter)}\ \hat{S}_2^{(iter)}\ \hat{S}_3^{(iter)}\ \hat{S}_4^{(iter)}]^{\mathrm{T}} \tag{5.33}$$

   and substitute $\hat{\mathbf{S}}^{(iter)}$ into Eq. 5.31 for the next iteration.

It is Eq. 5.31 that counts for the actual interference subtraction. As iterations proceed, the ICI in general decreases, and hence the symbol estimation in Eq. 5.32 becomes more and more accurate. Once the ICI related errors in the non-diagonal positions of $\tilde{\mathbf{S}}$ have been eliminated, $\hat{\mathbf{S}}^{(iter)}$ becomes a pure diagonal matrix. The decoding process gives results equivalent to the MF decoding procedure which can provide the ML performance. This explains why PIC detection can potentially offer better performance than the conventional decoding in the SFBC-OFDM over a frequency selective channel.

However, there is a trade-off between the performance and the complexity. It is obvious that compared with MF decoding, the computational complexity increases. Nevertheless, the extra complexity is still moderate compared with true ML decoding for the frequency-variant channel, which would find the minimum square error over all four symbols in the $\mathbf{G}_4$ case, and hence have a complexity proportional to $|M|^4$.

Note that the initial pattern of the signal in Eq. 5.30 may contain errors since it is obtained by the conventional linear MF decoding. The interference subtraction in Eq. 5.31 is performed over all the symbols in one block together. This means that for certain patterns of data symbol errors or a particular noise vector, error propagation may occur. The error in one data symbol may introduce errors to other data symbols in the same block for the next iteration. It has been shown that this can cause the PIC detection to get 'stuck' in an oscillatory mode in some blocks. Hence an irreducible error floor may still remain in high E_b/N_0 regions of the BER curve, oscillating with the number of iterations, although usually at a lower level.

Another way to do PIC detection is to omit the LS decisions in the initial MF decoding and during the following iterations: the LS decision is only carried out once at the end. The performance is expected to be improved, since 'soft' symbols take into account both data information and information of ICI, instead of only the decided data symbols in 'hard PIC' iterations.

The 'soft PIC' is carried out in the following steps.

1. Obtain $\mathbf{\Psi}_{ND}$ according to 'step 1' in the 'hard PIC' approach.

2. Set the iteration number $iter = 0$. The initial values $\hat{\mathbf{S}}^{(0)}$ for the iteration are obtained directly from the output of the MF decoding $\tilde{S}_i$ $(i = 1, 2...4)$.

$$\hat{\mathbf{S}}^{(0)} = [\tilde{S}_1/\psi_{11} \quad \tilde{S}_2/\psi_{22} \quad \tilde{S}_3/\psi_{33} \quad \tilde{S}_4/\psi_{44}] \tag{5.34}$$

3. For the $iter$-th iteration ($iter > 0$), subtract interference from the initial value using the refreshed data from the ($iter$-1)-th iteration. Repeat 'step 3' for $iter = 1, 2...I$.

$$\tilde{\mathbf{S}}^{(iter)} = (\tilde{\mathbf{S}} - \mathbf{\Psi}_{ND}\hat{\mathbf{S}}^{(iter-1)})./\psi_{ii} \tag{5.35}$$

4. When $iter = I$, the decoding output is given by applying the LS detection to the data symbols from the ultimate iteration of Eq. 5.35.

$$\hat{S}_i^{(I)} = \arg\{\min_{S_m \in M} |\tilde{S}_i^{(I)} - S_m|^2\} \quad (i = 1, 2...4) \tag{5.36}$$

   Again, $\hat{S}_i^{(I)}$ will be de-modulated and may be de-interleavered before decoded by a soft-decision FEC decoder.

As discussed, an error floor oscillating with the number of iterations will occur in the BER performance of 'hard PIC' because of the non-optimum initial values from the LS detection in the conventional decoding. Although the LS detection is avoided in the 'soft PIC', the interference elements included in the initial values may also cause such a problem. Therefore, oscillating error floors will again exist at high E_b/N_0 regions of the BER performance of the 'soft PIC', although lower than that of the 'hard PIC'.

Fig. 5.7 gives the BER performance of MF decoding in $\mathbf{G}_4$ coded SFBC-OFDM over quasi-static frequency selective fading channels. The number of subcarriers is 256. The channel order is taken from the lowest '1' to the highest '16' for comparisons. The presence of error floors at high E_b/N_0 regions demonstrates that the diversity is reduced by

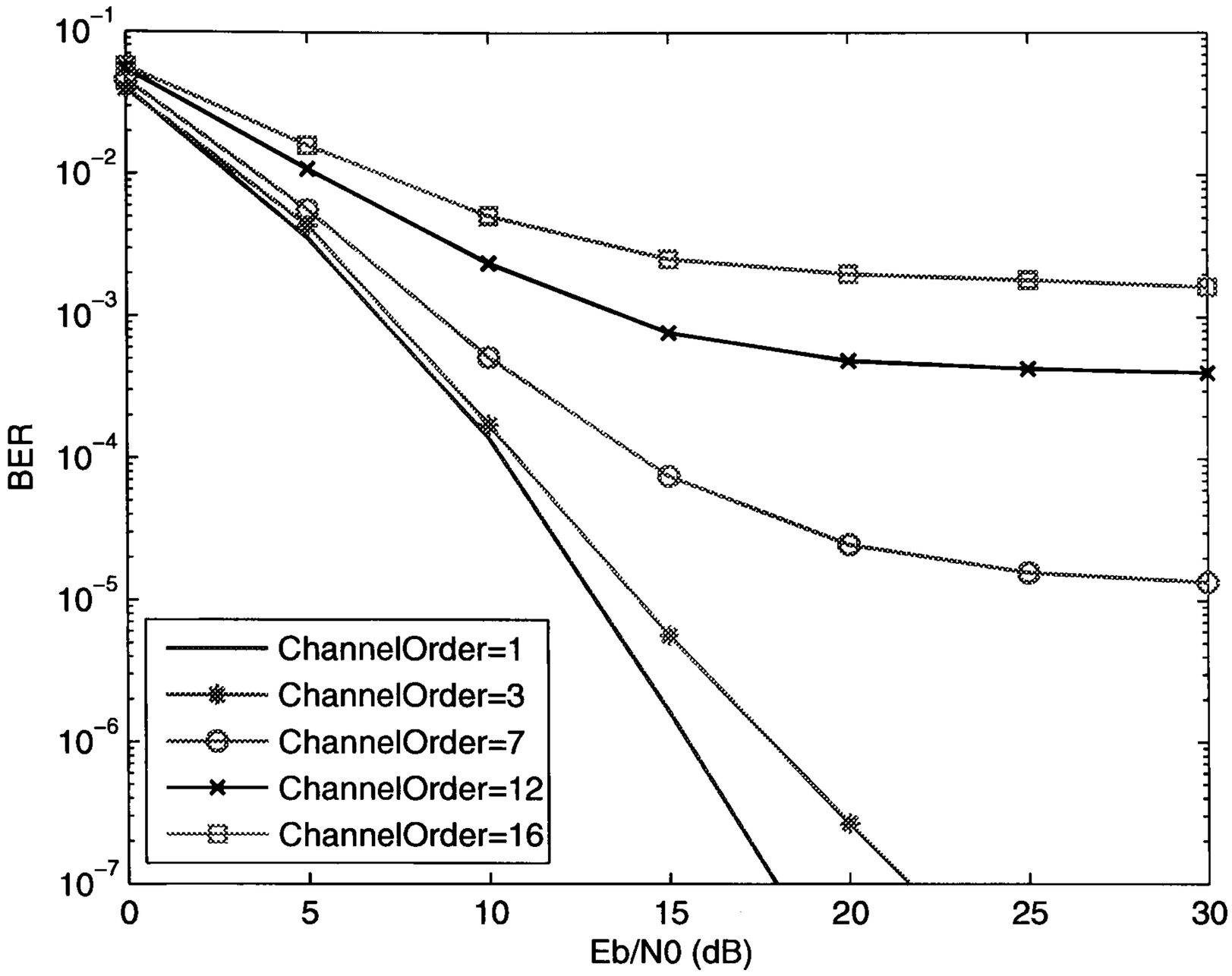


Figure 5.7: BER performance of MF decoding in $\mathbf{G}_4$ coded SFBC-OFDM over quasi-static frequency selective fading channels

the ICI due to frequency response variations between sub-channels. It is also seen that under the same channel environment, the degradation of the BER performance in the $\mathbf{G}_4$ case is much more than that in the $\mathbf{G}_2$ case (Fig. 5.3), although the maximum spatial diversity provided by $\mathbf{G}_4$ codes is theoretically twice as large. Therefore, $\mathbf{G}_4$ SFBC-OFDM is more sensitive to frequency response variations of the channel and requires a more powerful way to mitigate the effect of ICI.

Fig. 5.8 and Fig. 5.9 show the performance improvements obtained by using MF-based PIC approach. In all simulations, the channel order is 12. It is clear that in both 'hard PIC' and 'soft PIC', most gain comes from the first iteration ('I=1'): 6 dB for 'soft PIC' and 4 dB for 'hard PIC' at the BER of 10^{-3}, and the returns from increasing the number above 'I=1' are relatively small. As the iterations proceed, more gain is achieved. In 'soft PIC', the gain increase becomes negligible after four iterations ('I=4'), while

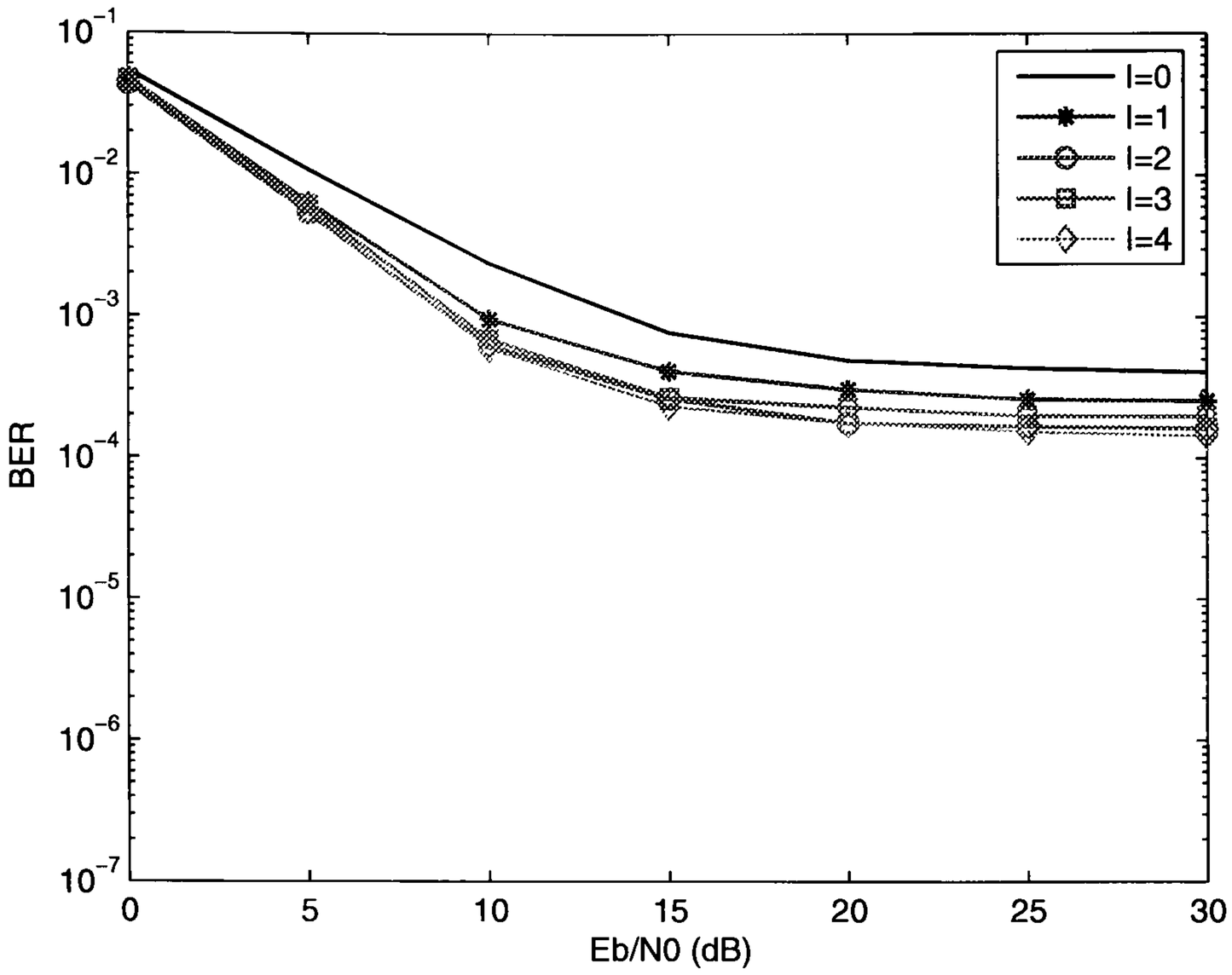


Figure 5.8: BER performance of MF-based PIC (hard) in $\mathbf{G}_4$ coded SFBC-OFDM over quasi-static frequency selective fading channels

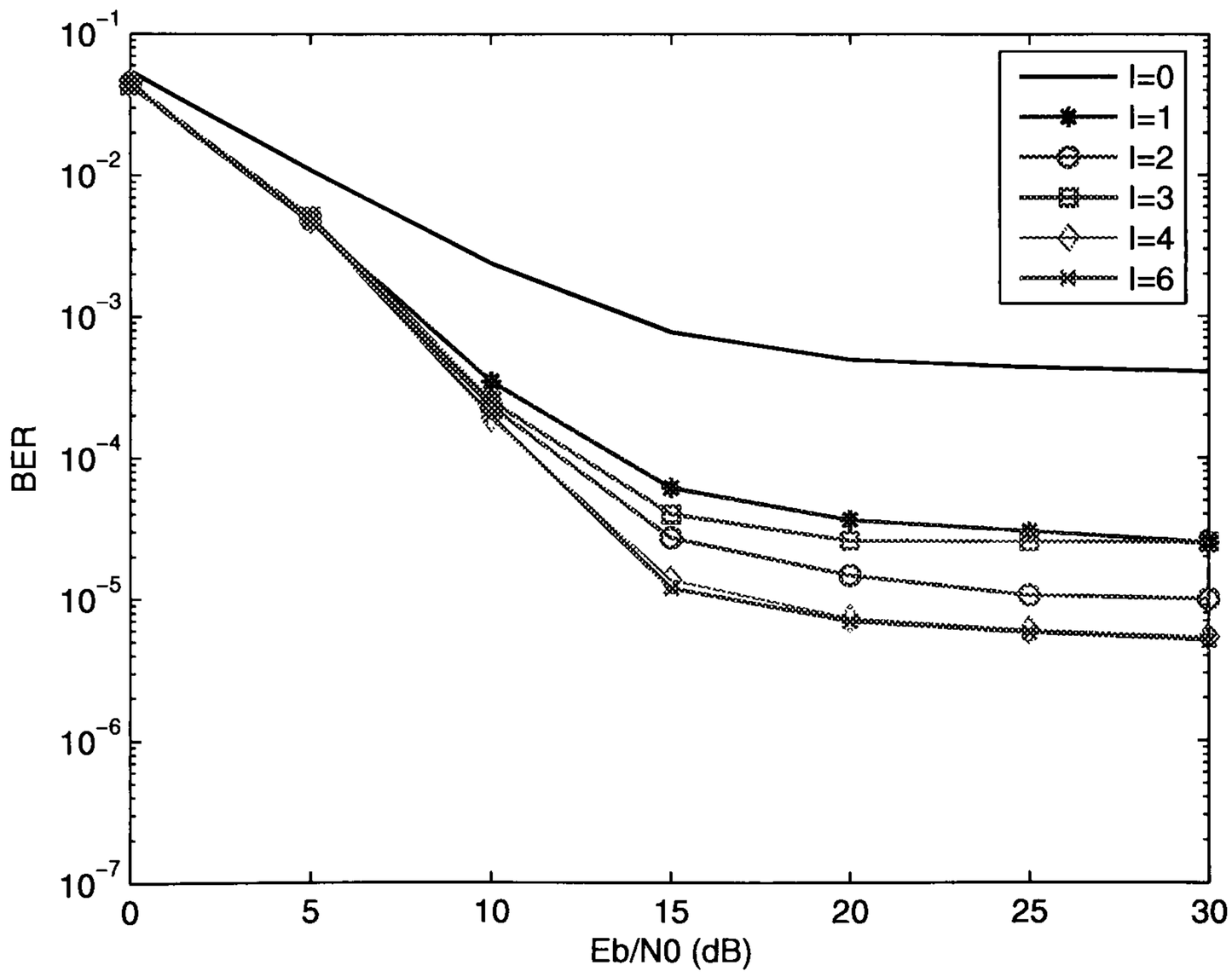


Figure 5.9: BER performance of MF-based PIC (soft) in $\mathbf{G}_4$ coded SFBC-OFDM over quasi-static frequency selective fading channels

in 'hard PIC', the decoder almost reaches its best performance at the second iteration ('I=2'). Also, 'soft PIC' presents much lower error floor than 'hard PIC' and obtains more improvement with an increasing number of the iterations. As discussed, in both cases, there still exists an error floor at high E_b/N_0 regions due to error propagation in some blocks of the symbol. The error oscillation mode causes the BER performance to 'oscillate' with different number of iterations. For example, the 'I=3' curve, which is expected to be better, is actually worse than the 'I=2' curve, while the 'I=4' curve is again better than 'I=2' curve, demonstrating the oscillatory behavior.

5.4 Code-aided Outer Iterative PIC Decoding for G_4 SFBC-OFDM

In order to improve the performance of the PIC detector, it is possible to exploit the additional information available from an outer FEC decoder in addition to the SFBC inner code. At the output of the outer FEC decoder, more reliable information is available from the whole outer codeword, and this can be used to further improve the interference cancelation in the PIC detector. The detection and outer decoding may then be iteratively repeated, in the manner of the 'turbo-principle' [21]. Relative to inner iterations, these are termed 'outer iterations'. In this thesis, an outer convolutional code is used.

An interleaver is used at the transmitter between the outer encoder and the SFBC encoder, and correspondingly a de-interleaver at the receiver between the PIC detector and the outer decoder. This enables frequency diversity to be exploited on long-delay channels.

Firstly, a SIHO Viterbi decoding is considered. The Viterbi decoder outputs code parity bits as well as data bits (both are hard information) so as to regenerate the coded data. This is then passed through the block interleaver and the modulator, and fed back to provide the initial values of the inner iteration in the next iteration. Therefore, the initial values

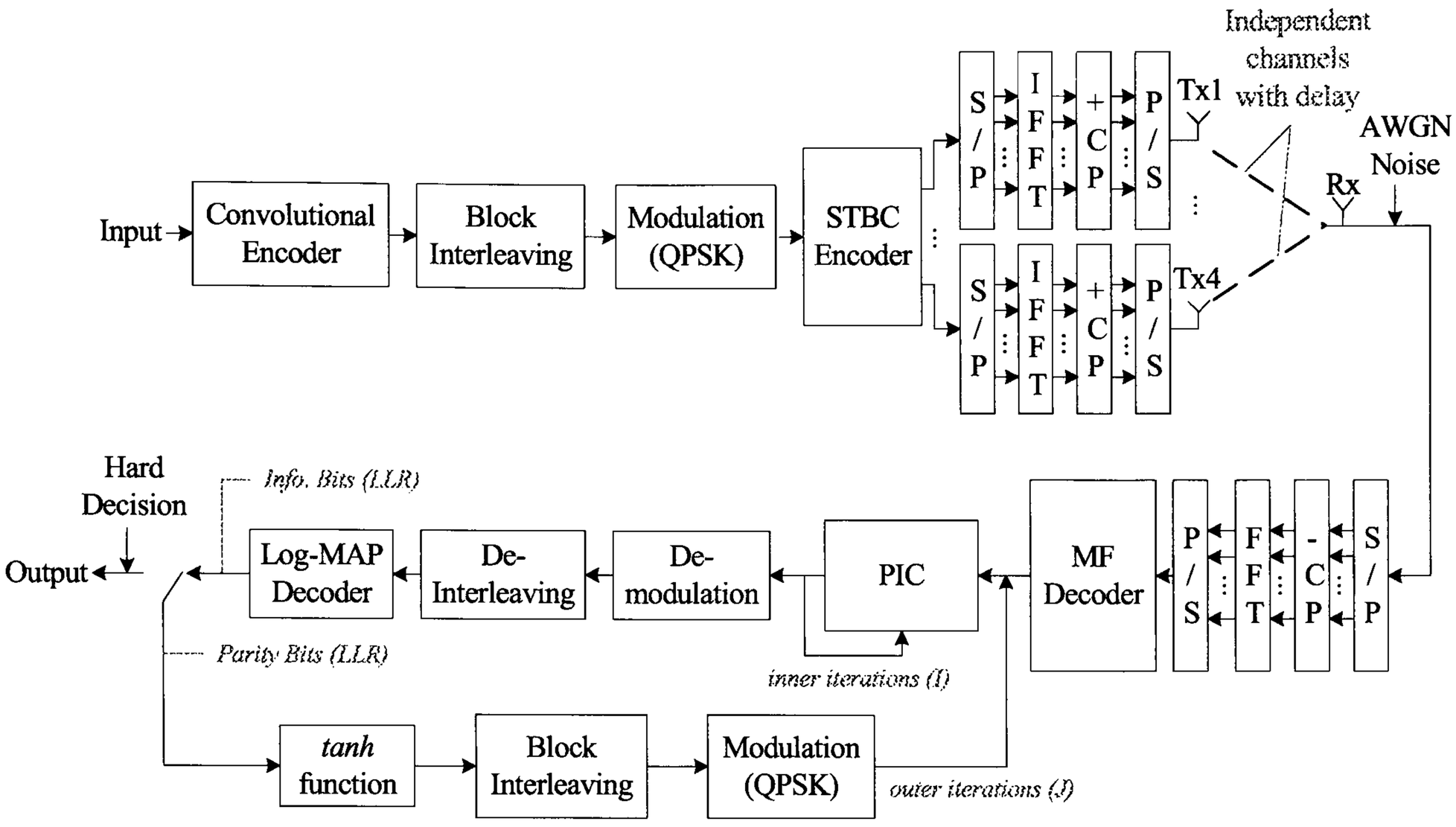


Figure 5.10: System structure of code-aided outer iterative PIC decoding for $\mathbf{G}_4$ SFBC-OFDM

$\hat{\mathbf{S}}^{(0)}$ originally obtained from the conventional MF decoding (Eq. 5.34) are corrected by the convolutional outer code before the next inner iteration, which makes the inner PIC iterations more efficient.

The Viterbi decoding algorithm provides a maximum likelihood 'hard' decision on the transmitted data sequence. However, making hard decisions represents a loss of valuable information: it is advantageous to retain 'soft' decisions on the probability of the coded symbols until all possible information has been extracted from the received signal values. Therefore, an algorithm is required that can output soft-decision maximum likelihood estimates as well as accept soft-decision inputs from the previous iteration of the decoding process. The decoder using this algorithm is termed a 'SISO' decoder.

Inspired by the turbo decoding, the MAP algorithm [18] can be used for this purpose. Specifically the Log-MAP version of this algorithm [24] is suggested. The difference is that the feedback information for the initial values of PIC iterations $\hat{\mathbf{S}}^{(0)}$ in Eq. 5.34 is soft, being obtained from the LLR of the parity bits, when the Log-MAP algorithm is

used, as opposed to hard when using Viterbi algorithm. The soft decision of coded bits is given by

$$\hat{S} = tanh(S_{LLR}/2) \tag{5.37}$$

where S_{LLR} represent the LLR of parity bits from the Log-MAP decoder. After interleaving and modulation, which are the same as in the coding process at the beginning, the 'soft symbols' are sent back as the initial values of PIC iterations. The system structure is illustrated in Fig. 5.10.

Note that the decoder in this case provides the LLR for both the information and parity bits. Also note that the Log-MAP algorithm used for a single convolutional code is different from that for a turbo code, where two RSC codes are included in the encoding structure. Instead of two RSC decoders helping each other in an iterative manner, only one decoder is concerned with the decoding process. No additional information can be obtained from the other decoder and it is useless to feedback the soft information to the decoder itself. Therefore, from the decoder point of view, the SISO Log-MAP decoder and SIHO Viterbi decoder have pretty much the same performance. It is the soft output from the Log-MAP decoder that benefits the inner PIC iterations, which can then provides better interference canceled data for the outer Log-MAP decoding iterations. Therefore, the overall performance is improved by both iterations helping with each other.

Fig. 5.11, 5.12 and 5.13 illustrate the performance of the code-aided outer iterative PIC with different numbers of outer iterations (J). '$J = 0$' means no outer iterations: that is, the outer decoder is applied only once, and there is no feedback of its output. For inner iterations, the 'soft PIC' is chosen, as it performs much better than the 'hard PIC'. From the results in Fig. 5.9, although most gain comes from the first iteration, '$I = 4$' provides the best performance, while the improvement seems to cease at '$I = 6$'. Therefore, to minimize the complexity without degradation in the performance, the inner iteration is implemented 4 times ($I = 4$). Note that however, much more improvement can be obtained from the code-aided outer iterations than from the inner alone. The FFT size is 256 and the channel order is 12 in all cases.

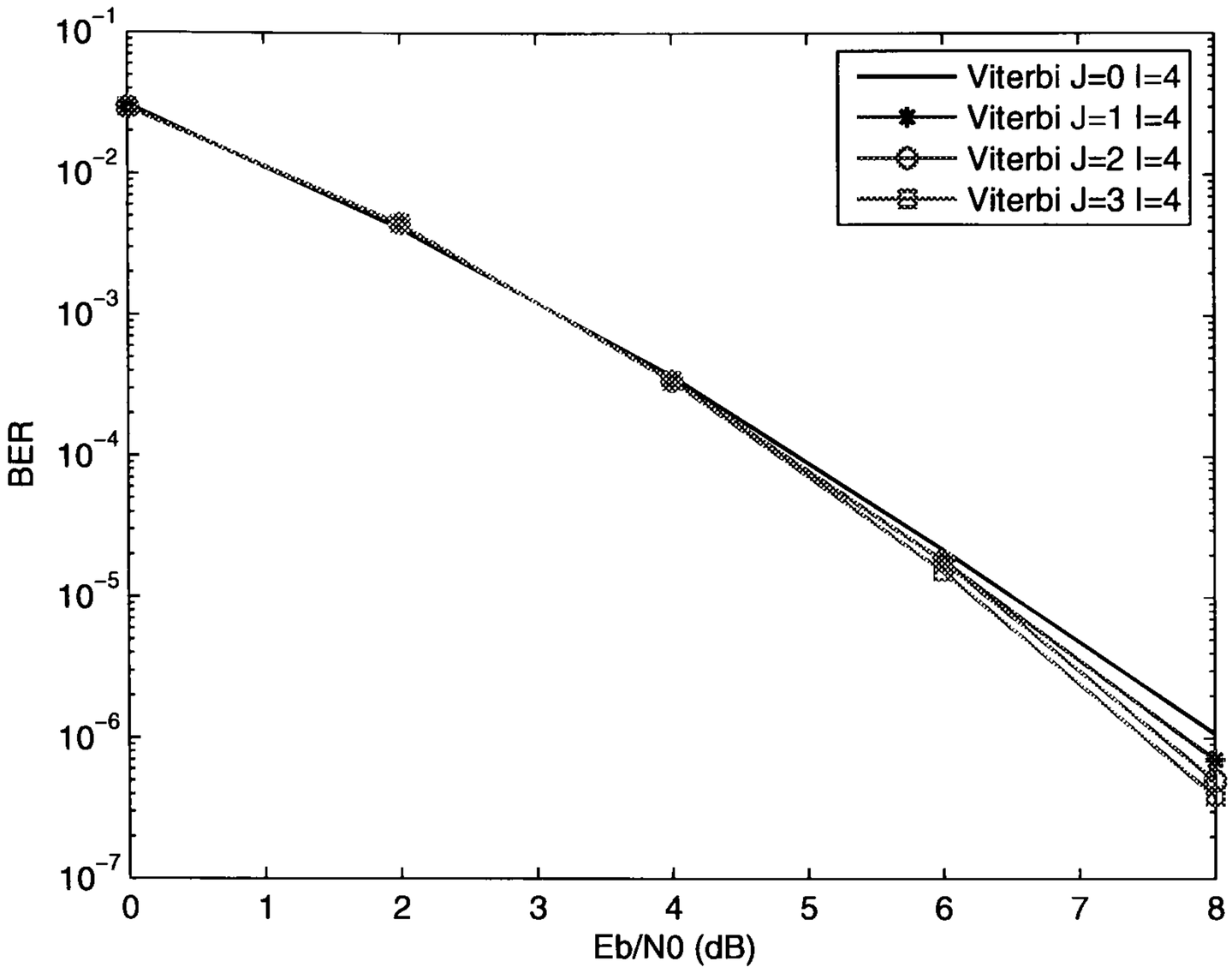


Figure 5.11: BER performance of code-aided outer iterative (Viterbi) PIC in $\mathbf{G}_4$ SFBC-OFDM over quasi-static frequency selective fading channels

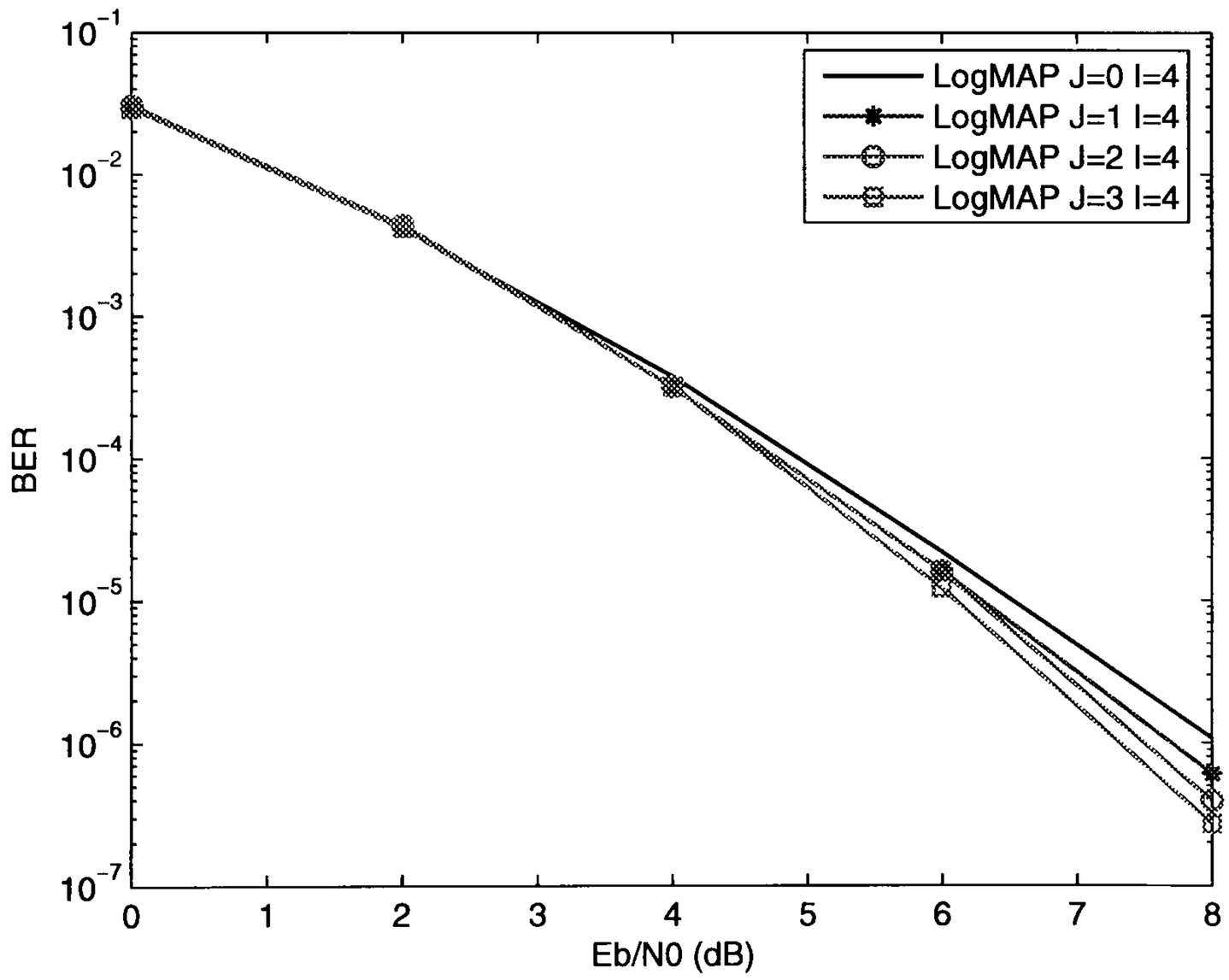


Figure 5.12: BER performance of code-aided outer iterative (Log-MAP) PIC in $\mathbf{G}_4$ SFBC-OFDM over quasi-static frequency selective fading channels

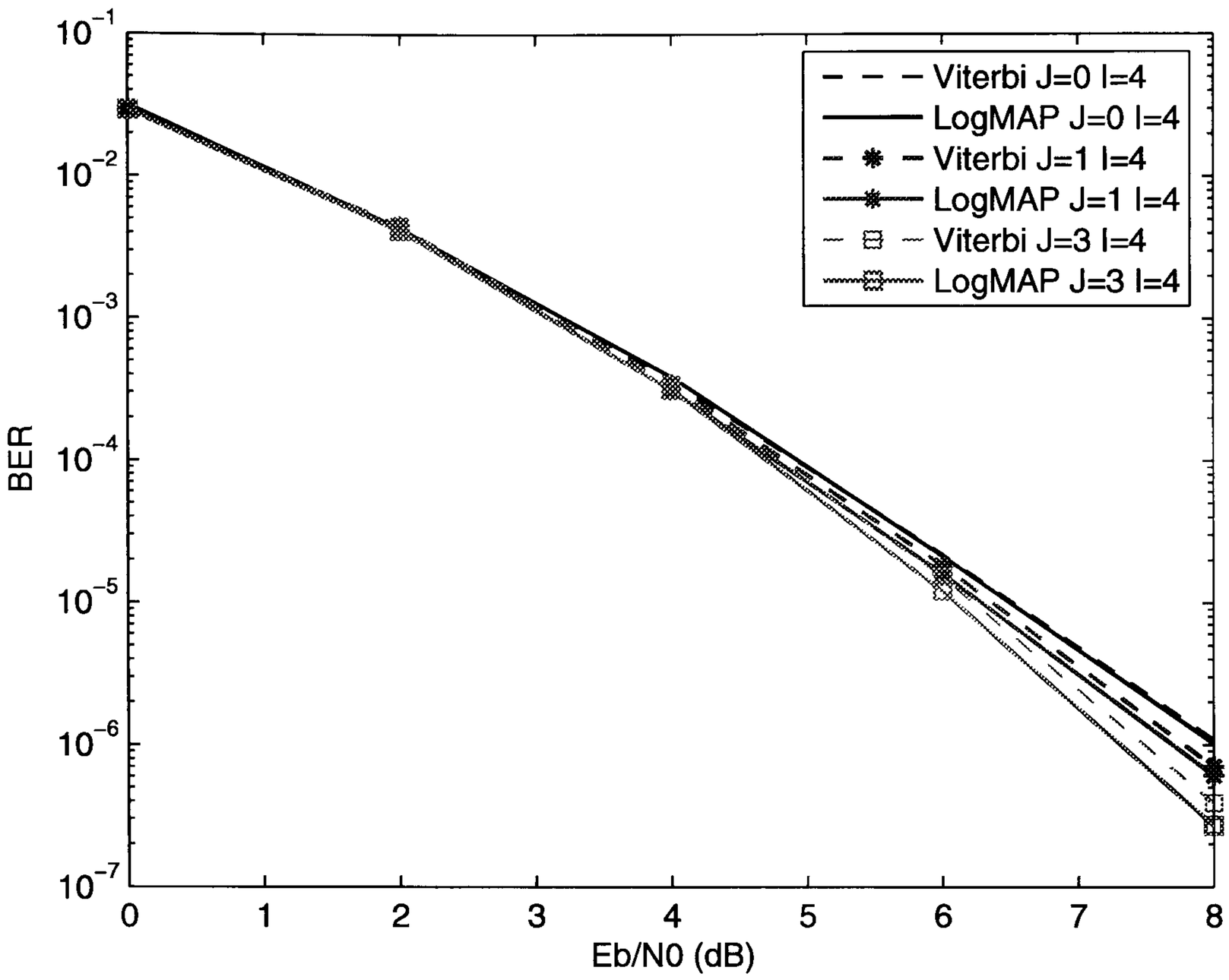


Figure 5.13: BER performance comparisons between code-aided outer iterative (Log-MAP) PIC in $\mathbf{G}_4$ SFBC-OFDM over quasi-static frequency selective fading channels

Fig. 5.11 and Fig. 5.12 give the performance of outer iterations by using SIHO Viterbi decoding and SISO Log-MAP decoding, respectively. Again, most improvement comes from the first iteration ('$J = 1$'). With the increased number of outer iterations ('$J > 1$') the performance is more improved, especially at higher E_b/N_0 (about 0.5 dB for Viterbi deocding and 0.7 dB for Log-MAP decoding at the BER of 10^{-6} after 3 outer itertions), although the returns from increasing the number above '$J = 1$' are relatively small.

In Fig. 5.13, the performance comparisons between outer iterations by using Log-MAP decoding and that by using Viterbi decoding are illustrated. In the outer-iteration process, Log-MAP decoder feed back soft information, while Viterbi decoder feeds back hard information. Without outer iterations ($J = 0$), Log-MAP decoding and Viterbi decoding have nearly identical performance. With 1 iteration ($J = 1$), Log-MAP decoder works slightly better than Viterbi decoder: about 0.1 dB gain at BER rate of 10^{-6}. After 3

iterations ($J = 3$), Log-MAP decoder gives a 0.2 dB gain over Viterbi decoder at BER rate of 10^{-6}. It seems likely that at higher E_b/N_0 and with more iterations, the gain of Log-MAP decoder over Viterbi decoder will become larger. Note that although small improvements can be provided by outer iterations, but it is mainly the coding itself that makes difference, compared with the uncoded cases (Fig. 5.8 and Fig. 5.9).

5.5 Summary

This chapter has mainly discussed two issues in the application of orthogonal STBC to OFDM systems. One is about the BER performance comparisons between STBC-OFDM and SFBC-OFDM over time and frequency selective fading channels. It has been concluded that STBC-OFDM is sensitive to channel time variations, but its decoding process is not easily affected by the multipath fading environment; SFBC-OFDM is vulnerable to both channel time variations between samples and channel frequency response variations between subcarriers. The other issue is about canceling ICI in $\mathbf{G}_2$ and $\mathbf{G}_4$ coded SFBC-OFDM systems, due to the channel frequency response variations between OFDM subcarriers, assuming the multipath channel remains constant during one OFDM symbol period. ZF Algorithm has been proved to be able to eliminate the ICI effect in $\mathbf{G}_2$ SFBC-OFDM, albeit at the cost of the diversity gain. However, due to its high complexity, ZF algorithm is not suggested in $\mathbf{G}_4$ SFBC-OFDM. In our solution, a PIC approach with soft iterations has been applied to subtract the ICI existing in the output of the MF decoder. Furthermore, to refine the output of the MF decoder, which is also the initial input of the inner PIC iterations, the outer convolutional coding has been added and the soft information provided by the SISO Log-MAP decoder has been fed back to the inner PIC iterations. By this means, the coded data bits are better detected, which can help the Log-MAP decoder provide more accurate feedback information in the outer iteration. It has been shown from the simulation results that the system benefits from both inner PIC and outer Log-MAP iterations.

Chapter 6

Performance Evaluation of IEEE 802.16 WMAN Physical Layer OFDMA

Contents

Recently, WiMAX technology, developed by the IEEE 802.16 task group (TG), has attracted a great deal of attention in the wireless communications world. It has been regarded as a strong competitor to the 3G system. Hundreds of companies have contributed to the development of the technology and many companies have announced product plans for this technology [44].

This chapter will provide an overview of the scalable OFDMA physical layer in the IEEE 802.16 standard, and access the practical performance of an OFDMA receiver in a typical mobile channel with the basic physical-layer configurations. This includes to implement

channel estimations from preambles and pilots as specified in the standard and measure how much deterioration in the estimation affects the BER; to incorporate practical timing recovery modeling and test the BER degradation caused by imperfect timing; to investigate the impact from frequency offset and the necessity of implementing frequency synchronization.

We will show that the degradation from channel estimation is less than 1 dB both in DL and UL, and moreover, the degradation can be further reduced by applying a code-aided iterative channel estimation approach. Also, to solve the problem of mismatched estimators, the estimator is optimized for the worst channel conditions so that the performance will at least be the same as the matched cases in these channels. Imperfect timing will affect the BER performance of OFDMA, and hence a timing recovery scheme is introduced. Frequency offset degrades the performance as well, however, within the tolerance specified in the standard, the correction is not necessary.

The work in this chapter is funded by British Telecommunications (BT).

6.1 Scalable OFDMA Physical Layer in IEEE 802.16

A basic description of the physical layer OFDMA of IEEE 802.16 standard is given in this section.

6.1.1 WiMAX Basics

The WiMAX technology has been playing a key role in fixed, portable and mobile broadband WMAN. The technology is based on the IEEE 802.16-2004 Air Interface Standard [2], which is proven to be an effective fixed wireless alternative to conventional

cable and DSL services [44]. The IEEE 802.16e Amendment [45] has been able to support the mobility of WiMAX system with added features and attributes to the IEEE 802.16-2004 standard. At present, the system performance is being evaluated based on the IEEE 802.16e Mobile Amendment and a network architecture for an end-to-end Mobile WiMAX network is being defined [44].

WMAN based on the WiMAX air interface standard is configured in much the same way as a traditional cellular network. It has base stations and uses a point-to-multipoint architecture to deliver services within a radius of several kilometers. The physical layer of the WiMAX system has a wide range of frequency from 2 up to 66 GHz. The sub-range 10-66 GHz is for Line of Sight (LOS) operation in fixed wireless networks, and single carrier (SC) modulation is chosen because of the low complexity of the system. The sub-range 2-11 GHz, including licensed (2-6 GHz) and licensed-exempt (7-11 GHz) bands, is for Non Line of Sight (NLOS) fixed, portable and mobile networks. Owing to its superior performance in multipath fading wireless channels, OFDM and OFDMA is recommended for the physical layer of the standard for NLOS applications.

6.1.2 OFDMA Symbol Description and Sub-channelization

The time domain description of an OFDM/OFDMA symbol is identical to that of a regular OFDM described in Chapter 3. The duration of CP should be selected by the base station (BS) for DL transmission and kept the same value by subscriber station (SS) on UL.

In the frequency domain, an OFDM/OFDMA symbol is composed of three types of sub-carriers: data subcarriers, pilot subcarriers and null subcarriers. Data subcarriers are used for data transmission, while pilot subcarriers are for various estimation purpose. As they both contain useful information, data and pilot subcarriers are denoted as active subcarriers. Null subcarriers transmit nothing at all, which usually include guard bands at both ends of the symbol and DC subcarriers in the middle. Guard bands are to enable the signal

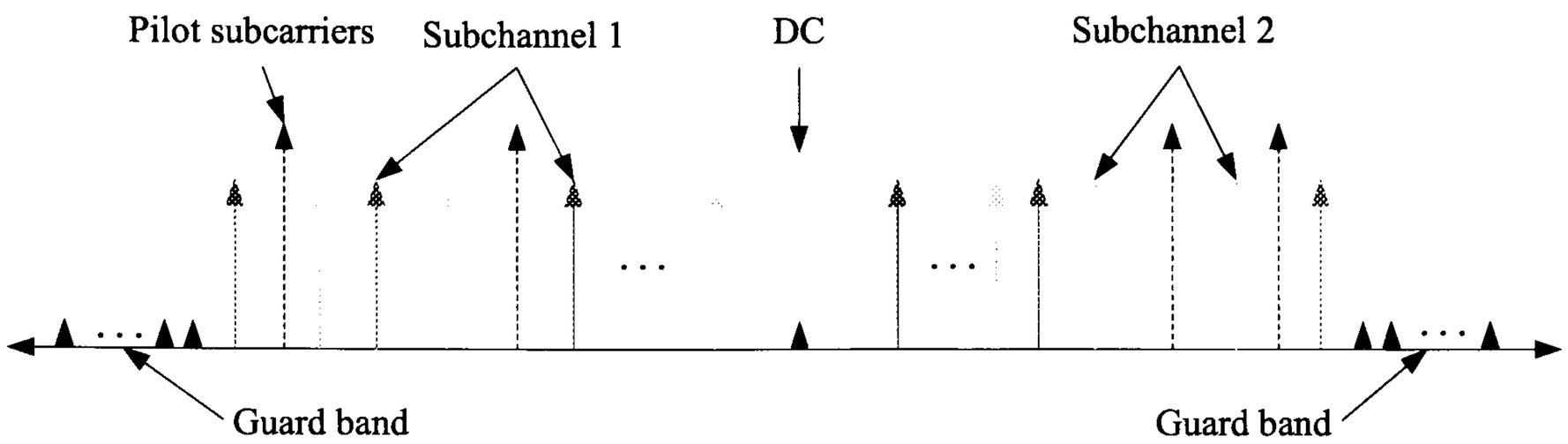


Figure 6.1: Diversity permutated OFDMA subcarriers

to naturally decay and create the FFT 'brick wall' shaping [2].

Different from OFDM, OFDMA is a multiple access scheme that allows subcarriers to be assigned to different users. Active subcarriers are grouped into subsets of subcarriers called sub-channels. The WiMAX OFDMA physical layer supports sub-channelization in both DL and UL [2]. Usually, more demands are placed on the DL transmission than UL, hence a sub-channel may be intended for several users in DL, while in UL, an user can be assigned one or more sub-channels and several users may transmit simultaneously.

Sub-channelization is compulsory in both DL and UL. To do this, two types of subcarriers permutation mechanisms have been proposed in the standard: contiguous permutation and diversity permutation. Just as their names imply, the contiguous permutation groups contiguous subcarriers to form a sub-channel, while the diversity permutation spread the subcarriers along the channel spectrum in order to obtain the full frequency diversity offered by the multipath channel; hence the subcarriers forming one sub-channel do not need to be consecutive. A schematic example of the OFDMA subcarriers (diversity permutation) of one symbol is depicted in Fig. 6.1. The standards provided some diversity permutation schemes, e.g., DL Fully Used Sub-Channelization (FUSC), DL Partially Used Sub-Channelization (PUSC), UL PUSC and additional UL optional PUSC (OPUSC).

Parameters	Values				
System bandwidth (B, MHz)	1.25	2.5	5	10	20
Sampling frequency (F_s, MHz)	1.429	2.857	5.714	11.429	22.857
Sampling time ($1/F_s, ns$)	700	350	175	88	44
FFT Size (N_s)	128	256	512	1024	2048
Subcarrier frequency spacing (f_s, kHz)	11.16				
Useful symbol time ($T_b = 1/f_s, \mu$s)	89.6				
Guard time ($T_g = T_b/8, \mu$s)	11.2				
OFDMA symbol time ($T_s = T_b + T_g, \mu$s)	100.8				

Table 6.1: OFDMA scalability parameters [4]

6.1.3 Scalable OFDMA

The concept of scalable OFDMA (S-OFDMA) was introduced to IEEE 802.16 WMAN OFDMA mode by the IEEE 802.16 TG e (TGe) [4]. S-OFDMA supports a wide range of bandwidths from 1.25 MHz to 20 MHz to flexibly address the need for various spectrum allocations required by different fixed, portable and mobile usage models. The scalability means that the FFT size can be adjusted according to the channel bandwidth required, while the subcarrier spacing is fixed at 11.16 kHz[1]. S-OFDMA is able to deliver the optimum performance since it keeps cost low for requirements on less FFT size and small bandwidth, while allows larger FFT size for increased performance with wider channels. The constant subcarrier frequency spacing and symbol duration reduce the impact to higher layers to the minimum when scaling the bandwidth [44]. S-OFDMA is especially indispensable to WiMAX in guaranteeing the performance in vehicular mobility multipath environments [4]. Table 6.1[2] gives a list of S-OFDMA parameters. The table

[1]This is obtained to support a delay spread of up to 20 μs, where the channel model is given by the International Telecommunications Union - Radiocommunications (ITU-R) Vehicular Channel Model B. For delay spread values of up to 20 μs), multipath fading can be considered as flat fading over 11.16 kHz subcarrier width [4].

[2]$F_s = floor(8/7B/0.008) \times 0.008$, as specified in the standard.

indicates that the scalability is able to ensure a minimum number of OFDMA symbols in a frame, which avoids the problem of high overhead in a small frame size. For example, for a short frame size of 2 ms, the number of OFDMA symbols is a moderate 19. However without scalability, for an OFDMA symbol with 1.25 MHz bandwidth and a FFT size of 2048, the symbol time will be 1612.8 μs, which means there are less than 2 OFDMA symbols if the frame duration is 2 ms. This may make the frame unusable due to the relatively high overhead in such a short frame.

6.1.4 TDD Frame Structure

The IEEE 802.16-2004 standard [2] supports Time Division Duplexing (TDD), Frequency Division Duplexing (FDD) and Half-FDD (H-FDD) modes. In TDD, DL and UL share a single channel but operate in different time slots. The channel bandwidth for DL and UL can be asymmetric and adjustable to adapt to varied spectrum allocations. Unlike TDD, FDD requires a pair of channels for DL and UL separately, but it allows simultaneous transmission in both directions. The channel bandwidth are fixed and generally equal for DL and UL.

TDD is the duplex mode adopted. Fig. 6.2 illustrates an OFDMA frame structure in TDD mode. Each frame is divided into DL and UL sub-frames separated by gaps to avoid DL and UL collisions. The DL sub-frame stars with a preamble symbol for synchronization and estimation purposes. The preamble symbol is followed by a Frame Control Header (FCH) that provides the frame configuration information such as coding scheme and usable sub-channels, and DL/UL-MAP messages giving information about sub-channel allocation etc.. The DL and UL bursts are for data transmissions. The data are mapped from the lowest numbered sub-channel and OFDMA symbol in a data region, and continued with an increased OFDMA symbol index in the same sub-channel. When the symbol edge of the date region is reached, the mapping is turned around to the next available sub-channel and starts again from the lowest numbered OFDMA symbol. The UL sub-

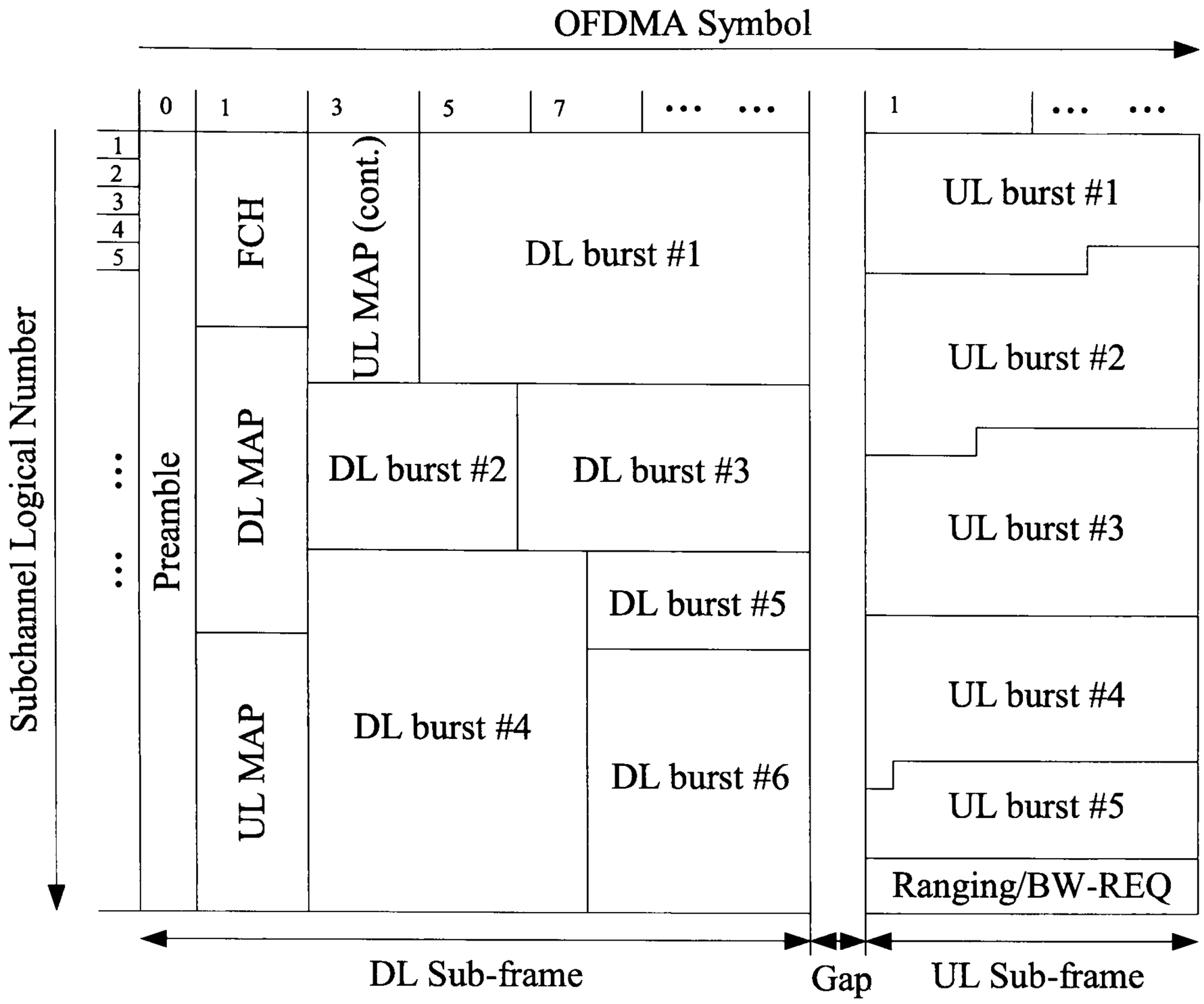


Figure 6.2: OFDMA frame structure (TDD) [2]

channels are also used for ranging (RNG) as well as bandwidth request (BW-REQ).

6.2 Channel Estimation in OFDMA

As discussed in Chapter 3, data detection in an OFDM based system is performed coherently by using a 'single-tap' frequency-domain equalization at the receiver, thus requiring knowledge regarding the channel frequency response. In our pervious work, it is assumed that PCK is known at the receiver. However in practice, it is necessary to obtain the channel state information by certain means of channel estimations. The channel estimation in IEEE 802.16 OFDMA physical layer can be done either in DL or UL.

In UL PUSC and OPUSC, a sub-channel is constructed from a certain number of 'tiles', across both subcarriers and symbols, with pilots scattered at appointed positions. A pilot-based channel estimation scheme is suited to this kind of structure. Here, a linear minimum mean square error (LMMSE) estimation approach by using Wiener filter [46] is employed. Given the frame structure of OFDMA, Wiener filtering can either perform a two-dimensional (2D) estimation or two cascaded one-dimensional (2×1D) estimations.

Inspired by the principle of 'iterative decoding', the additional benefit from outer coding and the iterative property of Log-MAP decoders an be exploited. There will be a potential gain from feeding back soft bit estimates from the Log-MAP decoder and use them together with the known pilot symbols for next-iteration estimation. Hence the quality of channel estimation is enhanced without adding extra pilots.

Unlike UL, DL transmission starts with a preamble, which can be effectively used as a training symbol for the channel tracking purpose. To do this, a code-aided channel tracking approach based on the Expectation Maximization (EM) algorithm [47] is applied. Besides the preamble symbol, scattered pilot subcarriers across both time and frequency also exist in DL FUSC and PUSC, although in a different way from UL. To make use of these pilots, the LMMSE channel estimation can be applied immediately after the channel tracking to further refine the estimates. The estimates from channel tracking, as well as the original pilots, are treated together as known pilots in the next-step LMMSE channel estimation. Note that in DL FUSC, sufficient pilots are provided for each user; hence a single LMMSE estimator is good enough to provide a satisfied performance. Whereas in DL PUSC, the pilots for each user is much less; therefore channel tracking and estimation are suggested to work collaboratively to create the best performance.

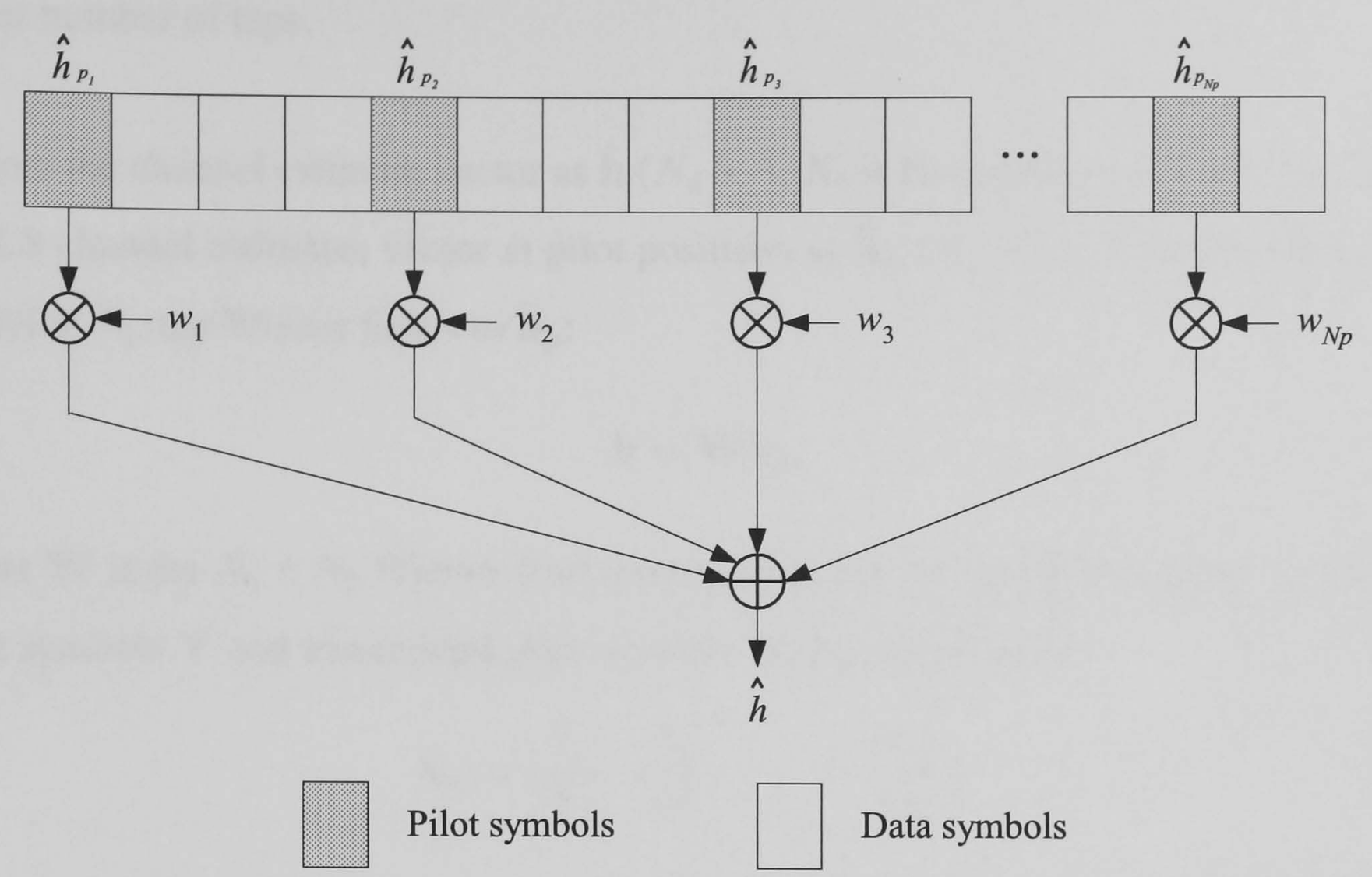


Figure 6.3: Wiener filtering

6.2.1 Channel Estimation by Wiener Filtering

The basic principle of pilot-assisted channel estimation is to insert pilot symbols known by the receiver into transmitted data at different time or frequency, and at the receiver, estimate the channel values at any time or frequency given the observations of the pilots. Wiener filtering is a LMMSE estimation approach that is capable of achieving a low accuracy bound [48]. The problem of this estimator is to find the channel estimates as a linear combination of the pilot estimates and provide noise suppression as well.

Provided channel information at pilot symbol positions, channel estimates on data symbol positions can be obtained by interpolation, if the channel correlations are known beforehand. Fig. 6.3 illustrates an interpolation by using a N_p-tap Wiener filter (N_p is the number of pilot symbols). An optimal interpolation is different for different data positions. Theoretically, an infinite length Wiener filter is optimal in the sense of minimum mean square error (MMSE). However, a finite length has to be used in practice to save complexity. Apparently, more estimation accuracy is obtained from a Wiener filter with

larger number of taps.

Denote the channel estimate vector as $\hat{\mathbf{h}}$ ($N_d \times 1$, N_d is the number of data symbols) and the LS channel estimates vector at pilot positions as $\hat{\mathbf{h}}_\mathbf{p}$ ($N_p \times 1$). $\hat{\mathbf{h}}$ can be obtained by applying N_p-tap Wiener filters to $\hat{\mathbf{h}}_\mathbf{p}$:

$$\hat{\mathbf{h}} = \mathbf{W}\hat{\mathbf{h}}_\mathbf{p} \tag{6.1}$$

where $\mathbf{W}$ is the $N_d \times N_p$ Wiener filter interpolation matrix. In OFDM, given N_p received pilot symbols $\mathbf{Y}$ and transmitted pilot symbols $\mathbf{X}$, $\hat{\mathbf{h}}_\mathbf{p}$ are given by

$$\hat{\mathbf{h}}_\mathbf{p} = [\frac{Y_1}{X_1} \quad \frac{Y_2}{X_2} \quad \cdots \quad \frac{Y_{N_p}}{X_{N_p}}]^{\mathrm{T}} \tag{6.2}$$

where both $\mathbf{X}$ and $\mathbf{Y}$ are $N_p \times 1$ vectors. Note that $\hat{\mathbf{h}}_\mathbf{p}$ are rough estimates including the effect of noise. The variance of estimation errors is equal to that of the noise.

Define the channel correlation as R_{hh}:

$$R_{hh} = \mathrm{E}\{h(\gamma_1)h^*(\gamma_2)\} \tag{6.3}$$

where γ_1 and γ_2 denote either time or frequency. The aim of a MMSE estimator is to minimize the mean square error (MSE) between actual channel values and channel estimates. According to [49], this is satisfied by letting

$$\mathbf{W} = \mathbf{R}_{\mathbf{dp}}\mathbf{R}^{-\mathbf{1}}_{\hat{\mathbf{p}}\hat{\mathbf{p}}} \tag{6.4}$$

where $\mathbf{R}_{\mathbf{dp}}$ is a $N_d \times N_p$ matrix containing cross correlations between data symbol positions and pilot symbols positions; $\mathbf{R}_{\hat{\mathbf{p}}\hat{\mathbf{p}}}$ is a $N_p \times N_p$ matrix containing auto-correlations between pilot symbol positions including noise smoothing. Given the noise power σ_n^2, $\mathbf{R}_{\hat{\mathbf{p}}\hat{\mathbf{p}}}$ is obtained by

$$\mathbf{R}_{\hat{\mathbf{p}}\hat{\mathbf{p}}} = \mathbf{R}_{\mathbf{pp}} + \sigma_n^2(\mathbf{X}\mathbf{X}^{\mathrm{H}})^{-1} \tag{6.5}$$

where $\mathbf{R}_{\mathbf{pp}}$ is a $N_p \times N_p$ matrix containing auto-correlations between pilot symbol positions. The elements in $\mathbf{R}_{\mathbf{dp}}$ and $\mathbf{R}_{\mathbf{pp}}$ are all obtained from the channel correlation R_{hh}.

Eq. 6.5 can be simplified to [46][3]

$$\mathbf{R}_{\hat{\mathbf{p}}\hat{\mathbf{p}}} = \mathbf{R}_{\mathbf{pp}} + \frac{\beta}{\mathrm{SNR}_p}\mathbf{I}_{\mathbf{N_p}} \tag{6.6}$$

where $\mathbf{I}_{\mathbf{N_p}}$ is a $N_p \times N_p$ identity matrix and β is a constant depending on the signal constellation, e.g., $\beta = 17/9$ for 16-QAM and $\beta = 1$ for QPSK; SNR_p denotes the pilot symbol power E_s versus the noise power σ_n^2:

$$\mathrm{SNR}_p = \frac{E_s}{\sigma_n^2} \tag{6.7}$$

Note that if the powers of pilots are not equal, SNR_p shall be written into the matrix $\mathbf{I}_{\mathbf{N_p}}$ individually on the corresponding diagonal positions.

Substituting Eq. 6.6 into Eq. 6.4, the interpolation matrix $\mathbf{w}$ is expressed as

$$\mathbf{w} = \mathbf{R}_{\mathbf{dp}}(\mathbf{R}_{\mathbf{pp}} + \frac{\beta}{\mathrm{SNR}_p}\mathbf{I}_{\mathbf{N_p}})^{-1} \tag{6.8}$$

Finally, the channel estimates can be obtained by

$$\hat{\mathbf{h}} = \mathbf{R}_{\mathbf{dp}}(\mathbf{R}_{\mathbf{pp}} + \frac{\beta}{\mathrm{SNR}_p}\mathbf{I}_{\mathbf{N_p}})^{-1}\hat{\mathbf{h}}_{\mathbf{p}} \tag{6.9}$$

Eq. 6.9 basically gives the desired channel estimates as the multiplication of an interpolation matrix with the LS channel estimates at pilot positions. Notice that the interpolation matrix $\mathbf{W}$ is independent of the received symbols; it only depends on the positions and numbers of the pilot and data symbols. Furthermore, if the channel model or certain property of the fading channel (e.g., power delay profile, delay spread, normalized Doppler frequency and so on) is given, $\mathbf{R}_{\mathbf{dp}}$ and $\mathbf{R}_{\mathbf{pp}}$ are known beforehand. Therefore during simulations, the inversion of $(\mathbf{R}_{\mathbf{pp}} + \frac{\beta}{\mathrm{SNR}_p}\mathbf{I})$ does not need to be calculated each time the transmitted pilots $\mathbf{X}$ change, but only when E_b/N_0 changes, which saves a lot of complexity in matrix inversions.

In OFDM system, estimations are obtained across time or frequency by using channel time or frequency correlations respectively. A 2×1D estimation first interpolates in frequency (or time) to obtain the estimates at data symbol positions (may including the pilot

[3]Here we don't show the derivative process from Eq. 6.5 to Eq. 6.6. Interested readers are referred to reference [46].

symbol positions as well) in that direction, and then turns to the other direction for estimates at the remaining data symbol positions. Instead, it is also possible to implement a 2D interpolation by performing estimations across two directions simultaneously. The only difference is that the channel frequency and time auto-correlations are used individually in 2×1D estimations, while in 2D estimations, they are the products of the two.

Define the channel time and frequency correlations as R^t_{hh} and R^f_{hh} respectively:

$$R^t_{hh} = \mathrm{E}\{h(t_1)h(t_2)\} \tag{6.10}$$

$$R^f_{hh} = \mathrm{E}\{h(f_1)h(f_2)\} \tag{6.11}$$

The discrete (sampled at OFDM symbol intervals or OFDM subcarrier spacings) form of R^t_{hh} and R^f_{hh} are written in matrices $\mathbf{R^t_{hh}}$ and $\mathbf{R^f_{hh}}$. The mapping of discrete channel time and frequency correlations onto matrices $\mathbf{R^t_{hh}}$ and $\mathbf{R^f_{hh}}$ are described as follows. (m, n) stands for the element at m-th row and n-th column of $\mathbf{R^t_{hh}}$ and $\mathbf{R^f_{hh}}$. For $\mathbf{R_{dp}}$, m ($1 \leq m \leq N_d$) are the order numbers in the data symbol sequence or subcarrier sequence, and n ($1 \leq n \leq N_p$) are those in the pilot symbol sequence or subcarrier sequence. For $\mathbf{R_{pp}}$, both m and n ($1 \leq m, n \leq N_p$) denote order numbers in the pilot symbol sequence or subcarrier sequence. m_p and n_p represent the the m-th and n-th symbol position or subcarrier position in all OFDM symbols or OFDM subcarriers, e.g., assuming in Fig. 6.3, data and pilots are allocated across OFDM symbols, the second pilot ($n = 2$) is located in the fifth OFDM symbol ($n_p = 5$).

1. *Channel time correlation*

   For a time-fading signal with a maximum Doppler frequency f_d and a Jakes spectrum, the time correlation matrix $\mathbf{R^t_{hh}}$ is given by [3]

$$\mathbf{R^t_{hh}}(m, n) = J_0(2\pi f_d T_s(m_p - n_p)) \tag{6.12}$$

   where J_0 is the 0-th order Bessel function of the first kind and T_s is the OFDM symbol period including the guard time[4]; m_p and n_p are data or pilot positions over

[4] Although due to Jake's model, channel varies over OFDM samples, but to simplify the simulation, we keep the channel constant over the OFDM symbol period, therefore ICI is ignored.

all transmitted OFDM symbols.

2. *Channel frequency correlation*

   Channel frequency correlations are obtained differently depending on the power-delay profile of the multipath channel. For a uniformly distributed power-delay profile, $\mathbf{R_{hh}^f}$ is calculated by [46]

$$\mathbf{R_{hh}^f}(m,n) = \frac{1-\exp(-2\pi jL(m_p-n_p)/N_s)}{2\pi jL(m_p-n_p)/N_s} \tag{6.13}$$

   where L is the order of the multipath channel; N_s is the number of subcarriers (or FFT size); m_p and n_p are data or pilot subcarriers positions over all OFDM subcarriers. If the channel has an exponentially decaying power-delay profile, $\mathbf{R_{hh}^f}$ is expressed as [3]

$$\mathbf{R_{hh}^f}(m,n) = \frac{1}{1+2\pi j\tau_{rms}(m_p-n_p)f_s} \tag{6.14}$$

   where f_s is subcarrier spacing, taken from the inverse of the useful OFDM symbol time T_b (excluding the guard time); τ_{rms} is the root-mean-square (rms) delay spread of the multipath channel.

The MSE of the above LMMSE channel estimation is calculated by

$$\hat{\text{MSE}} = E\{|\mathbf{h}-\hat{\mathbf{h}}|^2\} \tag{6.15}$$

where $\mathbf{h}$ and $\hat{\mathbf{h}}$ contains the actual and estimated channel values respectively. In simulations, $\hat{\text{MSE}}$ is usually compared with the MSE reference to examine the accuracy of the estimation. The MSE reference matrix $\mathbf{\Omega_{MSE}}$ ($N_d \times N_d$) is given by [50]

$$\mathbf{\Omega_{MSE}} = \mathbf{I_{N_h}} - \mathbf{wR_{dp}^H} \tag{6.16}$$

Then the reference MSE value is obtained by taking the average of the diagonal elements of $\mathbf{\Omega_{MSE}}$.

$$\text{MSE} = 1/N_d \sum_{i=1}^{N_d} \mathbf{\Omega_{MSE}}(i,i) \tag{6.17}$$

The actual MSE will overlap with the estimated MSE in perfect Wiener filter channel estimation.

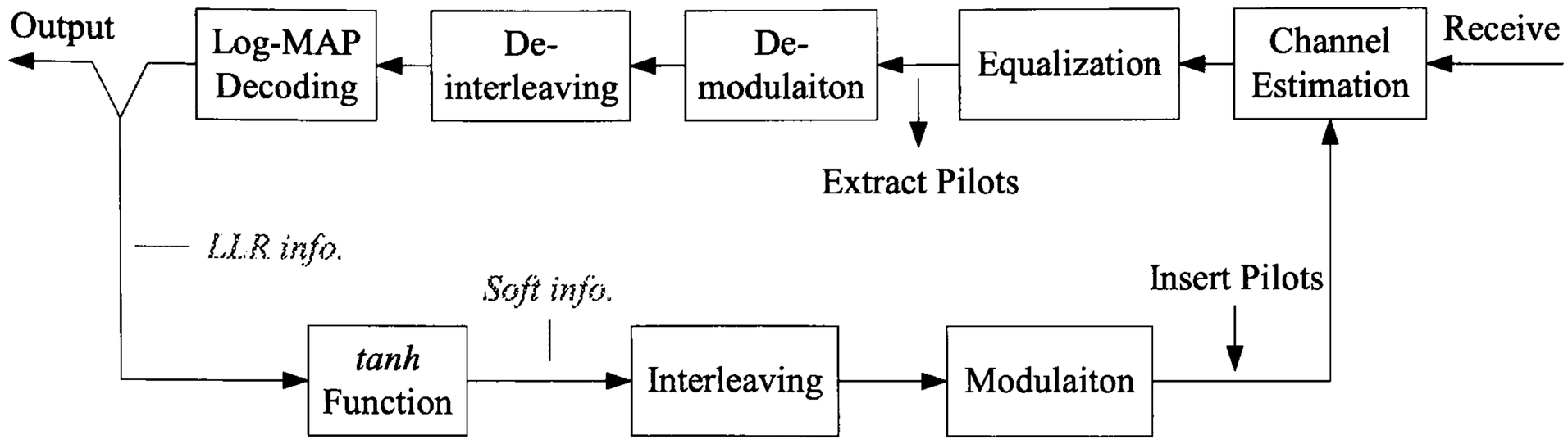


Figure 6.4: Receiver structure of code-aided iterative channel estimation and signal detection

6.2.2 Code-aided Iterative Channel Estimation by Wiener Filtering

Having looked at the pilot-based channel estimation by Wiener filtering, this section considers adding outer coding and exploring the benefit from the iterative channel estimation. The use of the Wiener filter for iterative channel estimation over flat fading channels was proposed in [51]. Here, this approach is applied to IEEE 802.16 OFDMA systems. Fig. 6.4 illustrates a simplified code-aided iterative receiver structure cooperated by the channel estimation and signal detection (Log-MAP decoder). The iterative process is described as follows.

Firstly, the channel estimates are obtained from Wiener filtering, either 2D or 2×1D, based on the pre-arranged pilots. Then the equalized data pass through the reverse processes of those at the transmitter end. In the case of QPSK, demodulation is simply done by separating the real and imaginary parts. The demodulated signals are then sent to the SISO Log-MAP decoder, the output of which contains the LLR of data bits as well as coded parity bits, so as to regenerate the coded data. Finally, a hard decision is made on the soft data bits. Such non-iterative estimations are not able to achieve enough accuracy when the number of pilots is limited.

However, it is possible to improve the estimator by exploring the useful information from the estimates. In this approach, the soft information of coded parity bits (taking *tanh* function over the half of the LLR of the parity bits) are fed back and passed through the

interleaver and modulator (same as those at the transmitter), and allocated in the original positions. These soft data symbols from the decoder, together with the original pilots, are all regarded as the 'known' pilot symbols in the next iteration. The benefit is that the performance can be enhanced without adding additional pilots. This estimation and decoding process are repeated for several iterations. After each iteration, the estimator uses the improved soft data bits as pilots to obtain more accurate estimates.

The Wiener filter interpolation equation and the MSE reference matrix for the iterative case becomes

$$\hat{\mathbf{h}} = \mathbf{R_{pp}}(\mathbf{R_{pp}} + \frac{\beta}{\mathrm{SNR}}\mathbf{I_{(N_p+N_d)}})^{-1}\hat{\mathbf{h}}_\mathbf{p} \tag{6.18}$$

$$\mathbf{MSE} = \mathbf{I_{(N_d+N_p)}} - \mathbf{wR_{pp}^H} \tag{6.19}$$

where $\hat{\mathbf{h}}_\mathbf{p}$ are the channel LS estimates at all positions and $\mathbf{R_{pp}}$ are channel auto-correlations between all pilot and data symbols or subcarriers.

Although a price has to be put on the receive complexity, it is still worth to carry out iterative estimation, since it can achieve a significant gain over the non-iterative case, provided the Wiener filter has a large number of taps in iterative steps.

6.2.3 Code-aided Channel Tracking Based on EM Algorithm

The above channel estimation is designed to estimate a channel varying both in time and frequency with pilots scattered in OFDM symbols and subcarriers. It is noticed that in OFDMA DL sub-frame (Fig. 6.2), a preamble symbol, designed for various estimation purposes, is transmitted. It is advantageous to make use of the preamble symbol in the DL channel estimation.

The channel tracking based on the preamble is especially suitable for estimating a channel that remains constant or varies very slowly between OFDM symbols. In other words,

the time fading of the channel is negligible, which greatly simplifies the channel estimation problem. Moreover, the latency caused by channel estimation is reduced, from several OFDM symbols in the pilots-based scheme to one OFDM symbol (if the length of preamble is one symbol). This is particularly attractive for a packet-type transmission system, where data is preferred to be recovered in real time before the entire burst has been received [3].

In this section, a code-aided channel tracking scheme based on the EM algorithm [52] is applied, which iterates between the channel tracker and data decoder [47]. Different from Winer filtering, channel tracking is based solely on the initial estimates; therefore no pre-knowledge of channel statistic information is needed.

6.2.3.1 EM algorithm

EM algorithm is an iterative technique to find the ML estimates of parameters θ from an observation y [52]. The model depends on the unobserved variables x: parameters θ can be estimated if the unobserved data x is known. The observed data y and the unobserved data x are jointly called the complete data. EM algorithm performs alternately between an expectation (E) step and a maximization (M) step.

The 'E step' computes an expectation (with regard to the unobserved data x) of the joint conditional log-likelihood[5] of the complete data x and y under parameter θ, given the observed variables y and the parameter estimates from the current iteration $\hat{\theta}^{(iter)}$. The expectation is denoted as $Q(\theta|\hat{\theta}^{(iter)})$.

$$Q(\theta|\hat{\theta}^{(iter)}) = E_x[\log p(y, x|\theta)|y, \hat{\theta}^{(iter)}] \tag{6.20}$$

Note that $\hat{\theta}^{(iter)}$ is the current estimates used to calculate the expectation, while θ is an variable ultimately optimized to maximize Q in the next step. x is a variable, the prob-

[5]log-likelihood is often used instead of the true likelihood because it leads to easier formulas, while still attains its maximum at the same point as the likelihood

ability of which is given by $p(x|y,\hat{\theta}^{(iter)})$. Therefore, the right part of Eq. 6.20 can be written in the continuous case as

$$E_x[\log p(y,x|\theta)|y,\hat{\theta}^{(iter)}] = \int_{x\in\gamma} \log p(y,x|\theta)\cdot p(x|y,\hat{\theta}^{(iter)})d_x \tag{6.21}$$

where $p(x|y,\hat{\theta}^{(iter)})$ is the marginal probability of unobserved data x depending on the observed data y and the parameter estimates from the current iteration $\hat{\theta}^{(iter)}$; γ is the range of x. For the first iteration, the initial estimate $\hat{\theta}^{(0)}$ is decided independent of the algorithm.

Now consider more about the right part of Eq. 6.21. According to the Bayer's formula,

$$p(y,x|\theta) = p(y|x,\theta)\cdot p(x|\theta) \tag{6.22}$$

Since the distribution of x does not depend on the parameters θ to be estimated, Eq. 6.22 can be simplified to

$$p(y,x|\theta) = p(y|x,\theta)\cdot p(x) \tag{6.23}$$

Substitute Eq. 6.23 to Eq. 6.21:

$$\begin{aligned} Q(\theta|\hat{\theta}^{(iter)}) = &\int_{x\in\gamma} \log p(y|x,\theta)\cdot p(x|y,\hat{\theta}^{(iter)})d_x \\ &+ \int_{x\in\gamma} \log p(x)\cdot p(x|y,\hat{\theta}^{(iter)})d_x \end{aligned} \tag{6.24}$$

Again, the second term of the right part of Eq. 6.24 does not depend on θ. As far as the next 'M step' is concerned, this term can be dropped. Therefore, the the 'E step' can be finally written as

$$Q(\theta|\hat{\theta}^{(iter)}) = \int_{x\in\gamma} \log p(y|x,\theta)\cdot p(x|y,\hat{\theta}^{(iter)})d_x \tag{6.25}$$

The 'M step' computes the ML estimates of the parameters θ by maximizing the expected likelihood found on the 'E step'.

$$\hat{\theta}^{(iter+1)} = \arg\max_{\theta} Q(\theta|\hat{\theta}^{(iter)}) \tag{6.26}$$

This step iteratively improves the previous estimate $\hat{\theta}^{(iter)}$ and constructs new estimates $\hat{\theta}^{(iter+1)}$, which are used to begin another 'E step'. The process are repeated until the estimate has converged or met a certain stopping criterion.

6.2.3.2 Channel Frequency Response Tracking in OFDM System

Apply the EM algorithm in its discrete expression to an OFDM system for estimating the frequency response of the multipath channel [47]. The channel is assumed to be constant during one OFDM symbol and vary slowly from one symbol to another.

Suppose the size of an OFDM symbol is N_s. Denote the transmitted symbols as S and write them orderly onto the diagonal positions of a matrix $\mathbf{S}$ ($N_s \times N_s$). The received symbols and the channel frequency responses are written in the vector form as $\mathbf{R}$ ($N_s \times 1$) and $\mathbf{H}$ ($N_s \times 1$). $\mathbf{S}$, $\mathbf{R}$ and $\mathbf{H}$ respectively correspond to the unobserved data x, the observed data y and the parameter θ in the EM algorithm.

Therefore, the 'E step' becomes

$$Q(\mathbf{H}|\hat{\mathbf{H}}^{(iter)}) = \sum_{\mathbf{S}} \log p(\mathbf{R}|\mathbf{S},\mathbf{H}) \cdot p(\mathbf{S}|\mathbf{R},\hat{\mathbf{H}}^{(iter)}) \tag{6.27}$$

The noise samples $\mathbf{N}$ ($N_s \times 1$), which are the only unknown parameter at the receiver, is assumed to be zero-mean AWGN with a variance of σ_n^2. Given that $\mathbf{R} = \mathbf{SH} + \mathbf{N}$,

$$\begin{aligned} \log p(\mathbf{R}|\mathbf{S},\mathbf{H}) &\propto -\|\mathbf{R}-\mathbf{SH}\|^2 \\ &= -(\mathbf{R}-\mathbf{SH})^{\mathrm{H}}(\mathbf{R}-\mathbf{SH}) \\ &\propto -\mathbf{H}^{\mathrm{H}}\mathbf{S}^{\mathrm{H}}\mathbf{SH} + \mathbf{H}^{\mathrm{H}}\mathbf{S}^{\mathrm{H}}\mathbf{R} + \mathbf{R}^{\mathrm{H}}\mathbf{SH} \end{aligned} \tag{6.28}$$

Substituting Eq. 6.28 into Eq. 6.27, the 'E step' is expressed as

$$Q(\mathbf{H}|\hat{\mathbf{H}}^{(iter)}) \propto \sum_{\mathbf{S}} (-\mathbf{H}^{\mathrm{H}}\mathbf{S}^{\mathrm{H}}\mathbf{SH} + \mathbf{H}^{\mathrm{H}}\mathbf{S}^{\mathrm{H}}\mathbf{R} + \mathbf{R}^{\mathrm{H}}\mathbf{SH}) \cdot p(\mathbf{S}|\mathbf{R},\hat{\mathbf{H}}^{(iter)}) \tag{6.29}$$

Denote S_k ($k = 1, 2 \ldots N_s$) as the hard decisions of symbols on N_s subcarriers. The k-th diagonal element of $\mathbf{S}$ or $\mathbf{S}^{\mathrm{H}}\mathbf{S}$ depends only on S_k and its posteriori probabilities $p(S_k|\mathbf{R},\hat{\mathbf{H}}^{(iter)})$. Moreover, the soft information of $\mathbf{S}$, $\mathbf{S}^{\mathrm{H}}$ and $\mathbf{S}^{\mathrm{H}}\mathbf{S}$ are obtained from the posteriori probabilities:

$$\tilde{\mathbf{S}} = \sum_{\mathbf{S}} \mathbf{S} \cdot p(\mathbf{S}|\mathbf{R},\hat{\mathbf{H}}^{(iter)}) \tag{6.30}$$

$$\tilde{\mathbf{S}}^{\mathrm{H}} = \sum_{\mathbf{S}} \mathbf{S}^{\mathrm{H}} \cdot p(\mathbf{S}|\mathbf{R}, \hat{\mathbf{H}}^{(iter)}) \tag{6.31}$$

$$\widetilde{\mathbf{S}^{\mathrm{H}}\mathbf{S}} = \sum_{\mathbf{S}} \mathbf{S}^{\mathrm{H}}\mathbf{S} \cdot p(\mathbf{S}|\mathbf{R}, \hat{\mathbf{H}}^{(iter)}) \tag{6.32}$$

Therefore, the 'E step' can be written as

$$Q(\mathbf{H}|\hat{\mathbf{H}}^{(iter)}) \propto -\mathbf{H}^{\mathrm{H}}\widetilde{\mathbf{S}^{\mathrm{H}}\mathbf{S}}\mathbf{H} + \mathbf{H}^{\mathrm{H}}\tilde{\mathbf{S}}^{\mathrm{H}}\mathbf{R} + \mathbf{R}^{\mathrm{H}}\tilde{\mathbf{S}}\mathbf{H} \tag{6.33}$$

Note that $\tilde{\mathbf{S}}$, $\tilde{\mathbf{S}}^{\mathrm{H}}$ and $\widetilde{\mathbf{S}^{\mathrm{H}}\mathbf{S}}$ are $N_s \times N_s$ diagonal matrices with real numbers of $\sum_{S_k} S_k \cdot p(S_k|\mathbf{R}, \hat{\mathbf{H}}^{(iter)})$, $\sum_{S_k} S_k^{\mathrm{T}} \cdot p(S_k|\mathbf{R}, \hat{\mathbf{H}}^{(iter)})$ and $\sum_{S_k} |S_k|^2 \cdot p(S_k|\mathbf{R}, \hat{\mathbf{H}}^{(iter)})$ ($k = 1, 2 \ldots N_s$) on their diagonal elements, respectively.

To obtain the estimate of $\mathbf{H}$, $Q(\mathbf{H}|\hat{\mathbf{H}}^{(iter)})$ is maximized by firstly taking the derivative of $Q(\mathbf{H}|\hat{\mathbf{H}}^{(iter)})$ with regarding to $\mathbf{H}$ and then set the results to zeros. The derivative step requires some vector/matrix calculations[6], and the results are given by

$$\frac{\partial(-\mathbf{H}^{\mathrm{H}}\widetilde{\mathbf{S}^{\mathrm{H}}\mathbf{S}}\mathbf{H})}{\partial \mathbf{H}} = -[\widetilde{\mathbf{S}^{\mathrm{H}}\mathbf{S}} + (\widetilde{\mathbf{S}^{\mathrm{H}}\mathbf{S}})^{\mathrm{H}}]\mathbf{H} \tag{6.34}$$

$$\frac{\partial(\mathbf{H}^{\mathrm{H}}\tilde{\mathbf{S}}^{\mathrm{H}}\mathbf{R} + \mathbf{R}^{\mathrm{H}}\tilde{\mathbf{S}}\mathbf{H})}{\partial \mathbf{H}} = \tilde{\mathbf{S}}^{\mathrm{H}}\mathbf{R} + \tilde{\mathbf{S}}^{\mathrm{H}}\mathbf{R} \tag{6.35}$$

Since

$$\widetilde{\mathbf{S}^{\mathrm{H}}\mathbf{S}} = (\widetilde{\mathbf{S}^{\mathrm{H}}\mathbf{S}})^{\mathrm{H}} \tag{6.36}$$

the derivative of $Q(\mathbf{H}|\hat{\mathbf{H}}^{(iter)})$ is obtained:

$$\frac{\partial Q}{\partial \mathbf{H}} \propto -2\widetilde{\mathbf{S}^{\mathrm{H}}\mathbf{S}}\mathbf{H} + 2\tilde{\mathbf{S}}^{\mathrm{H}}\mathbf{R} \tag{6.37}$$

Then, let the right part of Eq. 6.37 equal zeros. The expression for the estimates of $\mathbf{H}$ in the $(iter{+}1)$-th iteration can be obtained:

$$\hat{\mathbf{H}}^{(iter+1)} = [(\widetilde{\mathbf{S}^{\mathrm{H}}\mathbf{S}})^{-1}\tilde{\mathbf{S}}^{\mathrm{H}}]^{(iter)}\mathbf{R} \tag{6.38}$$

For a constellation with constant modulus C, e.g., QPSK, $\widetilde{\mathbf{S}^{\mathrm{H}}\mathbf{S}}$ becomes $C\mathbf{I}$, where $\mathbf{I}$ is the identity matrix. Therefore, Eq. 6.38 can be effectively simplified to

$$\hat{\mathbf{H}}^{(iter+1)} = \frac{1}{C}[\tilde{\mathbf{S}}^{\mathrm{H}}]^{(iter)}\mathbf{R} \tag{6.39}$$

[6]Here, the following two matrix derivatives in Eq. 6.34 and Eq. 6.35 are used respectively: $\frac{\partial(\mathbf{x}^{\mathrm{H}}\mathbf{B}\mathbf{x})}{\partial \mathbf{x}} = (\mathbf{B} + \mathbf{B}^{\mathrm{H}})\mathbf{x}$, where $\mathbf{x}$ is a vector and $\mathbf{B}$ is a matrix; $\frac{\partial(\mathbf{x}^{\mathrm{H}}\mathbf{a})}{\partial \mathbf{x}} = \frac{\partial(\mathbf{a}^{\mathrm{H}}\mathbf{x})}{\partial \mathbf{x}} = \mathbf{a}$, where both $\mathbf{x}$ and $\mathbf{a}$ are vectors.

From the above algorithms, it is clear that with the help from each other, the decoder and estimator iteratively improve their soft coded symbols and estimates. The process of the code-aided channel tracking is described as follows.

1. At first, the LS channel estimates of the preamble symbol are regarded as the initial channel estimates $\hat{\mathbf{H}}^{(0)}$.

2. The received symbols $\mathbf{R}$ are then equalized using these initial estimates and decoded by the SISO Log-MAP decoder, where the soft coded symbols $\tilde{\mathbf{S}}^{(0)}$ are generated.

3. Finally, substituting $\tilde{\mathbf{S}}^{(0)}$ into Eq. 6.39, a new set of channel estimates $\hat{\mathbf{H}}^{(1)}$ are obtained.

This process are repeated until the iteration ends assuming the number of iterations I is set at the beginning, or if denote the final channel estimates as $\hat{\mathbf{H}}^{(I)}$, when $|\hat{\mathbf{H}}^{(I)} - \hat{\mathbf{H}}^{(0)}|^2$ reaches a negligible amount. $\hat{\mathbf{H}}^{(I)}$ are regarded as the initial channel estimates for the channel tracking of the next OFDM symbol.

Note that the optimum of the proposed channel tracking are strongly decided by the initial estimates $\hat{\mathbf{H}}^{(0)}$. Since it is assumed that the channel varies slowly between OFDM symbols, it is reasonable to set $\hat{\mathbf{H}}^{(0)}_{(l)} = \hat{\mathbf{H}}^{(I)}_{(l-1)}$, where l are the OFDM symbol indices.

In case scattered pilot symbols present within the OFDM symbol, they will be perfectly included in the soft coded symbols at their positions (the probability of decoding at these positions will be 1). Moreover, within each iteration, it is possible to apply Wiener filtering using these pilots to further refine the estimates from channel tracking.

6.2.4 Performance of Channel Estimation in OFDMA

This section provides simulation results of the performance of non-iterative and iterative channel estimation in IEEE 802.16 OFDMA UL PUSC/OPUSC and DL FUSC/PUSC modes. Also, the performance degradation due to channel estimation is compared with the one with perfectly known channels. Depending on the sub-frame structure and pilot allocation, an appropriate and efficient channel estimation scheme is chosen for DL and UL separately.

The system works at a carrier frequency of 3 GHz, selected as the middle point in the 2-6 GHz licensed frequency range. The OFDMA symbol has 1024 subcarriers with a total bandwidth of 10 MHz. The channel models used in the simulation are recommended by International Telecommunications Union - Radiocommunications (ITU-R) for typical test environment. Coding and interleaving are specified in the IEEE 802.16 physical layer standard [2]. QPSK modulation scheme are used for both data and pilots throughout the simulations. Note that in UL OPUSC and DL transmission, each pilot shall be transmitted with a boosting of 2.5 dB over the average power of each data tone.

6.2.4.1 ITU Channel Modeling and Standard Coding

Channel models recommended by ITU-R [5] are adopted for simulations of IEEE 802.16 physical layer OFDMA. The ITU-R reference channel models are intended for various test environment in International Mobile Telecommunications-2000 (IMT-2000), a global standard defined by ITU for 3G wireless communications. Here, this section will give a brief description of its channel impulse response model and time-varying property at various moving speed.

Table 6.2 gives the tapped delay-line parameters for three typical types of the ITU-R terrestrial test environments: 'B model' for indoor use (ITU-IB) at a moving speed of 1

Indoor-B		Pedestrian-B		Vehicular-A	
Delay (ns)	Power (dB)	Delay (ns)	Power (dB)	Delay (ns)	Power (dB)
0	0	0	0	0	0
100	-3.6	200	-0.9	310	-1.0
200	-7.2	800	-4.9	710	-9.0
300	-10.8	1200	-8.0	1090	-10.0
500	-18.0	2300	-7.8	1730	-15.0
700	-25.2	3700	-23.9	2510	-20.0
Moving Speed (km/h)		Moving Speed (km/h)		Moving Speed (km/h)	
1		3		60	

Table 6.2: ITU-R reference channel model tapped delay-line parameters [5]

km/h, 'B model' for pedestrian use (ITU-PB) at 3 km/h and 'A model' for vehicular use (ITU-VA) at 60 km/h. For each tap of the channel, two parameters are given: the time delay relative to the first tap and the average power relative to the strongest tap. Each tap has a Jake's Doppler spectrum, the Doppler shift can be calculated according to the carrier frequency and the relative moving speed between BS and SS.

For the FFT-1024 OFDMA with a bandwidth of 10 MHz, the sampling interval is 87.56 ns, which is the inverse of the sampling frequency of 11.429 MHz given in Table 6.1. Fig. 6.5, 6.6 and 6.7 present the power delay profile, the rms delay spread, the relative delay of each tap ('o' marked on the power delay profile) and the OFDMA sampling points of ITU-IB, ITU-PB and ITU-VA channel models respectively.

The figures show that ITU-IB, ITU-PB and ITU-VA have an approximately exponential decaying power delay profile. In all three channel models, the relative time delay of each channel tap do not exactly match the multiple of the OFDM sampling intervals. These non-sample-spaced channel models will increase the complexity in simulations. It is noticed that the ITU-PB and ITU-VA channels have extremely long maximum delays, compared to the OFDMA sampling interval; hence it is reasonable to find a sampling time

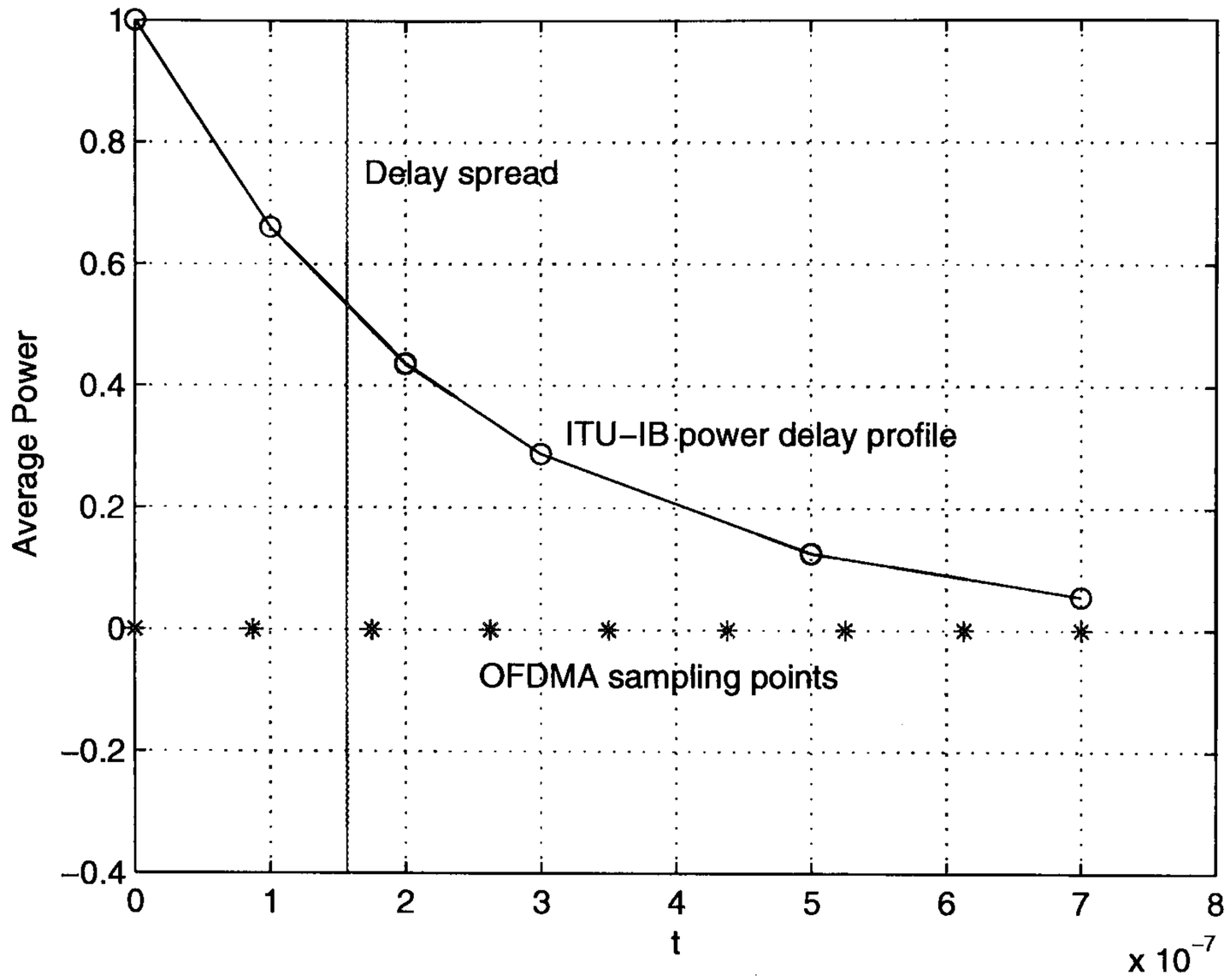


Figure 6.5: Power delay profile of ITU-IB channel model

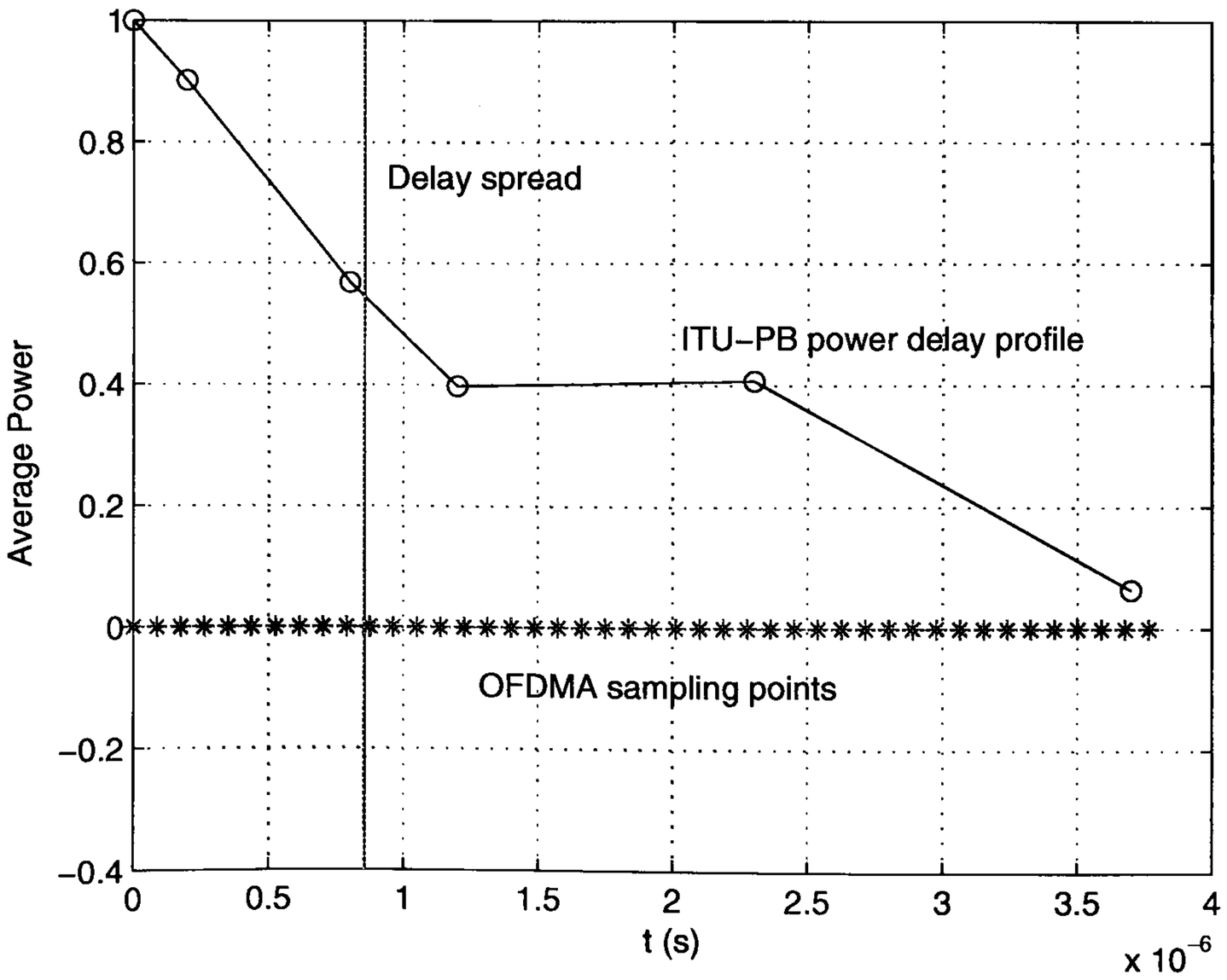


Figure 6.6: Power delay profile of ITU-PB channel model

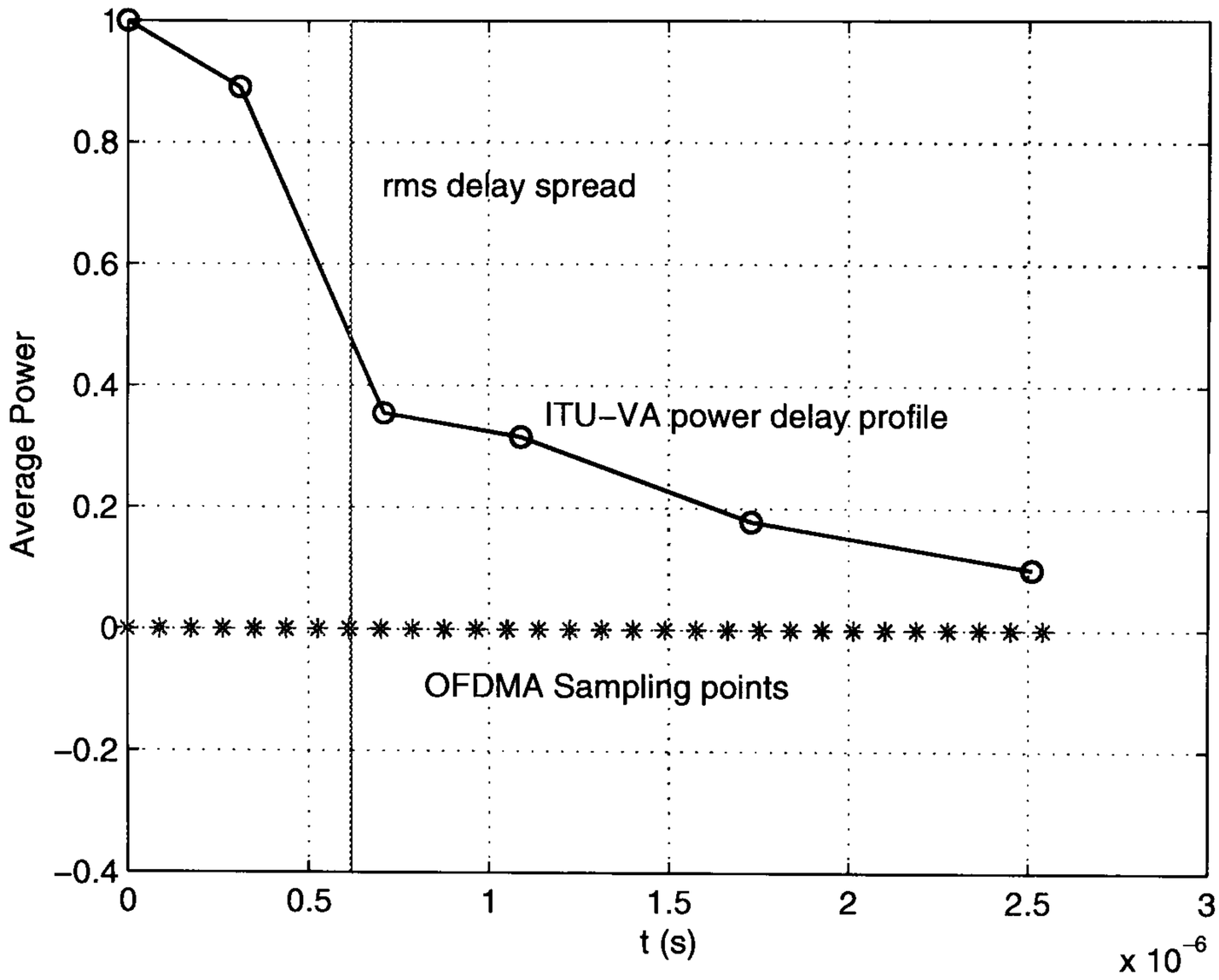


Figure 6.7: Power delay profile of ITU-VA channel model

closest to the time delay of each tap and use this sampling time as the new tap delay. This approximation will not cause much difference in the performance, but will simplify the simulation process greatly.

However, unlike ITU-PB and ITU-VA, ITU-IB channel has a maximum delay that is comparable to the OFDMA sampling interval. If similarly, the tap is moved to the nearest sample points, it may cause inaccuracy in the simulation results, especially for those taps fall in the middle of two sampling points. For this particular case, it is considered to double the OFDM sampling rate so that the new sampling points are twice closer to each other. Then the nearest sampling points are chosen as the new tap delays. Doubling sampling rate is realized by inserting zeros, the number of which equals the FFT size, in the middle of the IFFT block. The size of IFFT is doubled and the outputs are samples sampled at twice the original rate.

A rate 1/2, constraint length 7 convolutional code is specified in the IEEE 802.16 physical layer standard [2]. The generator polynomial is (171,133). The interleaver after coding

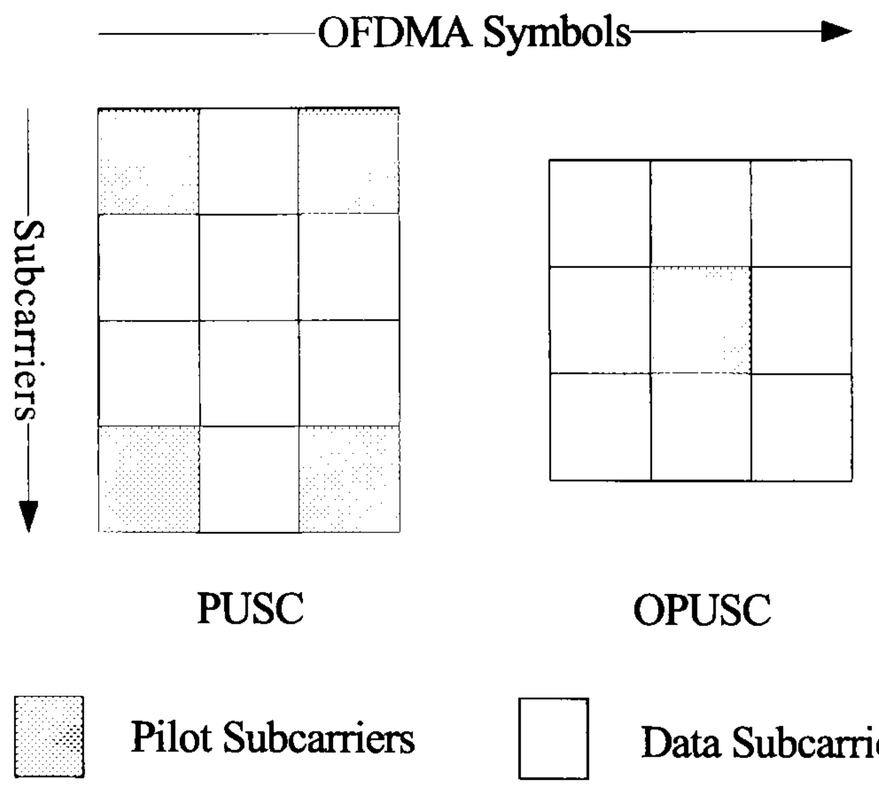


Figure 6.8: UL PUSC and OPUSC tile structure

and the de-interleaver before decoding are also described in the standard [2]. Besides, before interleaving, an optional N-time repetition coding is applied to the code bits to gain more frequency diversity. The repetition coding is simply done by repeating each code bit N_r times. For example, code sequence '1 0 1...' becomes '1 1 0 0 1 1...' after twice repetition coding. In our simulations, N_r=2 in all cases to obtain the benefit of diversity.

6.2.4.2 UL PUSC and OPUSC

The UL PUSC and OPUSC sub-frames are inserted with scattered pilot symbols across OFDMA symbols and subcarriers. The pilot subcarrier allocations in both modes use a tile structure, but with different pilot positions and number of pilots in a tile. Fig. 6.8 illustrates the tile structures of UL PUSC and OPUSC. In PUSC, a tile spans three symbols and four subcarriers with four pilots standing in four corners of the tile, while in OPUSC, a tile has a dimension of three symbols and three subcarriers with one pilot inserted in the middle of the tile. For this type of pilot allocations, a Wiener filter based channel estimation will be appropriate. Note that the size of the UL PUSC and OPUSC sub-frame must be a multiple of three symbols across time.

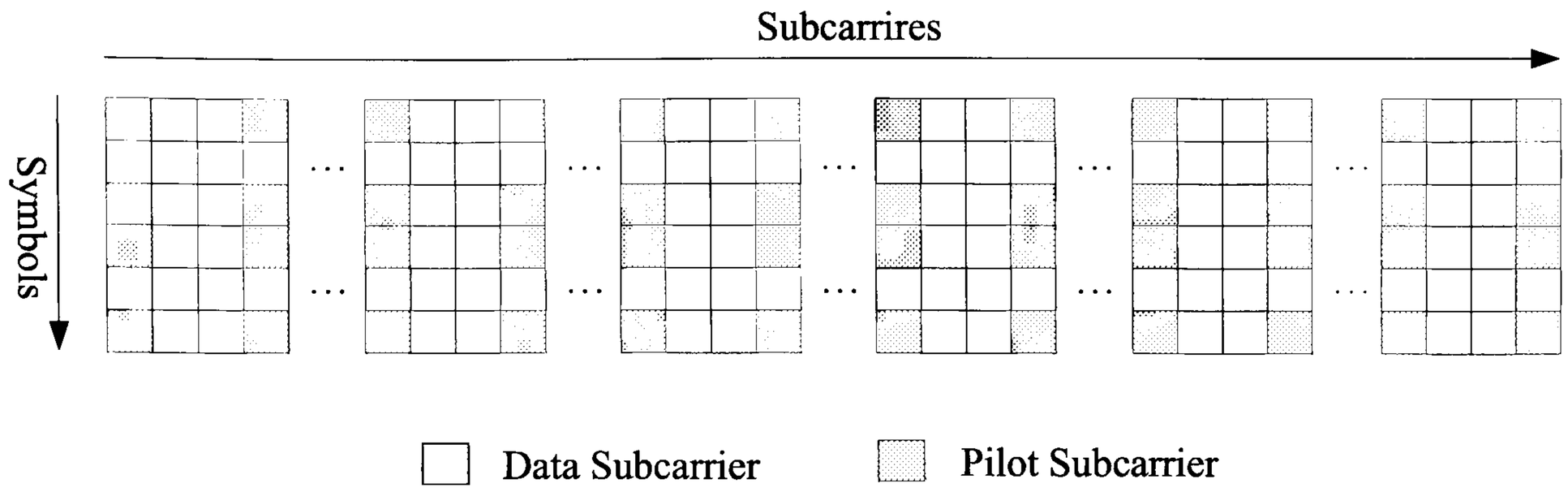


Figure 6.9: Schematics of two sub-channels for one user in UL PUSC

In a total of 1024 subcarriers, the uplink PUSC/OPUSC mode supports 35/48 sub-channels where each transmission uses 48 data subcarriers as the minimal block of processing. A burst in the uplink is composed of three symbols and one sub-channel containing six tiles. Within each burst, there are 48/48 data subcarriers and 24/6 fixed location pilot subcarriers in the PUSC/OPUSC mode. Therefore the used frequency band shall be partitioned into 210/288 tiles. After data and pilots are allocated within the tiles, a tile permutation mechanism across frequency is applied to tiles in a sub-channel. This is to provide frequency diversity by distributing tiles over all frequency bands, such that the performance degradation due to fading infrequency of mobile environments is minimized [4].

It is assumed that for both UL PUSC and OPUSC, each user is assigned two sub-channels that are composed of six distributed tiles across frequency and six continuous symbols in time (Fig. 6.9). Note that the tile permutation scheme only distributes tiles across frequency subcarriers, but not between two tiles adjacent in time. Within each tile, data and pilot subcarriers still remain continuous. In this case, the largest number of available pilots for channel estimation is limited. This is because the pilots are far apart across the whole bandwidth and the correlations between them are negligible. Moreover, pilots in frequency-adjacent tiles are related to different users, and hence will correspond to a different channel in the UL transmission. Therefore, the channel estimation can only be performed within two adjacent tiles across OFDMA symbols.

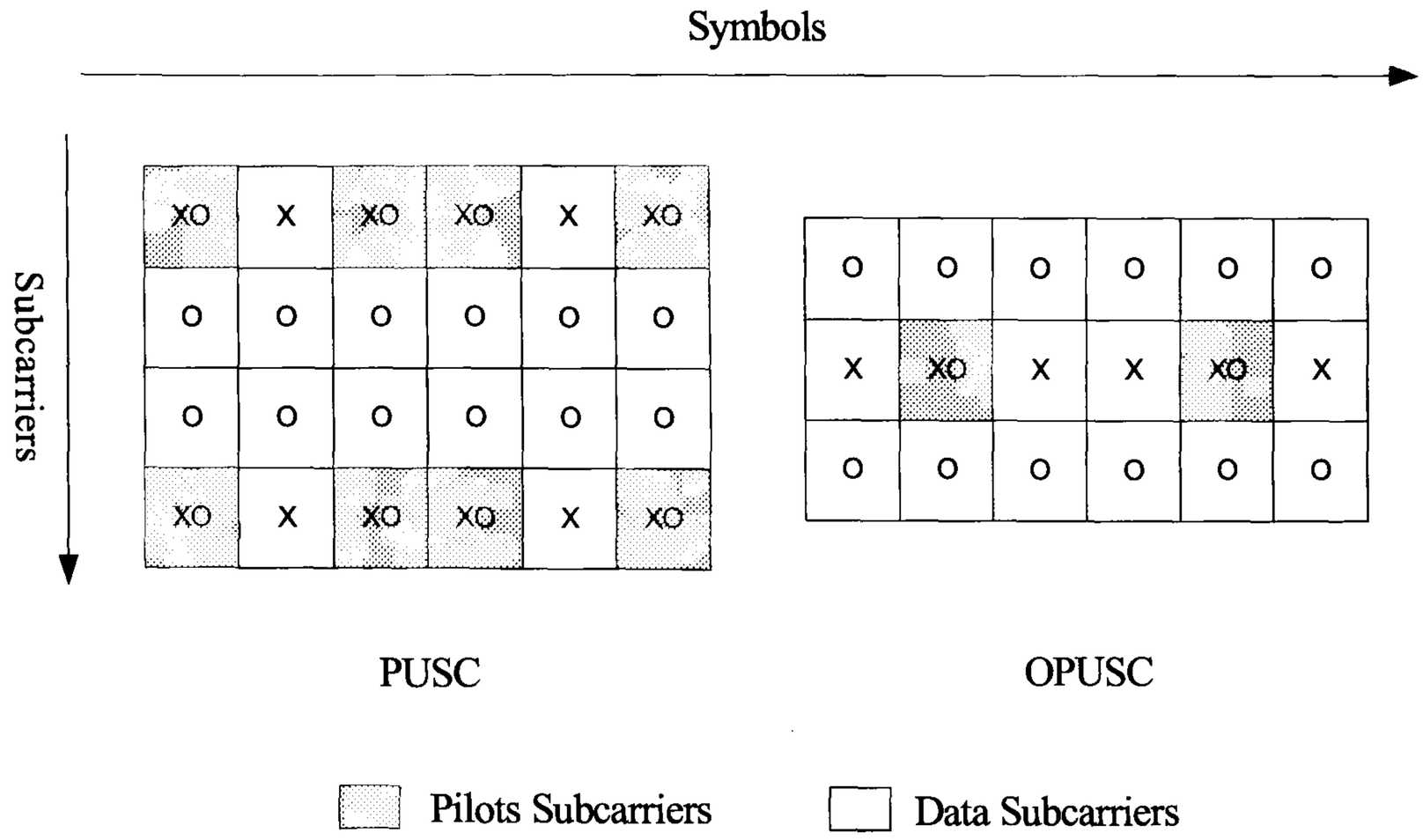


Figure 6.10: Channel estimation block in UL PUSC and OPUSC

The largest estimation blocks of UL PUSC and OPUSC are shown in Fig. 6.10. In PUSC, the channel estimation can be applied by either 2×1D or 2D approach. For 2×1D estimation, a 4-tap Wiener filter is first interpolated using time correlations to obtain time estimates ('×'s), including the estimates at pilot positions; then a 2-tap Wiener filter is applied across frequency to get frequency estimates ('○'s). Alternatively, it is possible to obtain all the estimates (including those at pilot positions) at once using an 8-tap 2D channel estimator. Similarly for OPUSC, a 1D estimation across time with a 2-tap Wiener filter ('×'s) is applied. However, the estimation across frequency are not necessary, since there is only one pilot available for estimation across frequency, which is not helpful. Therefore, the estimates of the two side rows of subcarriers ('○'s) are just replicas of the time estimates in the middle ('×'s).

Next part will present simulation results regarding the BER and MSE performance of channel estimation in OFDMA UL PUSC and OPUSC, both in non-iterative and iterative cases, in various channel environments.

Fig. 6.11, 6.12 and 6.13 show the BER and MSE performance of 2D estimation in UL PUSC in ITU-IB, ITU-PB and ITU-VA channel respectively. It is seen from the BER

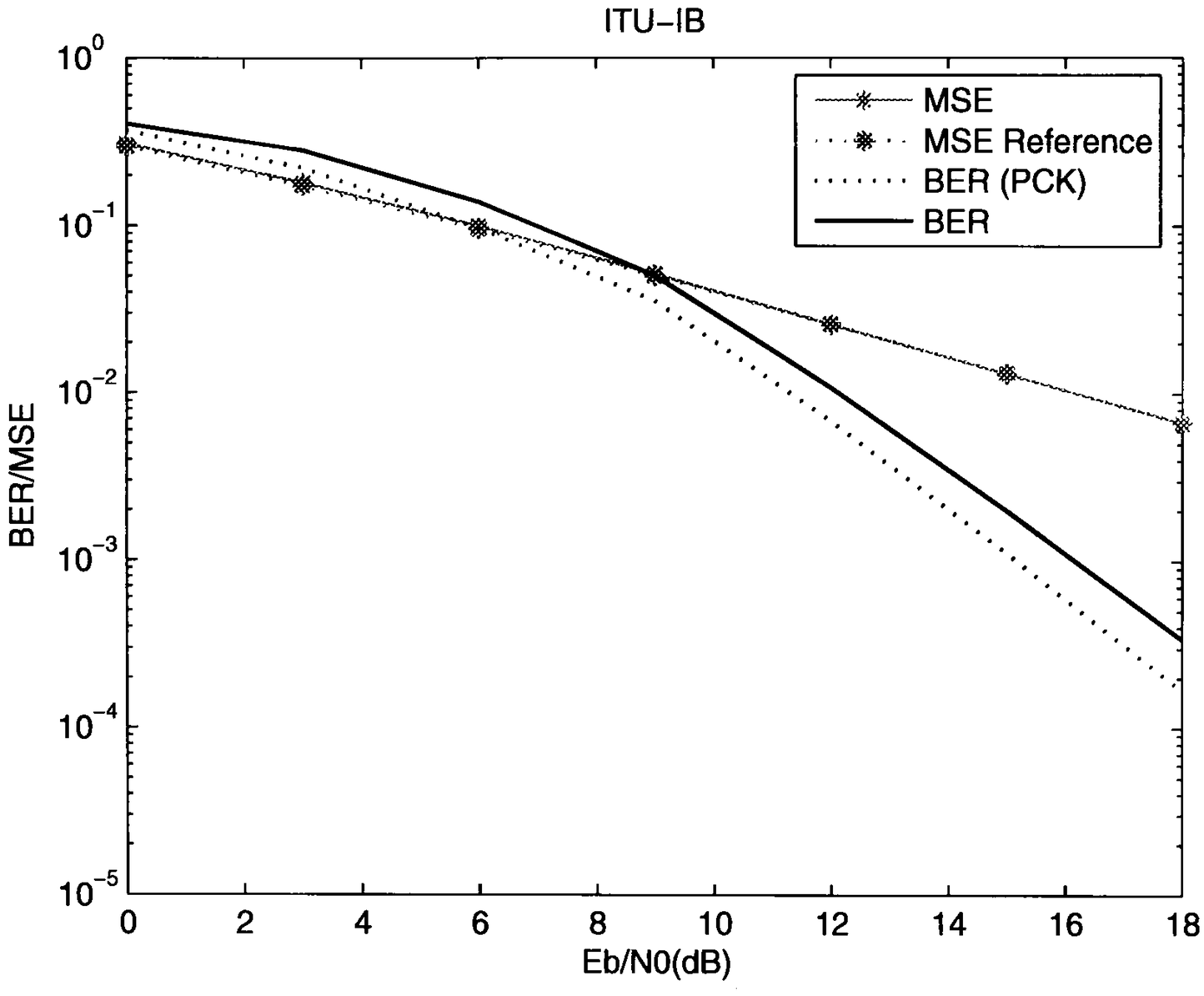


Figure 6.11: BER and MSE performance of 2D channel estimation in UL PUSC (ITU-IB)

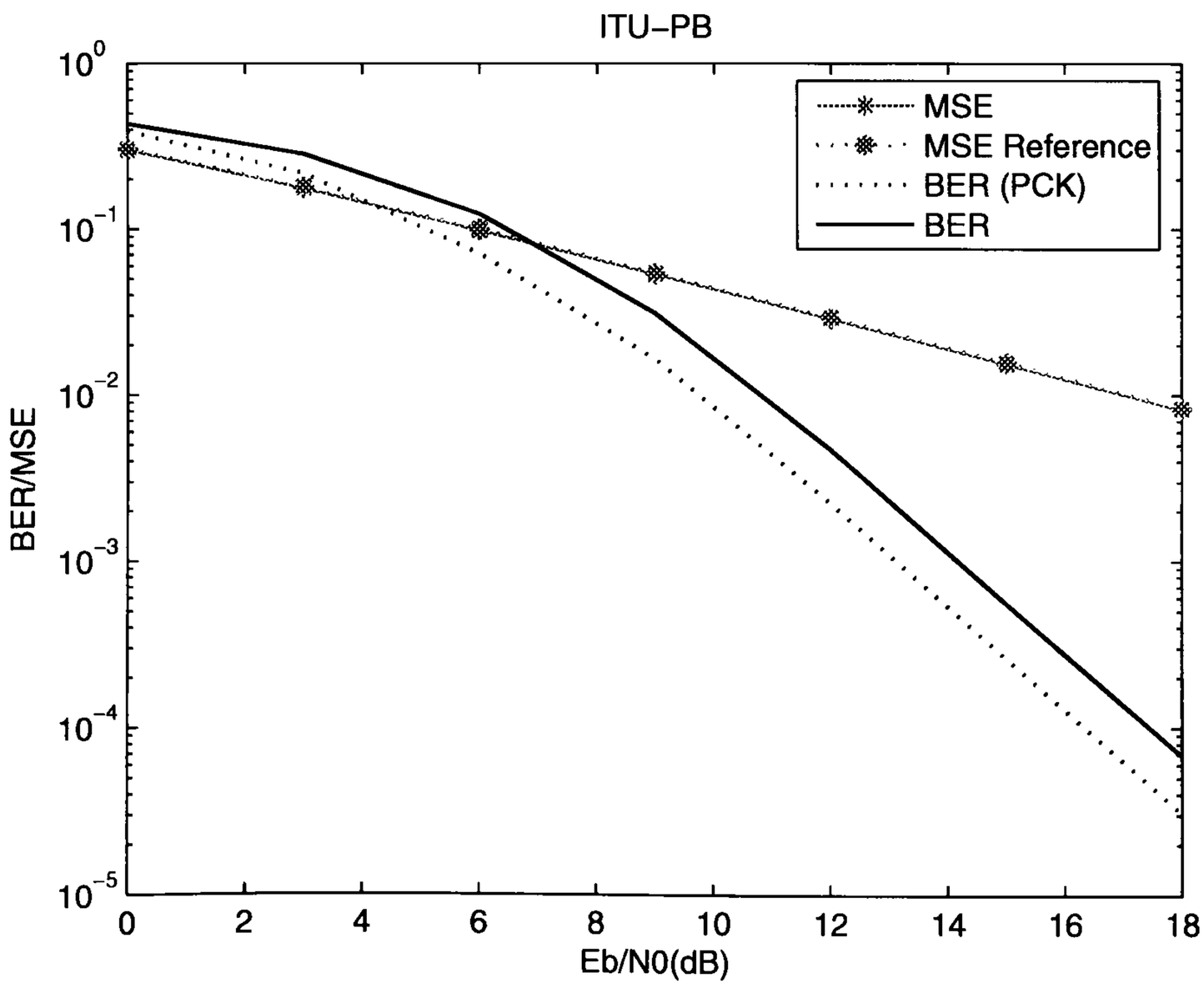


Figure 6.12: BER and MSE performance of 2D channel estimation in UL PUSC (ITU-PB)

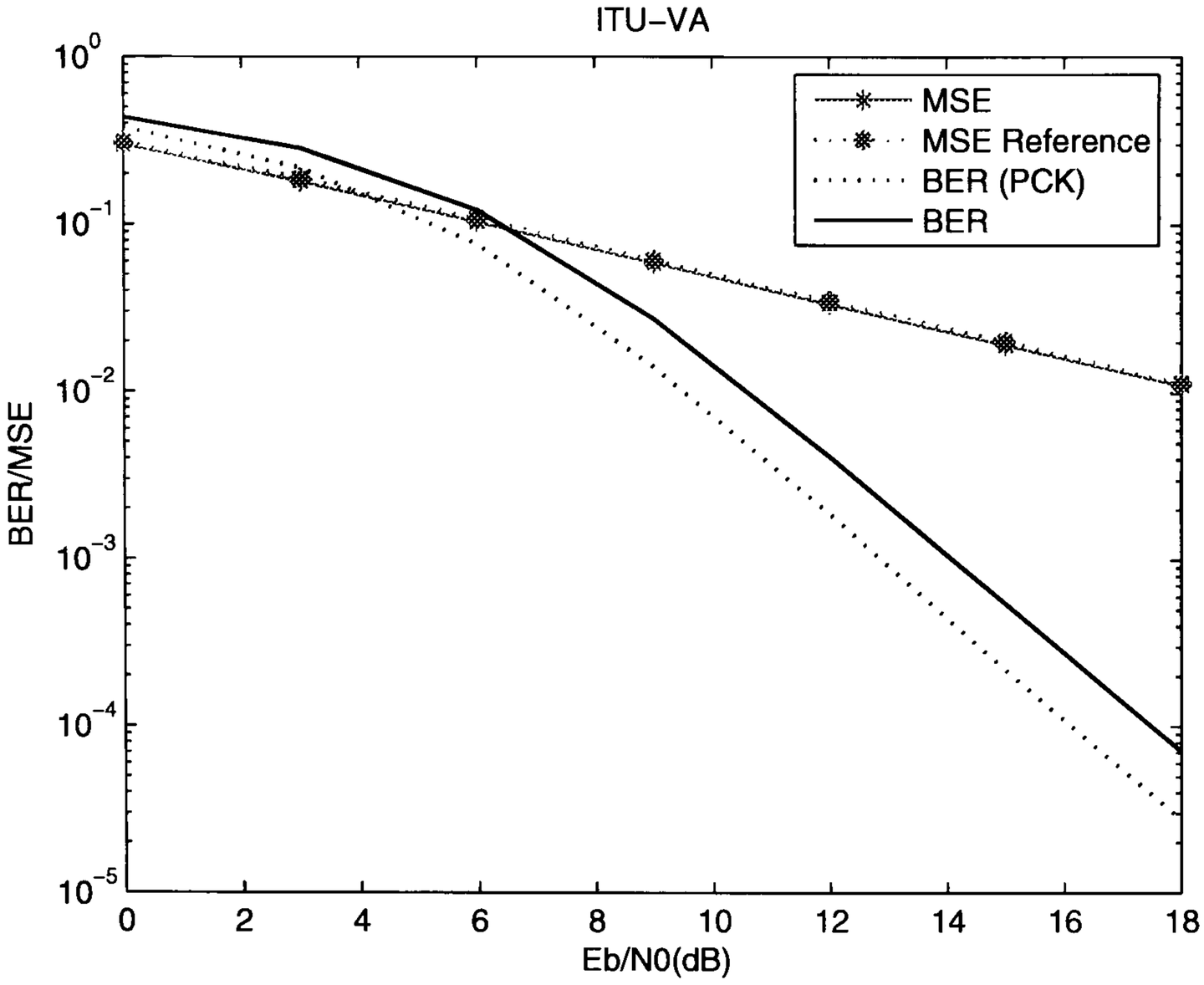


Figure 6.13: BER and MSE performance of 2D channel estimation in UL PUSC (ITU-VA)

performance that there is around 1.2 dB degradation (may vary according to different channels) from channel estimation compared with 'PCK'. Moreover, the performance of estimation will degrade as the channel environment gets worse, e.g., from ITU-IB to ITU-VA. This is demonstrated by both the MSE and BER curves: the MSE curve is higher, and also the gap between the two BER curves becomes greater.

The MSE and BER performance of non-iterative and iterative channel estimations in both UL PUSC and OPUSC are illustrated. Particularly for PUSC, 2D and 2×1D channel estimations are compared, without and with iterations. The ITU-PB channel model is taken as an example to show these comparisons.

The MSE performance performance of PUSC 2×1D and 2D channel estimation are presented in Fig. 6.14 and Fig. 6.15 respectively, while in each case, channel estimation with different iterations are compared. It is shown that iterations can improve the performance of channel estimation, and most of gain comes from the first iteration ('iter=1'). Without iterations ('iter=0'), 2D estimation performs better than 2×1D estimation, while

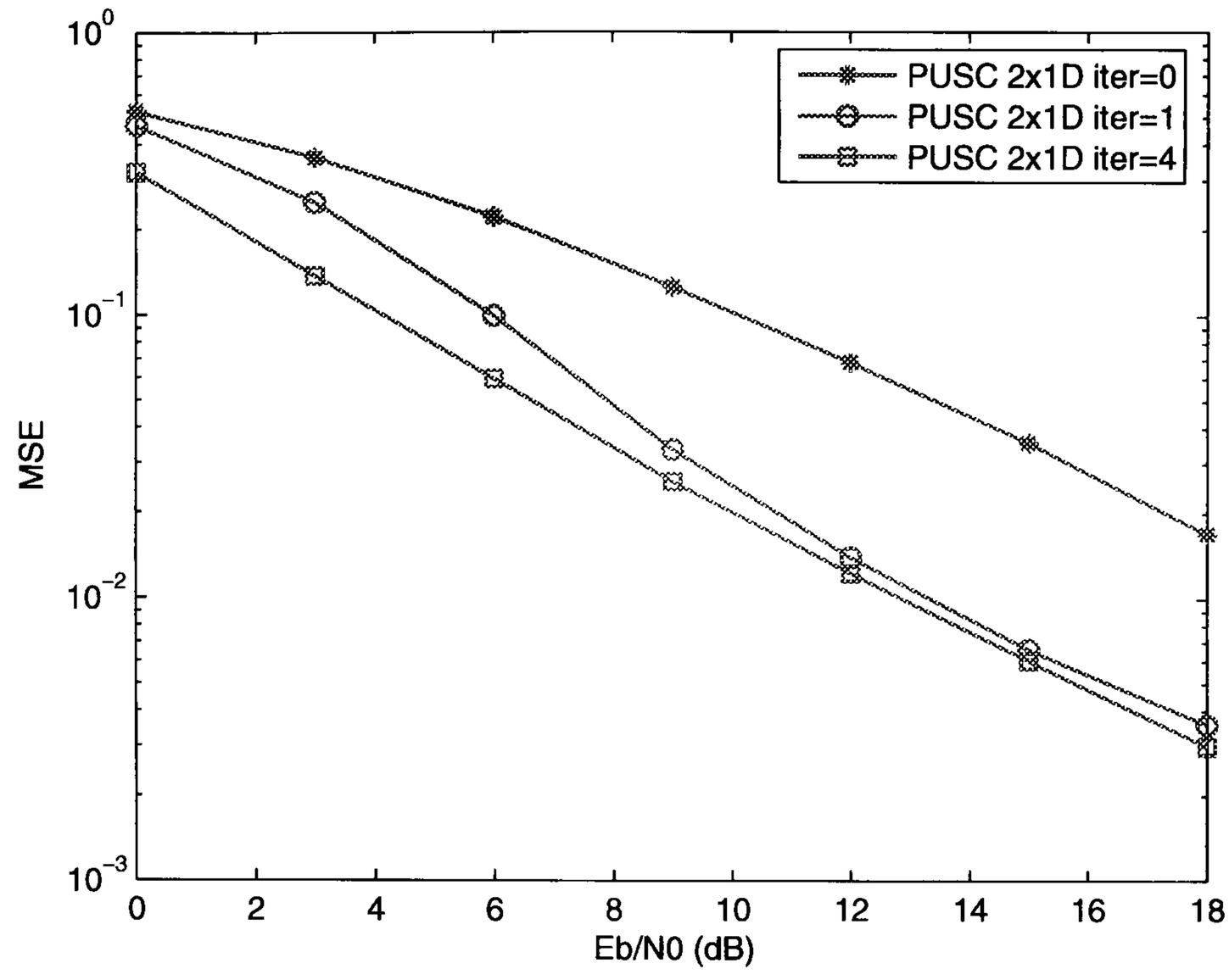


Figure 6.14: MSE performance of 2×1D iterative channel estimation in UL PUSC (ITU-PB)

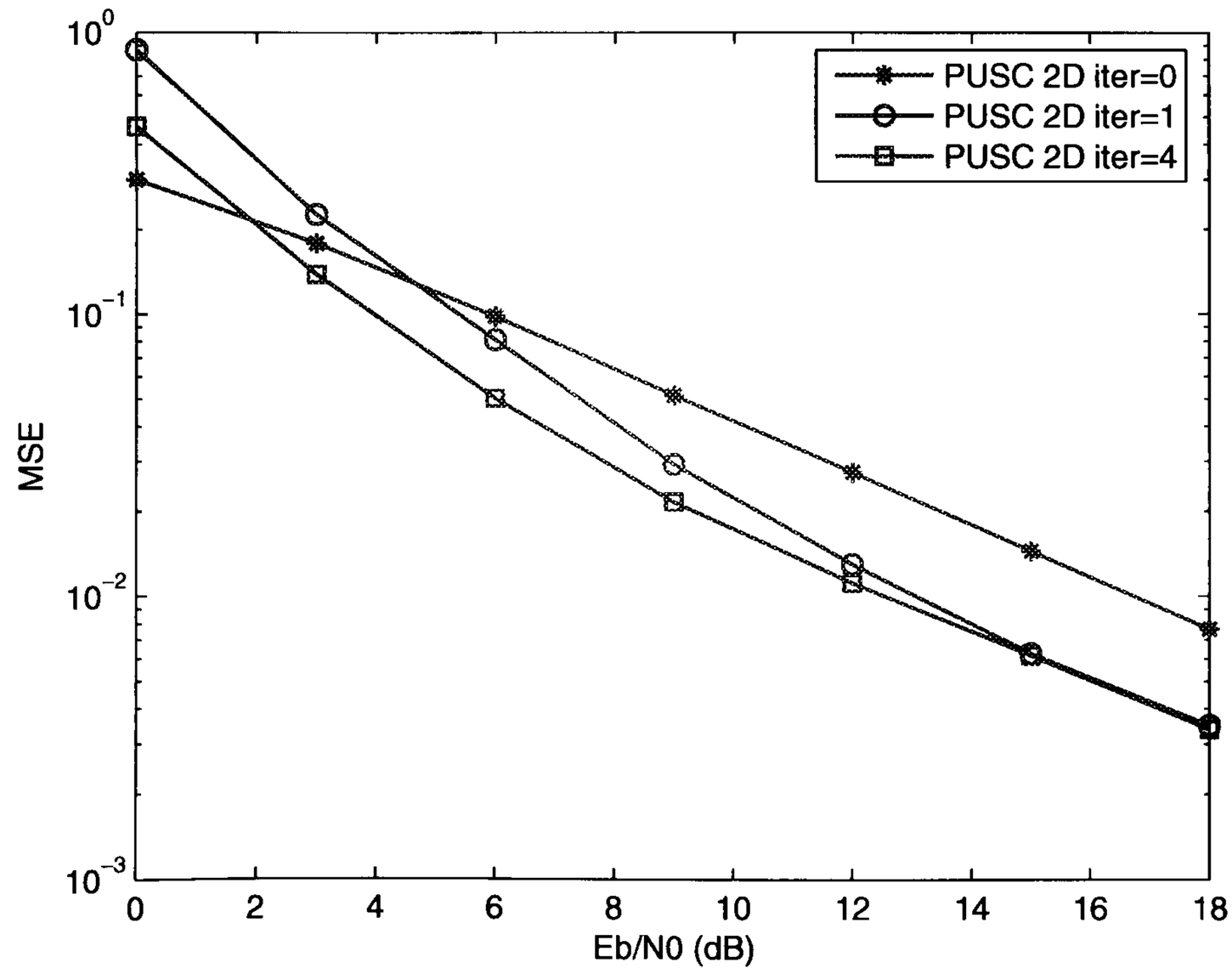


Figure 6.15: MSE performance of 2D iterative channel estimation in UL PUSC (ITU-PB)

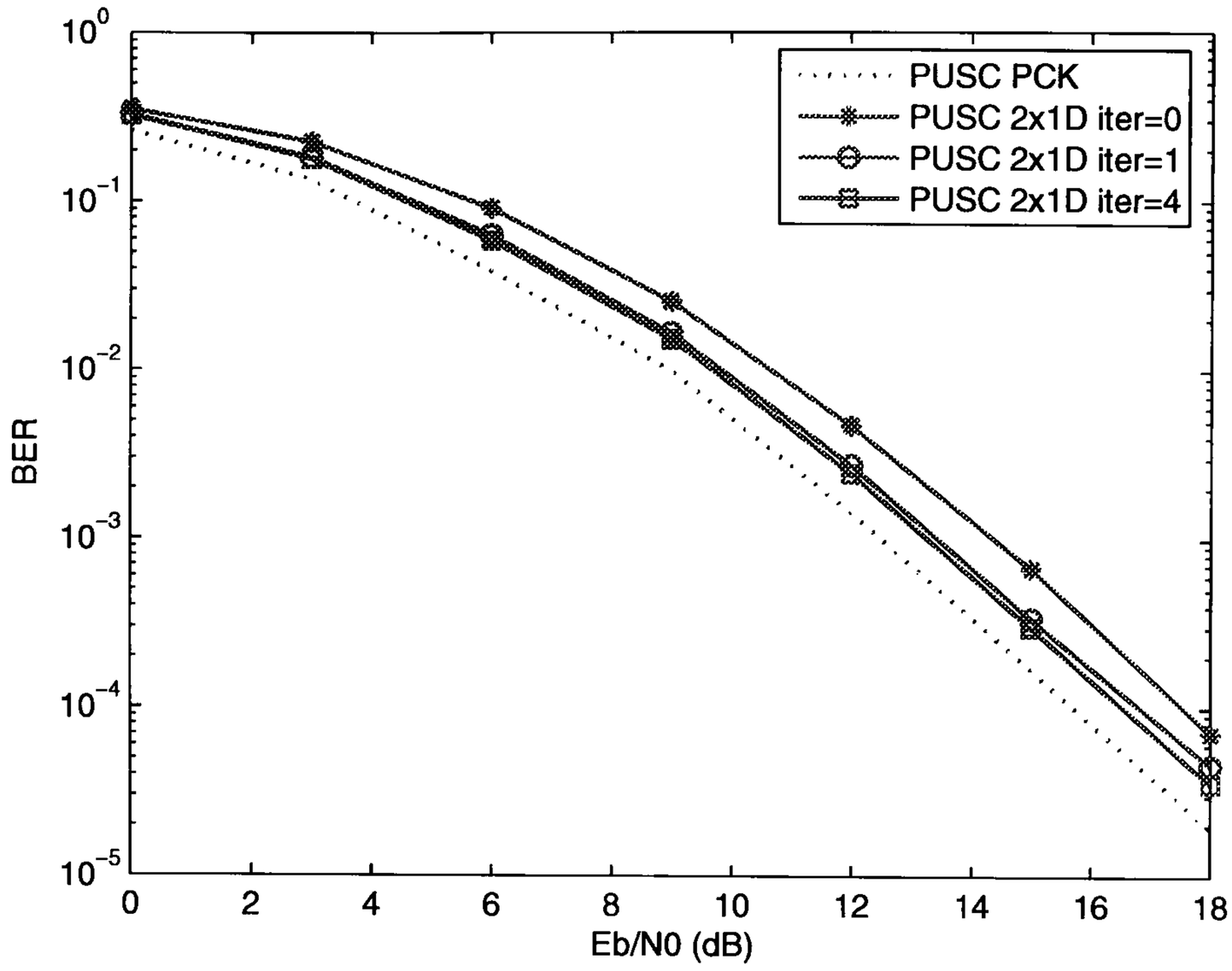


Figure 6.16: BER performance of 2×1D iterative channel estimation in UL PUSC (ITU-PB)

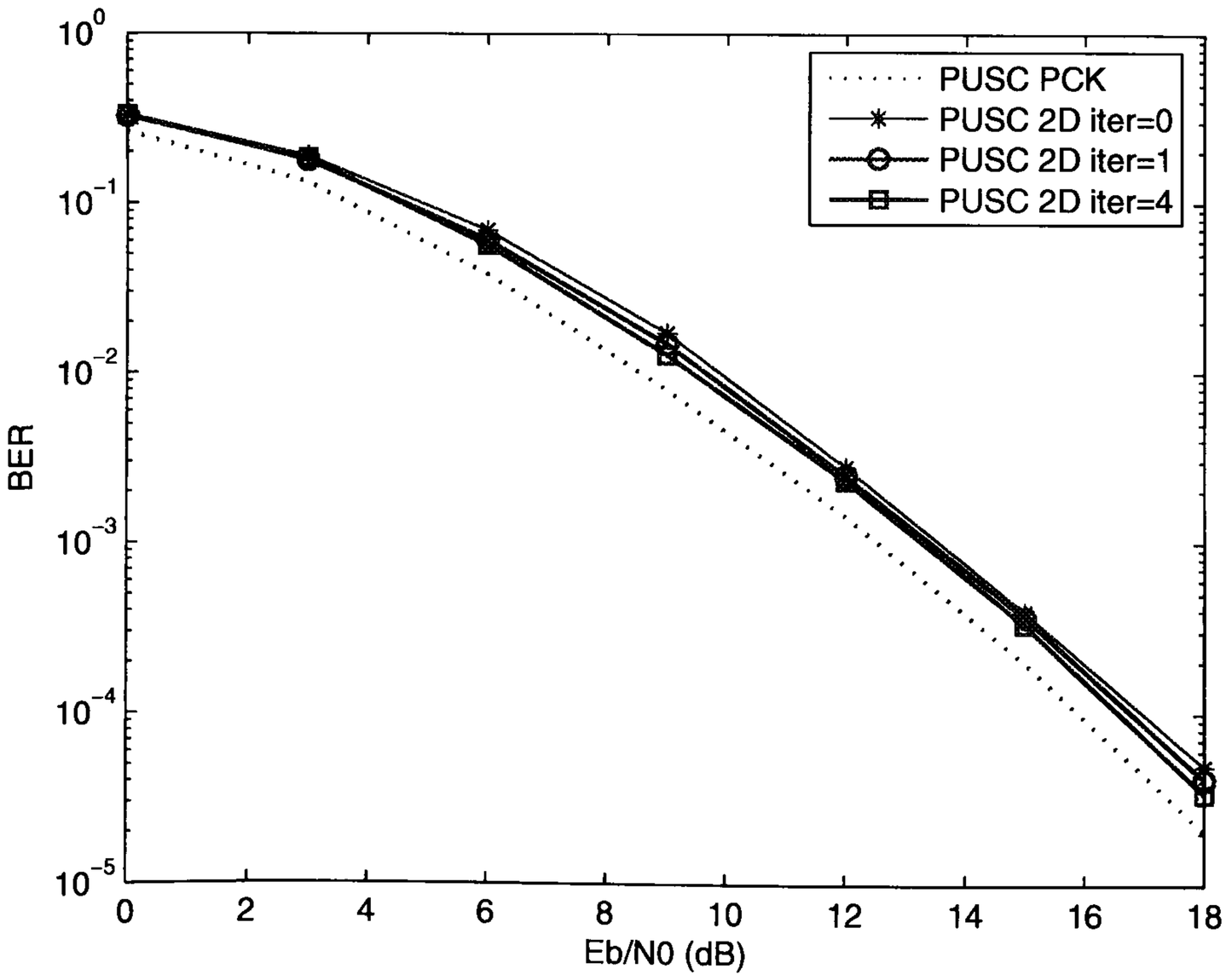


Figure 6.17: BER performance of 2D iterative channel estimation in UL PUSC (ITU-PB)

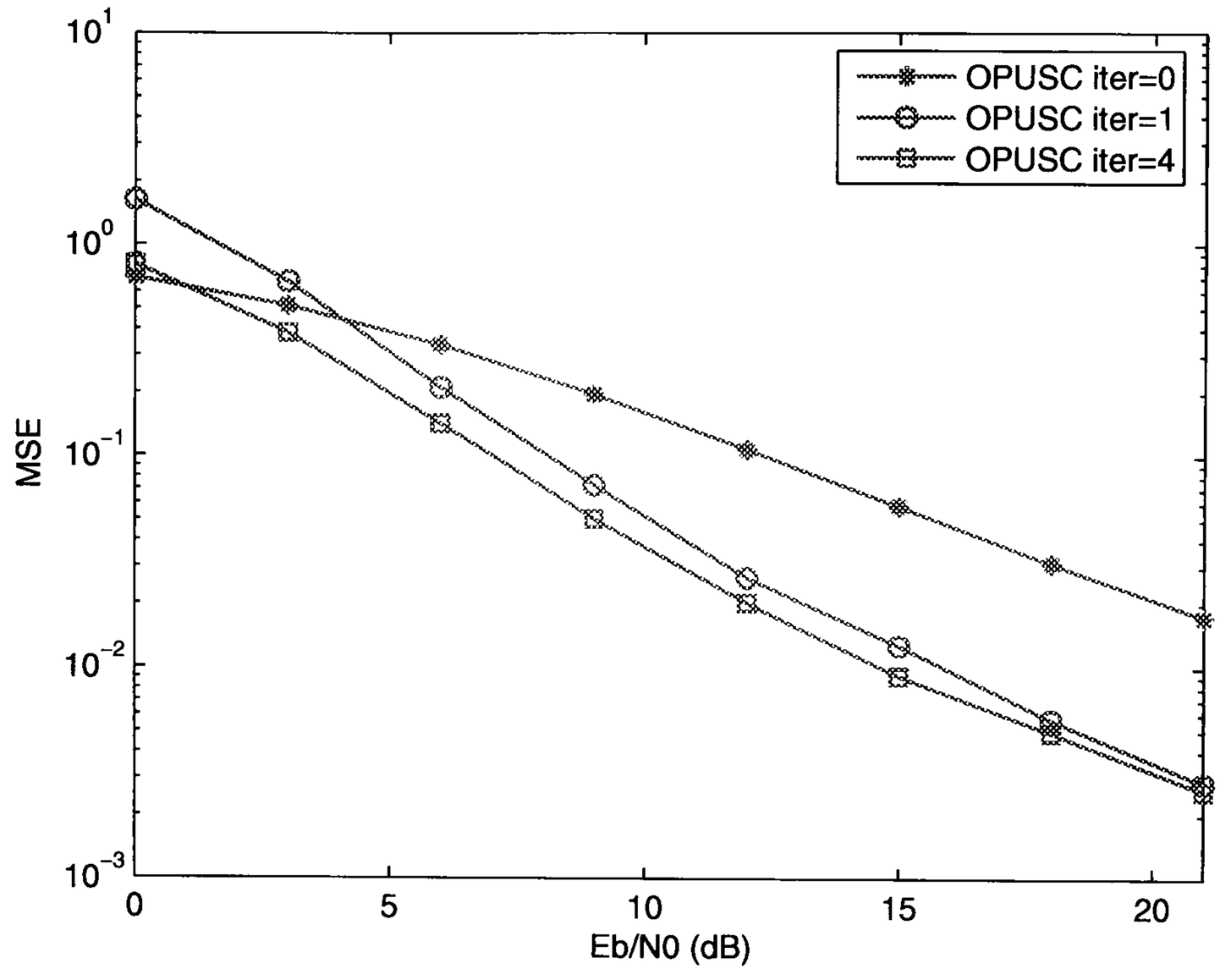


Figure 6.18: MSE performance of iterative channel estimation in UL OPUSC (ITU-PB)

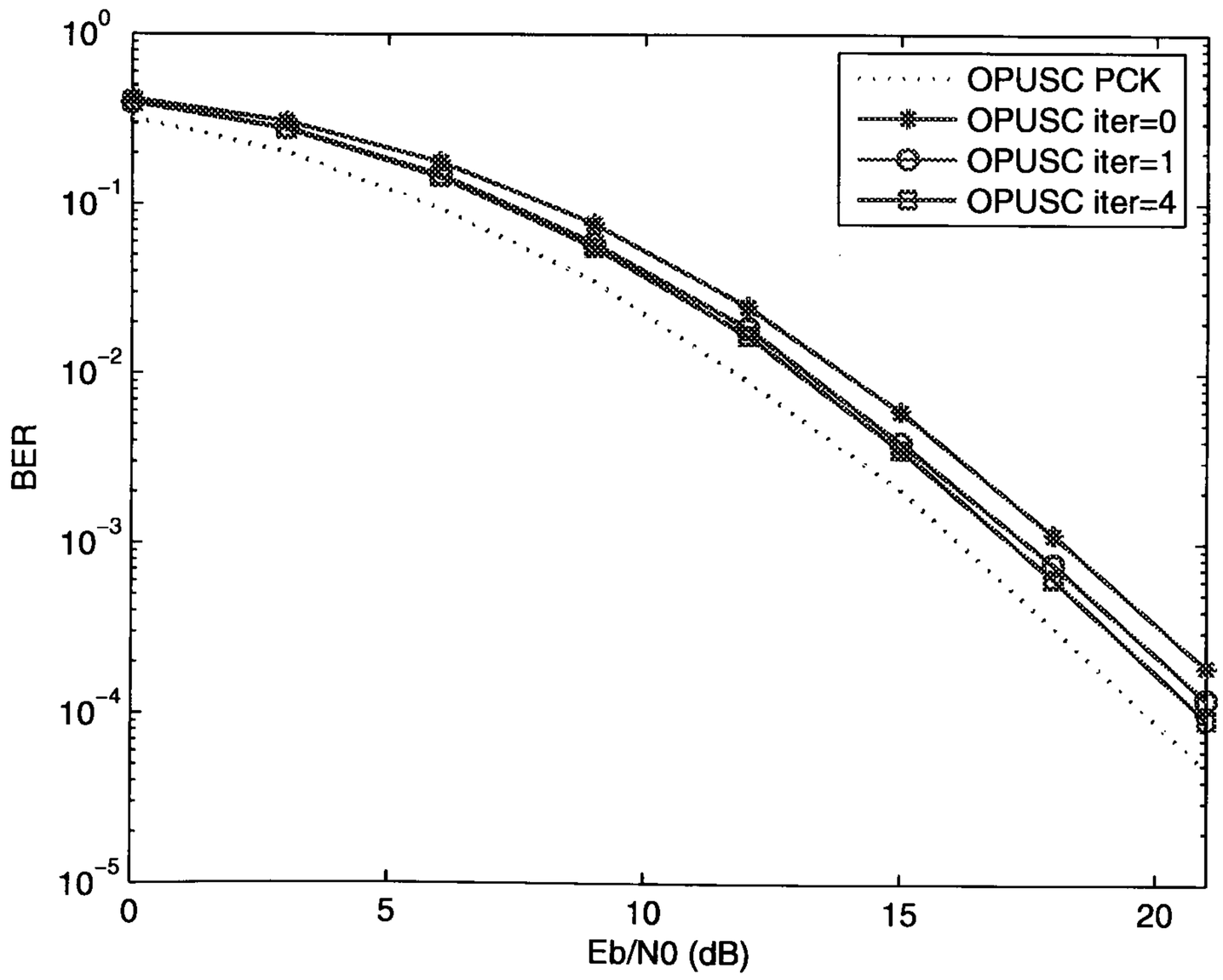


Figure 6.19: BER performance of iterative channel estimation in UL OPUSC (ITU-PB)

with increased iteration numbers, the performance of the two tend to converge ('iter=4'). In other words, 2×1D estimation is able to exploit more benefit from iterations than 2D estimation. The corresponding BER performance are shown in Fig. 6.16 and Fig. 6.17. It is noticed that 2×1D estimation with 4 iterations can bring a gain of about 1 dB at the BER of 10^{-4}, but in 2D estimation, the benefit from iterations is not as significant. Nevertheless it is still possible to obtain a gain of about 0.4 dB at the BER of 10^{-4}. It is also noticed that from the MSE performance, although not reflected from the BER performance, the actual MSE does not match the reference MSE at low E_b/N_0 regions. This is a common behavior in the iterative method, since at low E_b/N_0 regions, more errors are fed back through iterations. In this particular case, 2×1D estimation performs better than 2D estimation from this point view.

Now the computational complexity of 2×1D and 2D channel estimation in PUSC are considered. According to Eq. 6.9 and Eq. 6.18, the complexity of Wiener filtering mostly comes from the matrix multiplication, whose complexity is proportional to the square of the number of Wiener filter taps. In the non-iterative 2×1D estimation, this will be a 4-tap Wiener filter across time and a 2-tap Wiener filter across frequency, while in the iterative case, the numbers increase to 6 and 4 respectively. Similarly, for 2D estimation, this will be a 8-tap Wiener filter in the non-iterative case and then a full 24-tap Wiener filter in the iterative case. Therefore, the complexity of 2×1D estimation is proportional to $[(2^2+4^2)+(4^2+6^2)(I-1)]$, if I is the maximum iteration number. This is much less than that of 2D estimation, the complexity of which is 8^2 in the first and $[24^2(I-1)]$ in the subsequent iterations.

The MSE and BER performance of iterative 2×1D channel estimation in OPUSC are given by Fig. 6.18 and Fig. 6.19. Again, the first iteration provides most of the gain, while the improvement after that is small. At the BER of nearly 10^{-4}, a gain of about 1.3 dB can be provided by 4 iterations. Compared with PUSC 2×1D estimation, OPUSC can obtain more gain from iterations, since the number of pilots in the estimation block is much less than that in PUSC, hence more additional pilots are available in the iterative

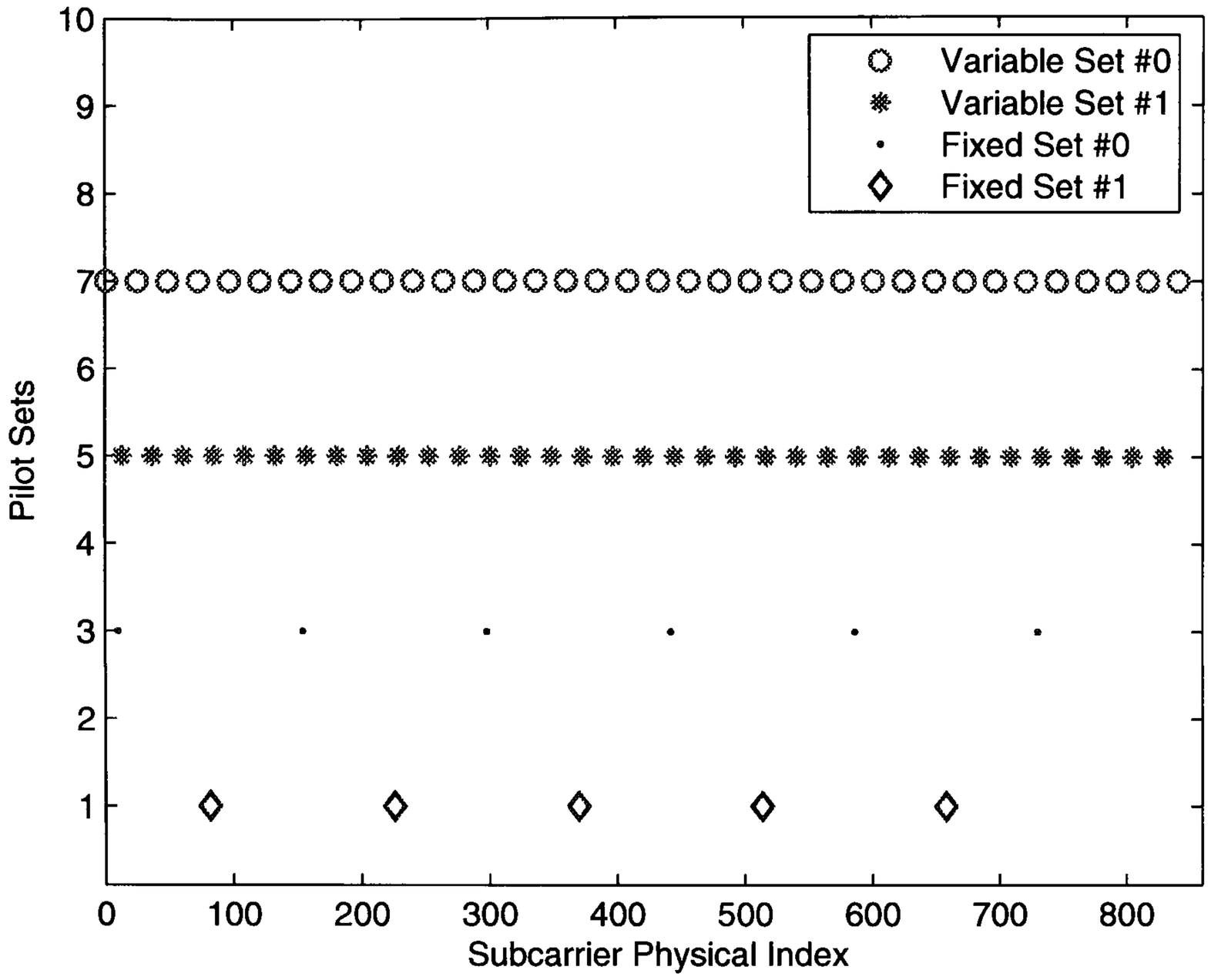


Figure 6.20: Pilot distribution for DL FUSC

case.

6.2.4.3 DL FUSC and PUSC

According to the standard [2], pilots allocation in DL is totally different from that in UL. Most distinctively, a preamble symbol, with a symbol power boost of 9 dB, is located at the beginning of each DL sub-frame, and can be used for estimation purpose. This section discusses the performance of channel estimation in DL FUSC and PUSC. Note that the channel information of FCH/MAP symbol next to the preamble also needs to be estimated.

Unlike UL tile structure, where pilots are fixed in certain positions before the tile permutation, the DL symbol is first allocated with scattered pilot subcarriers in appropriate

positions, and also zero subcarriers in the middle and at both ends. Then all the remaining subcarriers are used as data subcarriers, which will be divided into sub-channels according to the DL data subcarriers partitioning (permutation) scheme.

I. FUSC

As illustrated in the standard [2], the DL sub-frame contains 16 OFDM symbols, including a preamble symbol. All the 16 symbols and subcarriers are assigned for one user. There are two types of pilots: variable and fixed pilots, with two sets of each. The fixed sets are used in all OFDMA symbols, while the variable sets are divided into subsets that are used in odd and even symbols alternatively. This is to provide an appropriate tradeoff between allocated power and frequency diversity on pilots for channel estimation [4]. Fig. 6.20 shows the distribution of variable and fixed sets of pilots in the case of 1024 FFT. To distinguish them, different sets of pilots are marked with different levels.

Fig. 6.21 illustrates the block diagram of DL FUSC sub-frame. Since all the pilots are intended for one user, all pilots can be used in 2×1D Wiener filtering. Across OFDM symbols, an 8-tap Wiener filter can be applied to obtain time estimates in the data positions as well as the pilot positions ('×'s); then the remaining subcarriers can be estimated by Wiener filtering across frequency. Observing the pilots spacing in frequency, 4 pilots that are nearest to the estimates are chosen to do Wiener filtering ('○'s). More pilots will not be very helpful but only increase the complexity, since they are far away from the estimates and hence have less correlations. Note that fixed pilots sets are not meant for channel estimation, since they are rare and not well-regulated. However, they can be used for any other estimation purposes, e.g., tracking the remaining frequency offset after the initial training (preamble) symbol [3].

The BER and MSE performance of 2D estimation in UL PUSC in ITU-IB, ITU-PB and ITU-VA channel are given in Fig. 6.22, 6.23 and 6.24 respectively. Compared with that of UL PUSC in corresponding channel conditions, the BER performance of DL FUSC

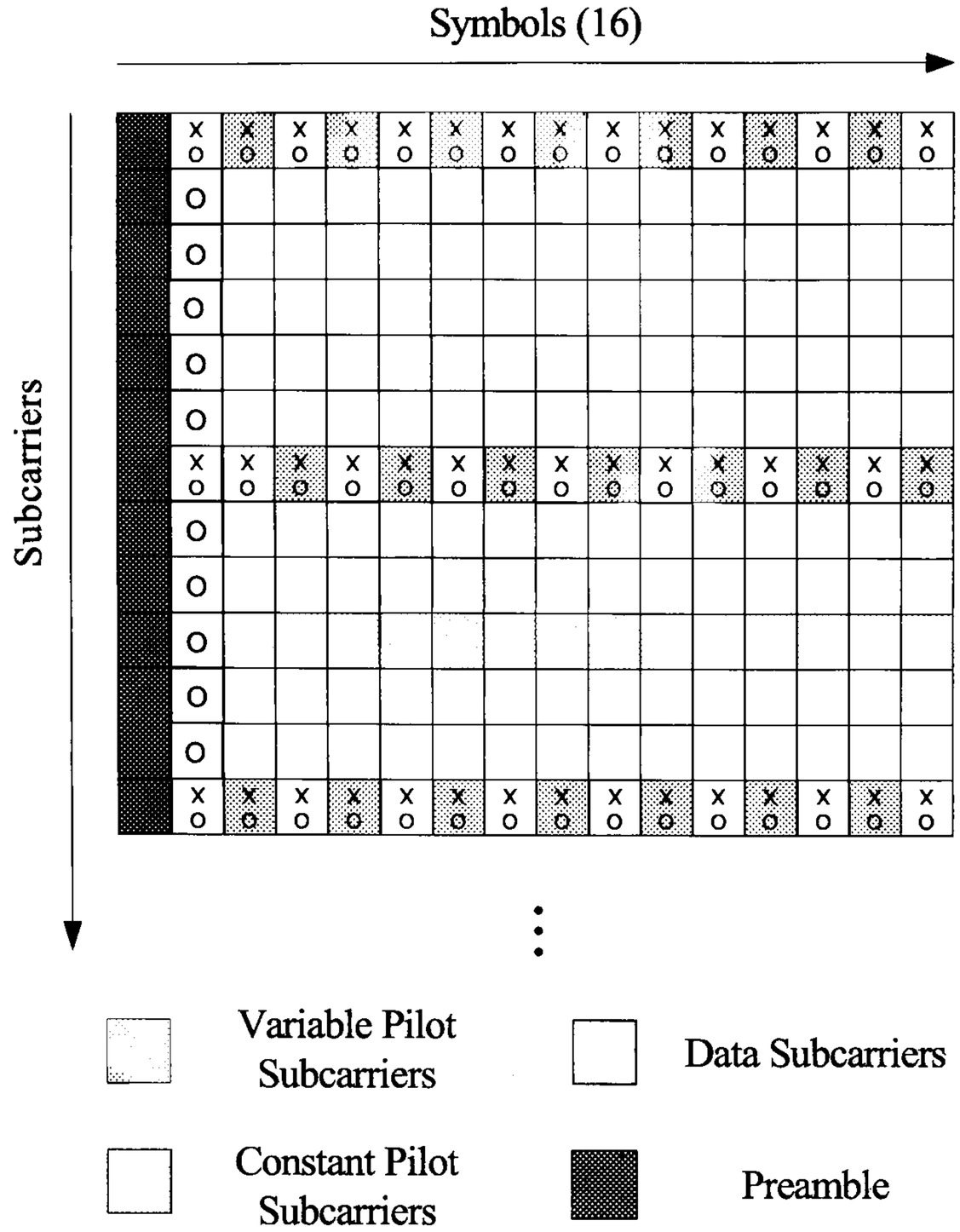


Figure 6.21: DL FUSC sub-frame

has much less degradation, 0.5 dB at the most ('ITU-VA'). This is because more pilots are provided for DL FUSC than that for UL PUSC. Again, both MSE and BER performance of channel estimation depend on the channel environment: the better the channel conditions, the better the performance is.

II. PUSC

DL PUSC uses a cluster structure, as illustrated in Fig. 6.25, which spans over two OFDM symbols of fourteen subcarriers, each with a total of four pilot subcarriers scattered within it. The symbol is divided into basic clusters after zero guard band and DC subcarriers are allocated. A DL PUSC sub-channel is composed of two clusters. Allocating subcarriers to sub-channel is performed by first allocating the pilot subcarriers within each cluster, and then taking all the remaining data subcarriers within the symbol and applying a per-

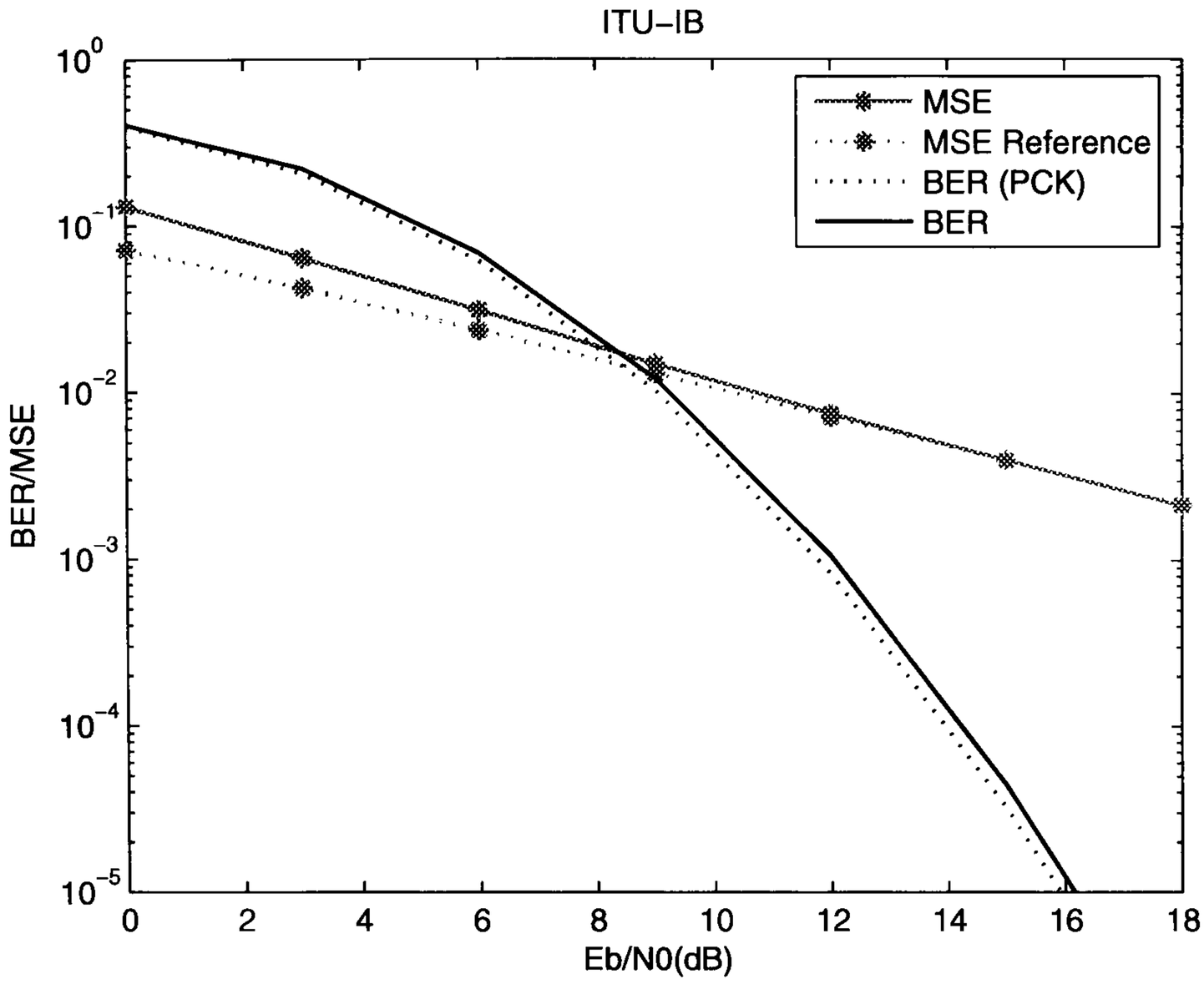


Figure 6.22: BER and MSE performance of 2×1D channel estimation in DL FUSC (ITU-IB)

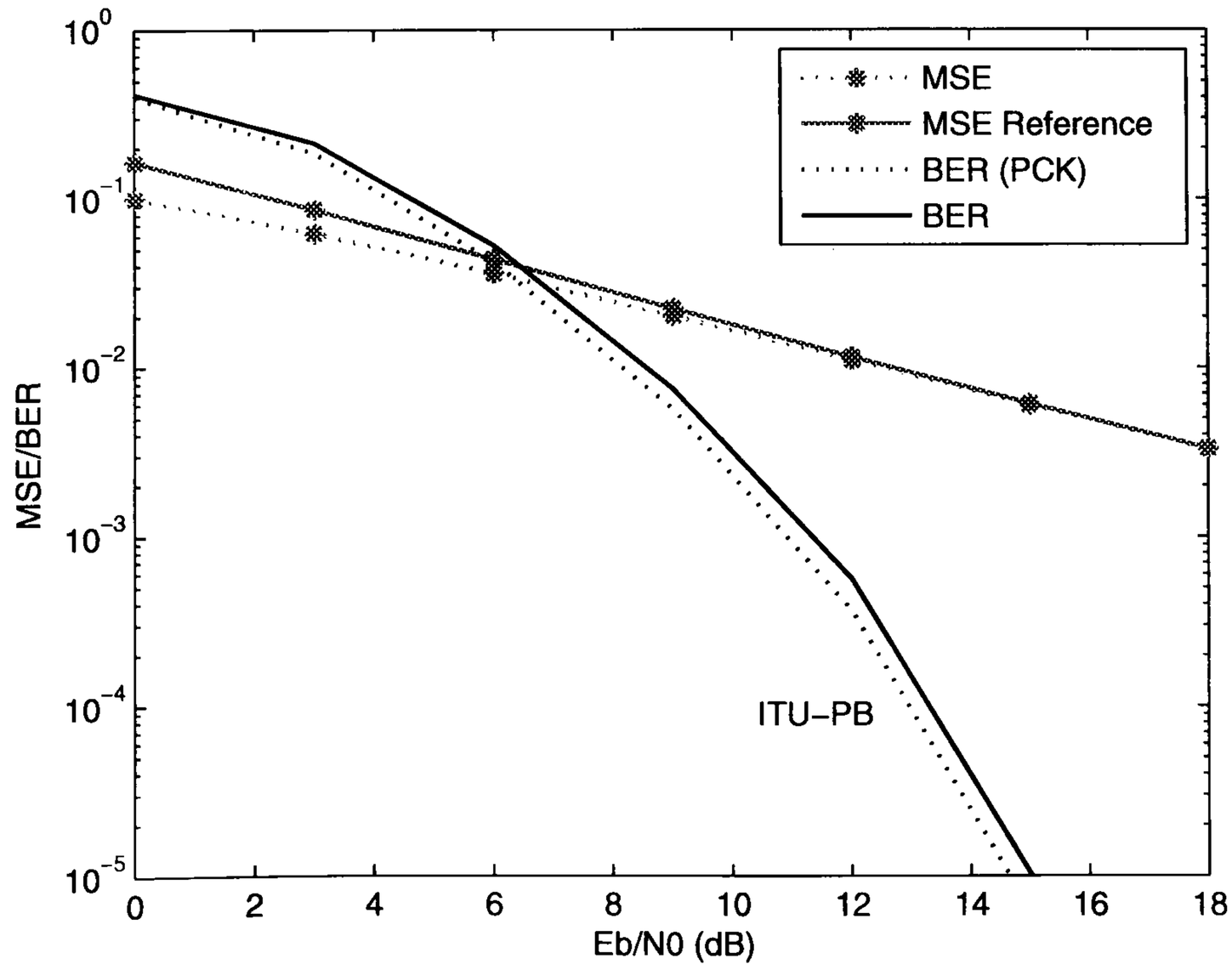


Figure 6.23: BER and MSE performance of 2×1D channel estimation in DL FUSC (ITU-PB)

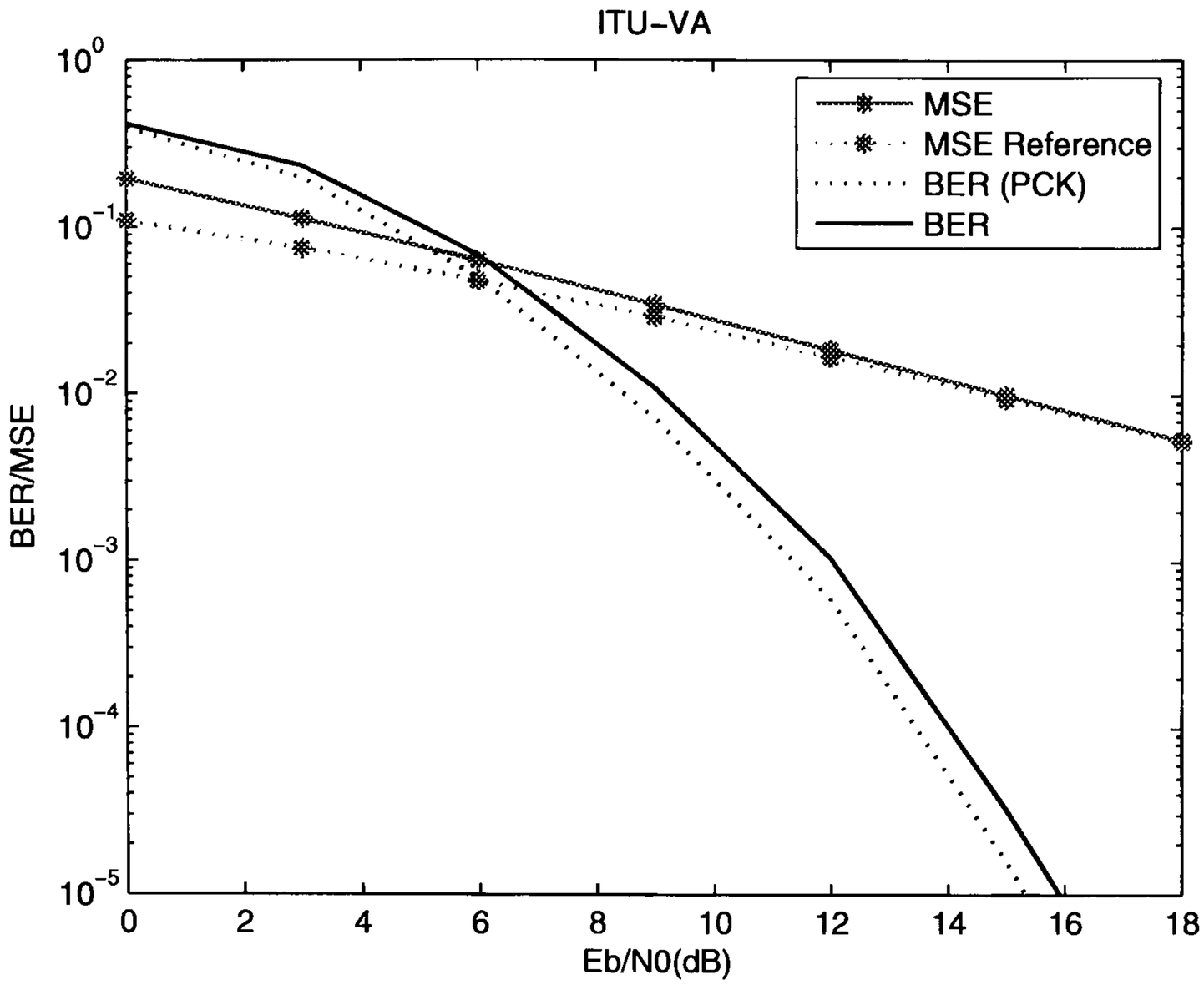


Figure 6.24: BER and MSE performance of 2×1D channel estimation in DL FUSC (ITU-VA)

mutation scheme to partition the data subcarriers into sub-channels containing 24 data subcarriers per symbol.

Assume in DL PUSC, each user occupies two sub-channels spanning across frequency. After permutation, data subcarriers of the user are separated across all the available active subcarriers, while the pilot subcarriers remain in the same positions. Fig. 6.26 illustrates a cluster following the preamble and FCH/MAP symbol. Other subcarriers will expand in frequency, and each user has four clusters for transmission its data, which are distributed across frequency.

Since data subcarriers for each user is scarcely scattered, it is not efficient to first locate the positions of data subcarriers, and then find several nearest pilots near them before applying Wiener filter interpolation. Moreover, the preamble is not used in this method, which is a waste. To make full use of the preamble symbol as well as the scattered pilots, a combined coded-aided iterative channel tracking (EM algorithm) and channel estimation (Wiener filtering) approach is employed. Note that the FCH/MAP symbol

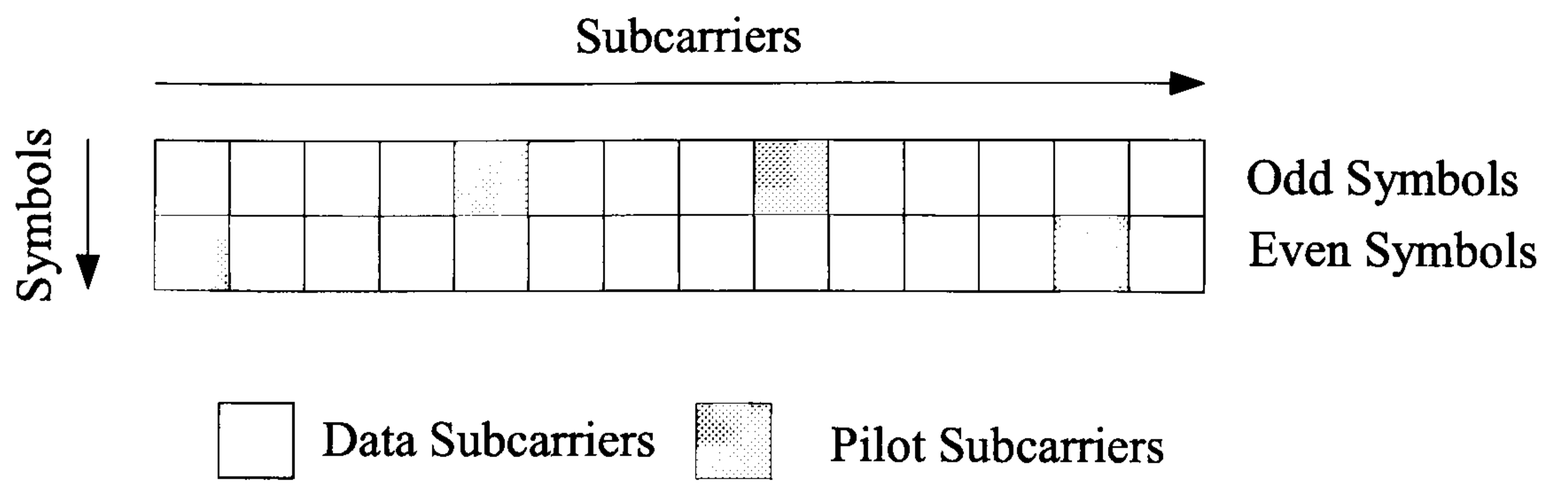


Figure 6.25: DL PUSC cluster structure

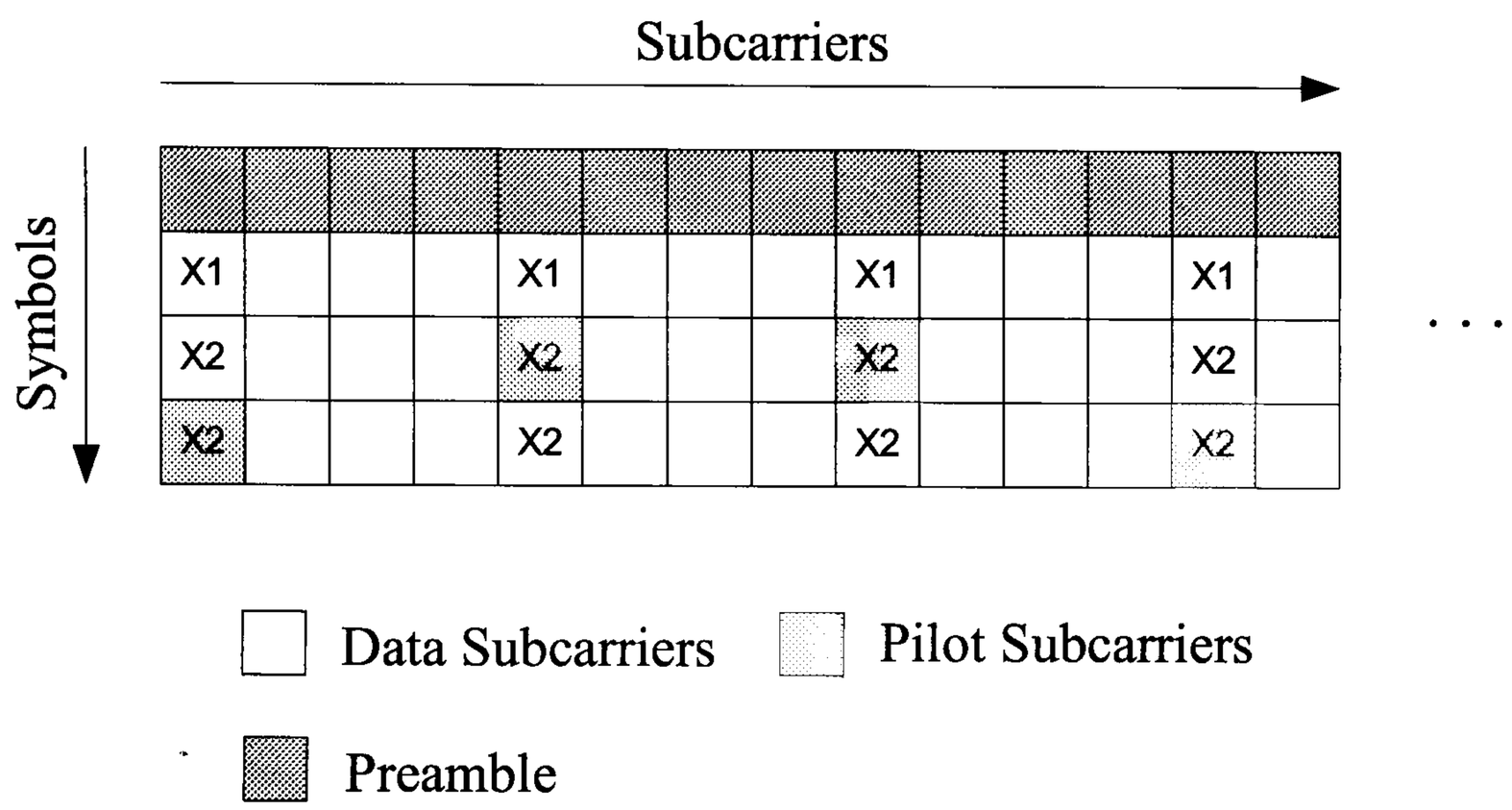


Figure 6.26: DL PUSC Channel estimation

uses all the active subcarriers and has no pilots within them, while one user has much fewer subcarriers but with pilots regularly assigned. Hence there is a slight difference between the estimation in FCH/MAP symbol and user's data symbols.

The process is described as follows. First of all, PCK is assumed at the preamble symbol and is regarded as the initial channel values in the channel tracking for FCH/MAP symbol. This initials at some positions can be refined by a 2-tap Wiener filter using the pilots across time ('x1's). Then a 840-tap Wiener filter across all the active subcarriers is applied to improve the initial estimates. Equalization of the received FCH/MAP symbol are performed using the improved estimates before decoding. Soft coded bits from the Log-MAP decoder are then repeated, interleaved and modulated into soft coded symbols. Finally, the new channel estimates can be obtained by Eq. 6.39. This process can be

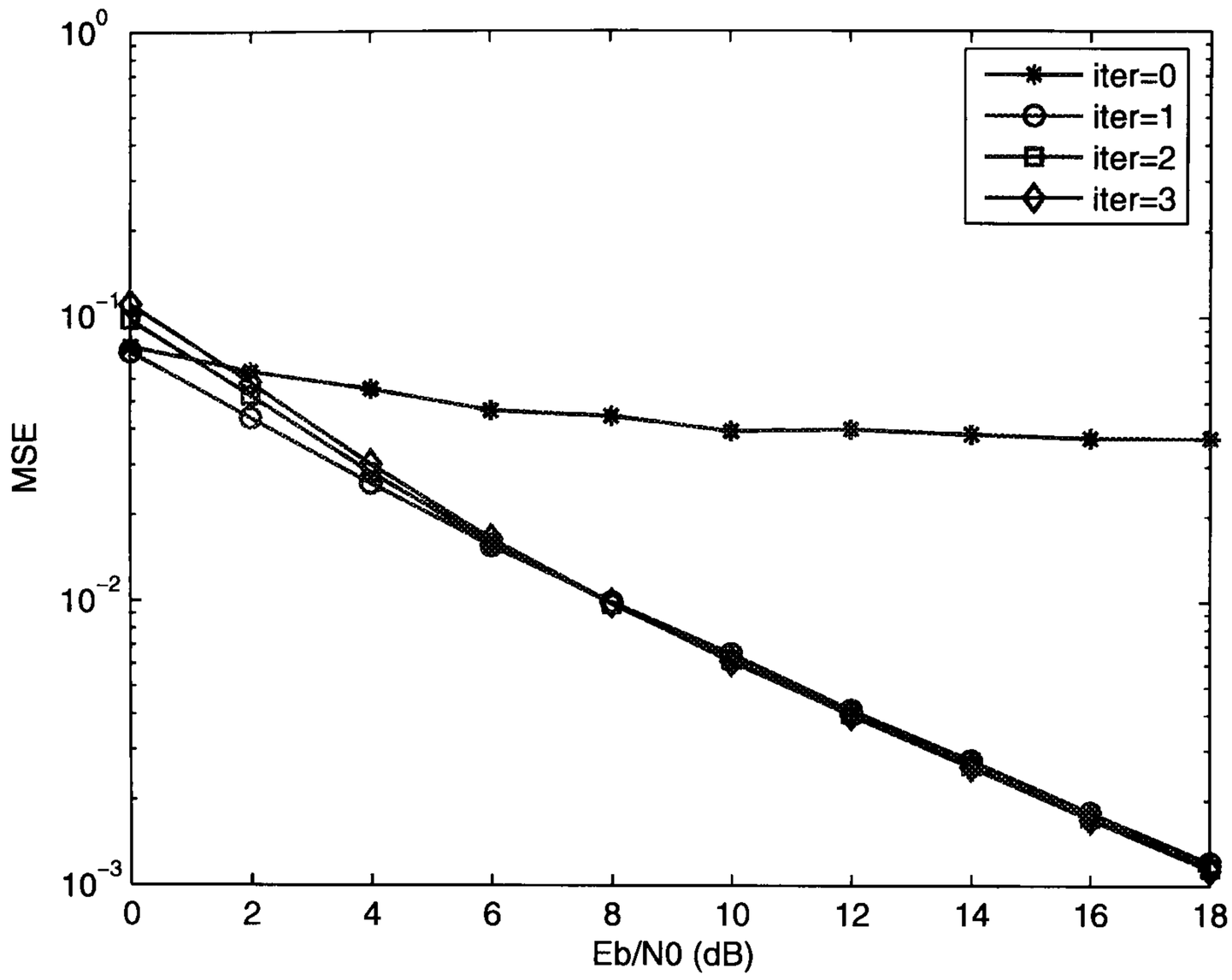


Figure 6.27: MSE performance of combined channel tracking and estimation in FCH/MAP symbol (DL PUSC)

repeated until the improvement from the iterations becomes limited.

The final channel estimates of FCH/MAP symbol are viewed as the initial estimates for the next two symbols for users. Similarly, the initials can be improved by applying a 3-tap Wiener filter across time using the original pilots as well as the estimates from the FCH/MAP symbol at pilot positions. A 276-tap (each symbol has 240 pilots and 48 data symbols for estimation, while 12 of them overlap) Wiener filter across frequency is interpolated to further improve the estimates. Then, taking out the obtained channel estimates at user's data positions, the equalization and Log-MAP decoding are performed. Finally, an EM algorithm is applied to obtain new estimates at user's data positions. After replacing the initial estimates at user's data subcarrier positions with the new estimates, the Wiener filtering can again be implemented, and the iteration goes on.

Simulation results of combined channel tracking and estimation are given. ITU-VA is adopted as the channel model in the simulations, and repetition coding is not implemented

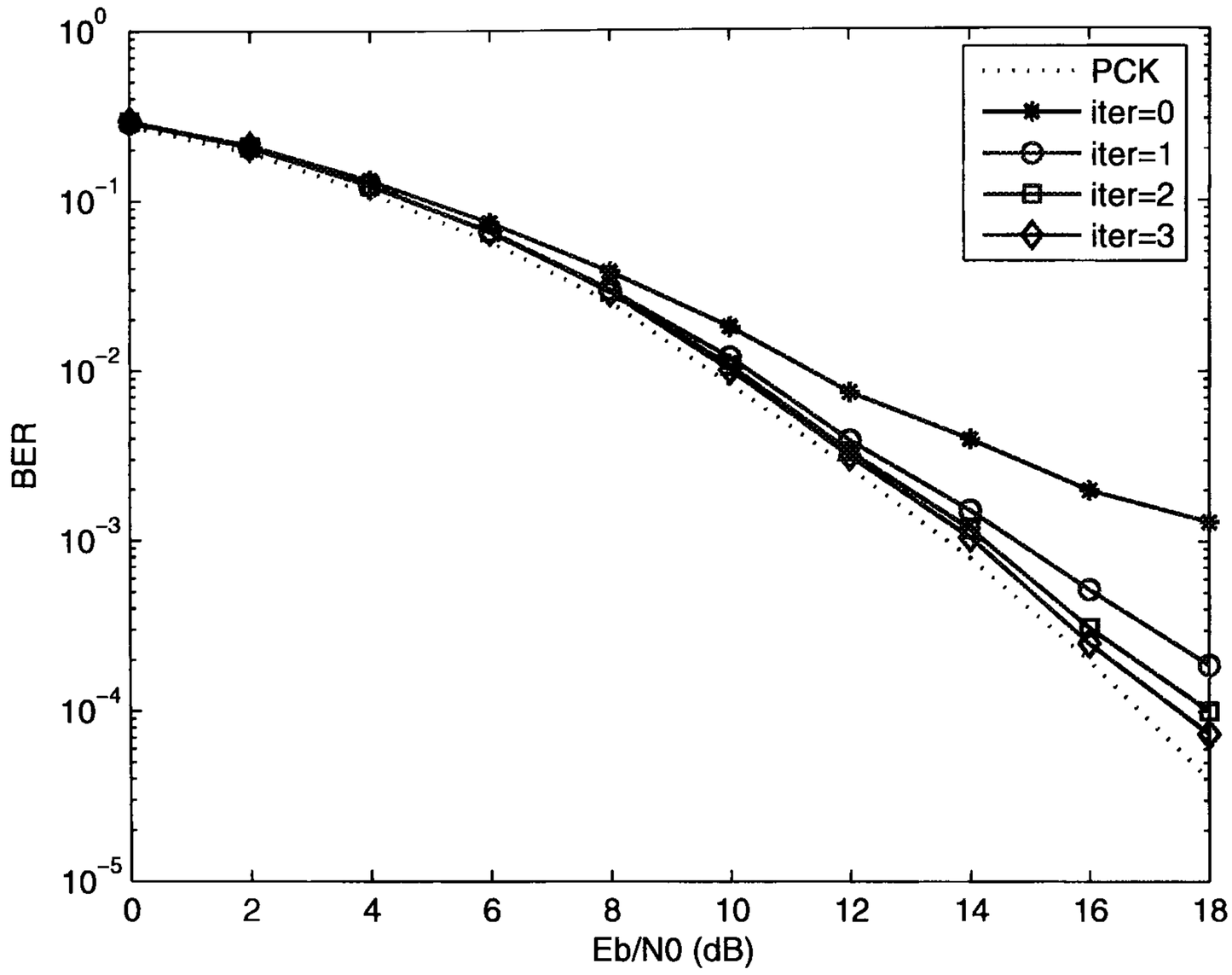


Figure 6.28: BER performance of combined channel tracking and estimation in FCH/MAP symbol (DL PUSC)

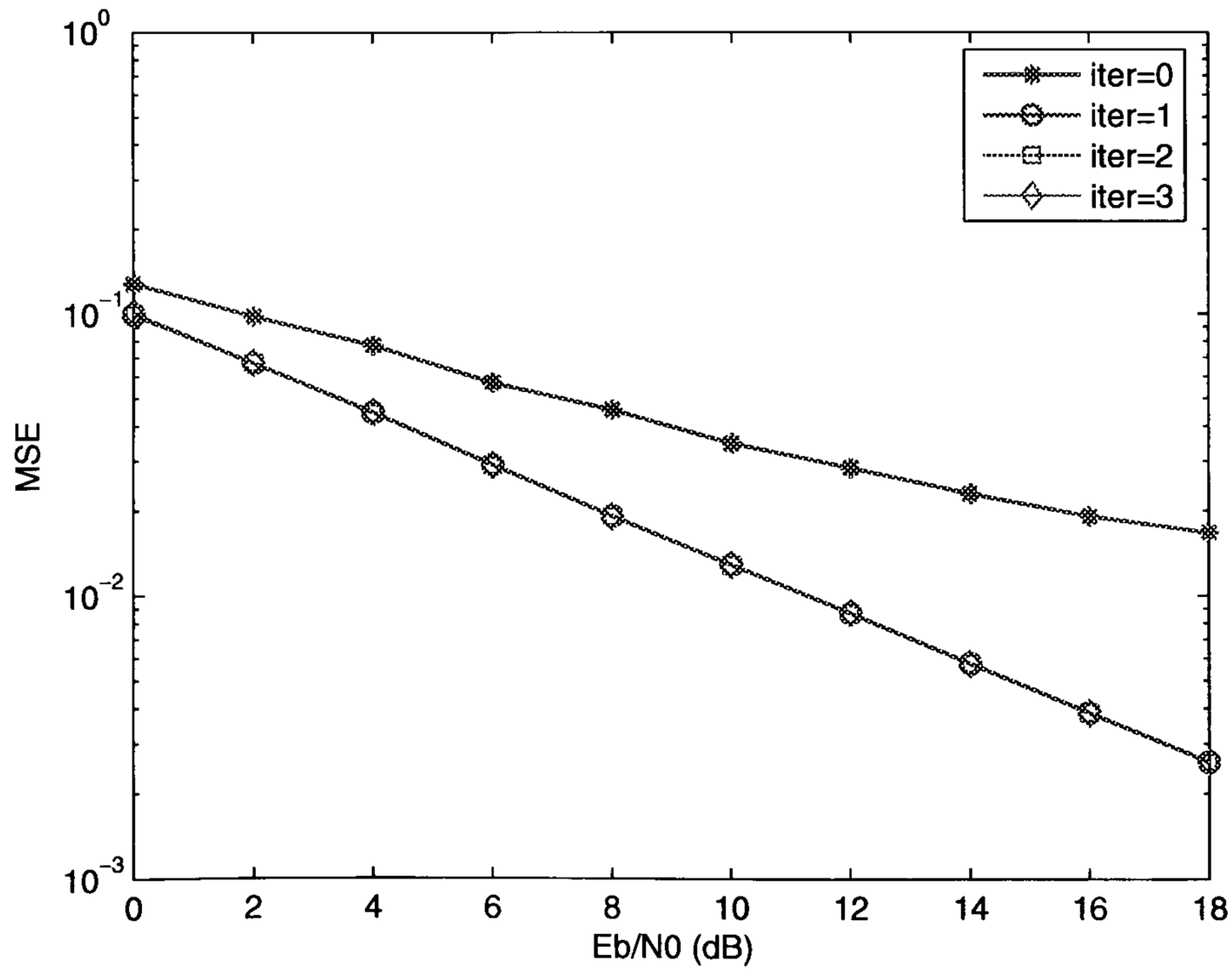


Figure 6.29: MSE performance of combined channel tracking and estimation in user's data symbols (DL PUSC)

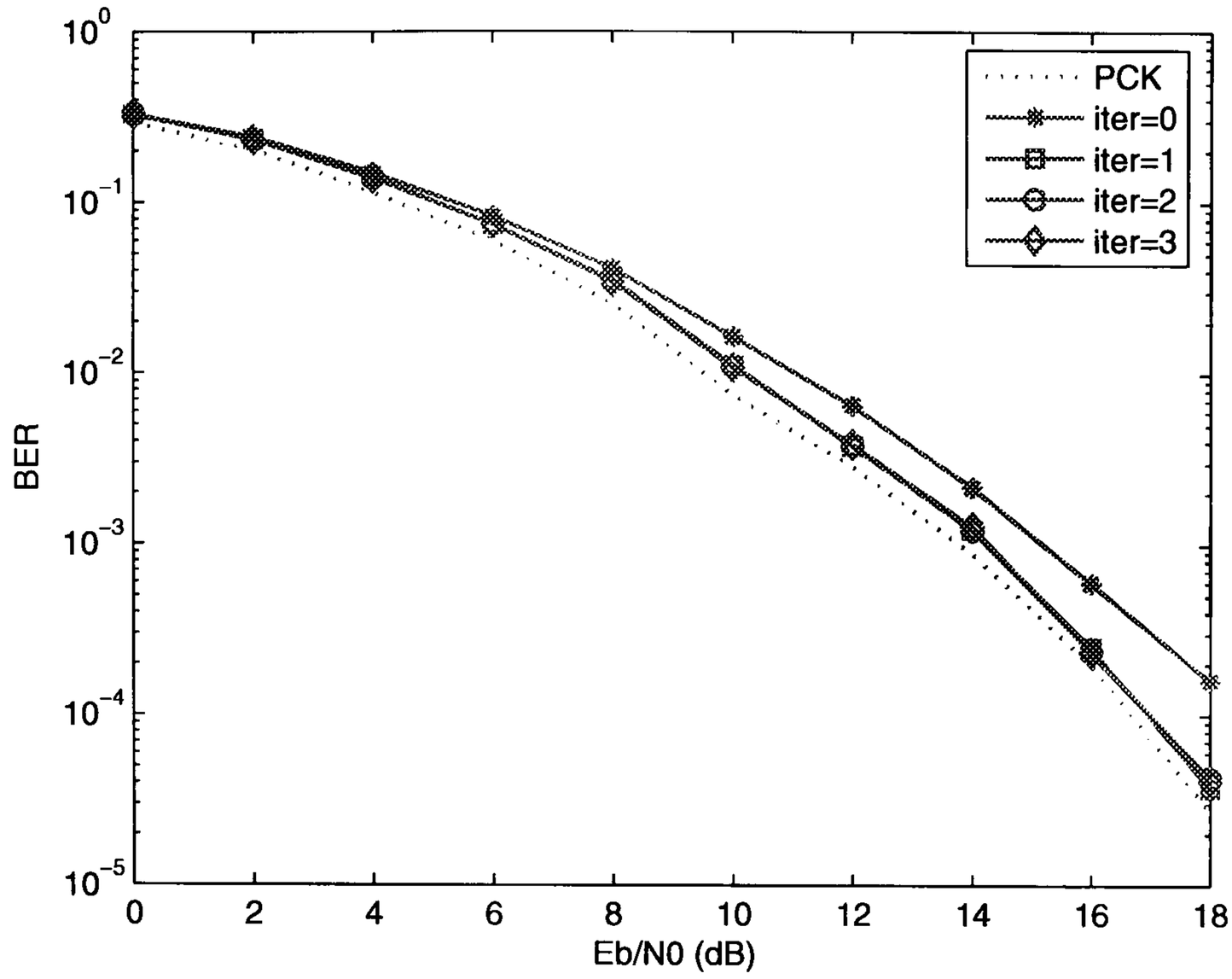


Figure 6.30: BER performance of combined channel tracking and estimation in user's data symbols (DL PUSC)

(N_r=1). Fig. 6.27 and Fig. 6.28 give the MSE and BER performance of tracking and estimation in FCH/MAP symbol. The results demonstrate that the performance is improved by iterations, and most gain comes from the first iteration. After three iterations, a gain of 4 dB is obtained at the BER of 10^{-3} and the degradation from channel estimation is less than 1 dB. Fig. 6.29 and Fig. 6.30 present the MSE and BER performance of tracking and estimation for user's data symbols. Again, most gain comes from the first iteration, which is 2 dB at the BER of 10^{-4}, while almost no improvement is obtained after that. With iterations, the degradation from channel estimation is as small as less than 0.5 dB. It is noticed that in the MSE and BER performance of FCH/MAP symbol, error floors exist in the high E_b/N_0 regions. This is due to the time-variant property of the channel between OFDM symbols. The same phenomenon appear in the MSE and BER performance of user's data symbol, but the user's data symbol is shorter than the FCH/MAP symbol, the error floor is therefore lower.

6.2.4.4 Mismatched Channel Estimation by Wiener Filtering

In previous Wiener filter based channel estimations, it is assumed that the delay spread in the channel power delay profile and Doppler frequency of the multipath channel are known when calculating the time and frequency correlations. However in practice, these information may not be able to be obtained, and a performance degradation may occur due to mismatched estimations using incorrect Doppler frequency and delay spread. Therefore, it is necessary to investigate how the Wiener filter performs in mismatched estimations. The BER performance will be compared with the one obtained by PCK, and the MSE performance will be compared between the one matched for the actual channel and the mismatched. The simulations results are obtained without coding.

I. Mismatched in Doppler Frequency

First, the estimation mismatched in Doppler frequency is discussed. ITU-IB and IUT-VA have the lowest and the highest vehicular speed in three ITU-R channel models used in the thesis. First, the case when user is stationary (ITU-IB), but Wiener filter is optimized for moving users (ITU-VA) is considered, the results of which are shown in Fig. 6.31 (UL PUSC) and Fig. 6.32 (DL FUSC). It is observed that although some degradation appears in MSE, the BER is not affected. The other way round, users are actually moving (ITU-VA), but estimator is optimized for stationary users (ITU-IB), the results of which are shown in Fig. 6.33 (UL PUSC) and Fig. 6.34 (DL FUSC). It is clear that both MSE and BER have unbearable degradations in this case.

II. Mismatched in delay spread

Now the mismatched estimation in delay spread is considered, where ITU-IB and ITU-PB are two extreme cases: ITU-IB has the shortest delay while ITU-PB has the longest. First, the channel has a shorter delay spread (ITU-IB), but the estimator is optimized for a channel with longer delay (ITU-PB). The results for UL and DL are shown in Fig. 6.35

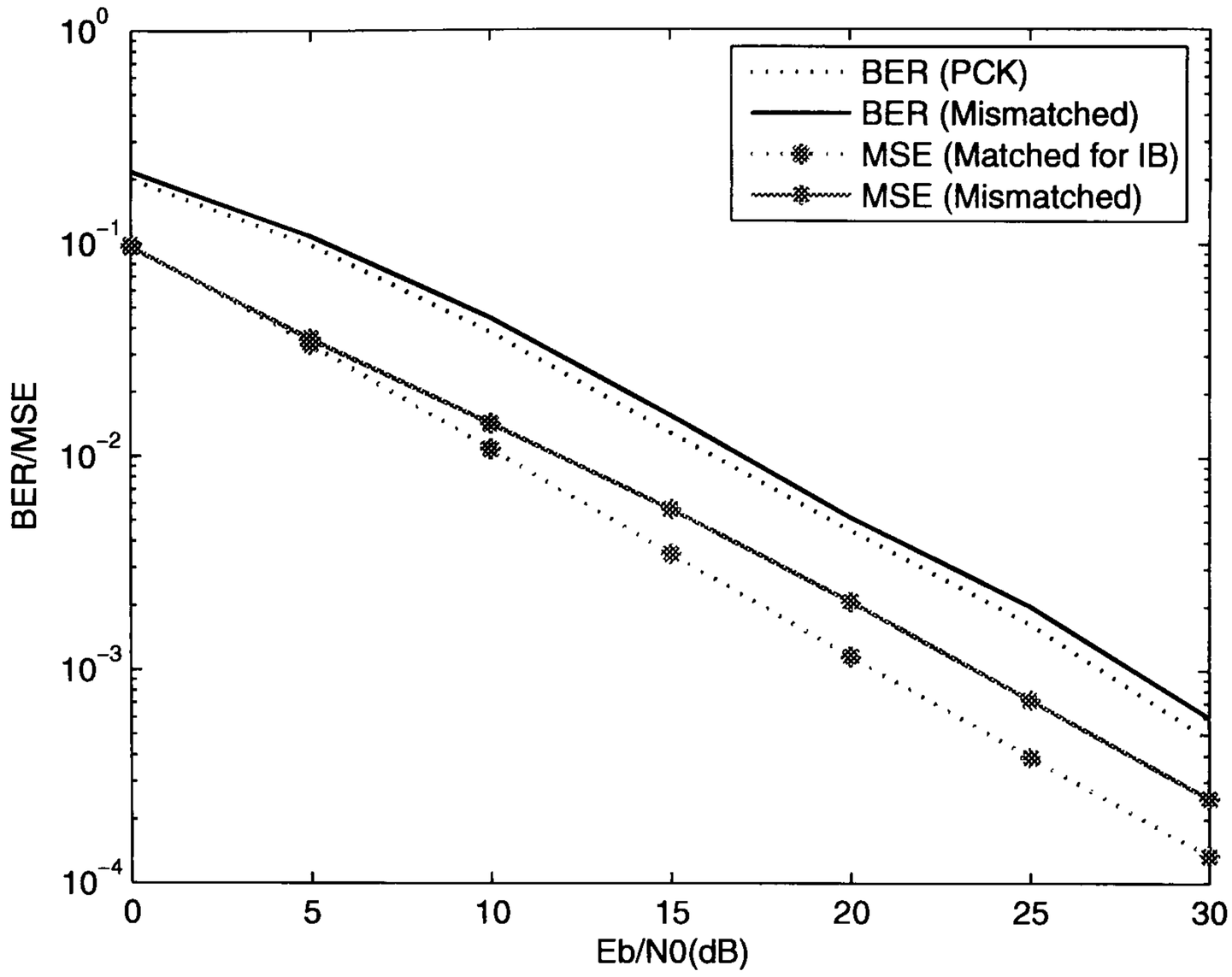


Figure 6.31: MSE and BER performance of mismatched channel estimation: users stationary (ITU-IB); estimator optimized for moving users (ITU-VA) (UL PUSC, uncoded)

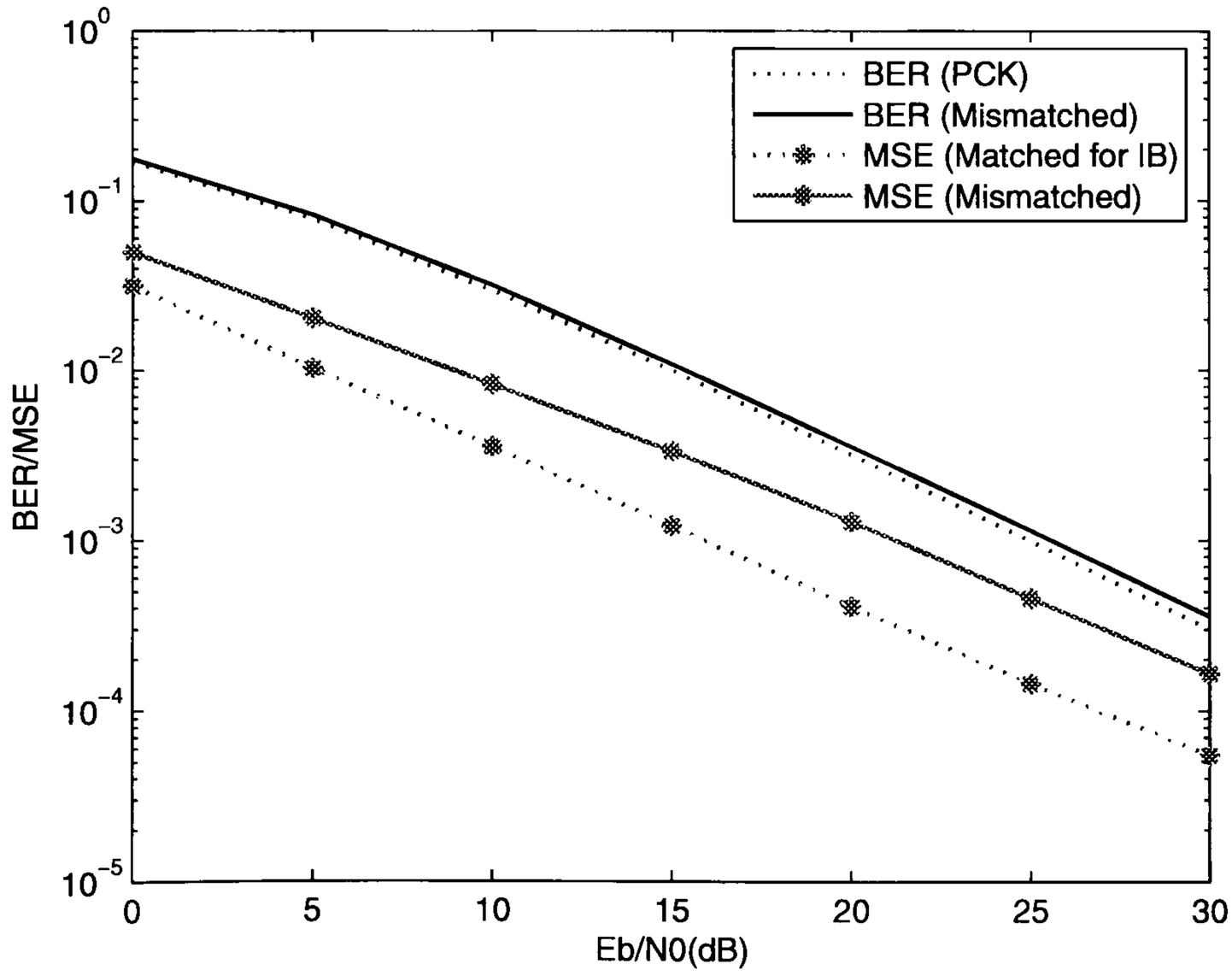


Figure 6.32: MSE and BER performance of mismatched channel estimation: users stationary (ITU-IB); estimator optimized for moving users (ITU-VA) (DL FUSC, uncoded)

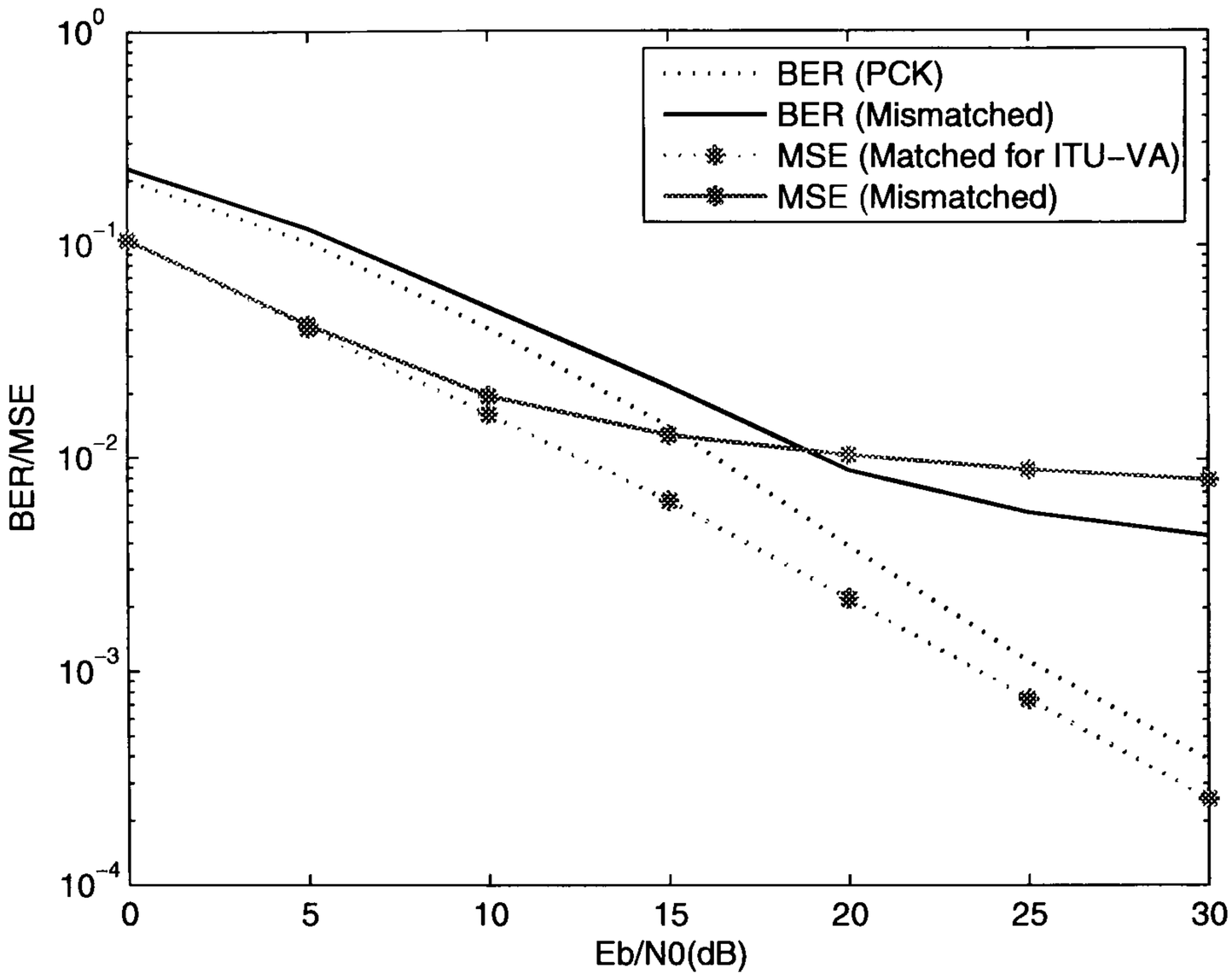


Figure 6.33: MSE and BER performance of mismatched channel estimation: users moving (ITU-VA); estimator optimized for stationary users (ITU-IB) (UL PUSC, uncoded)

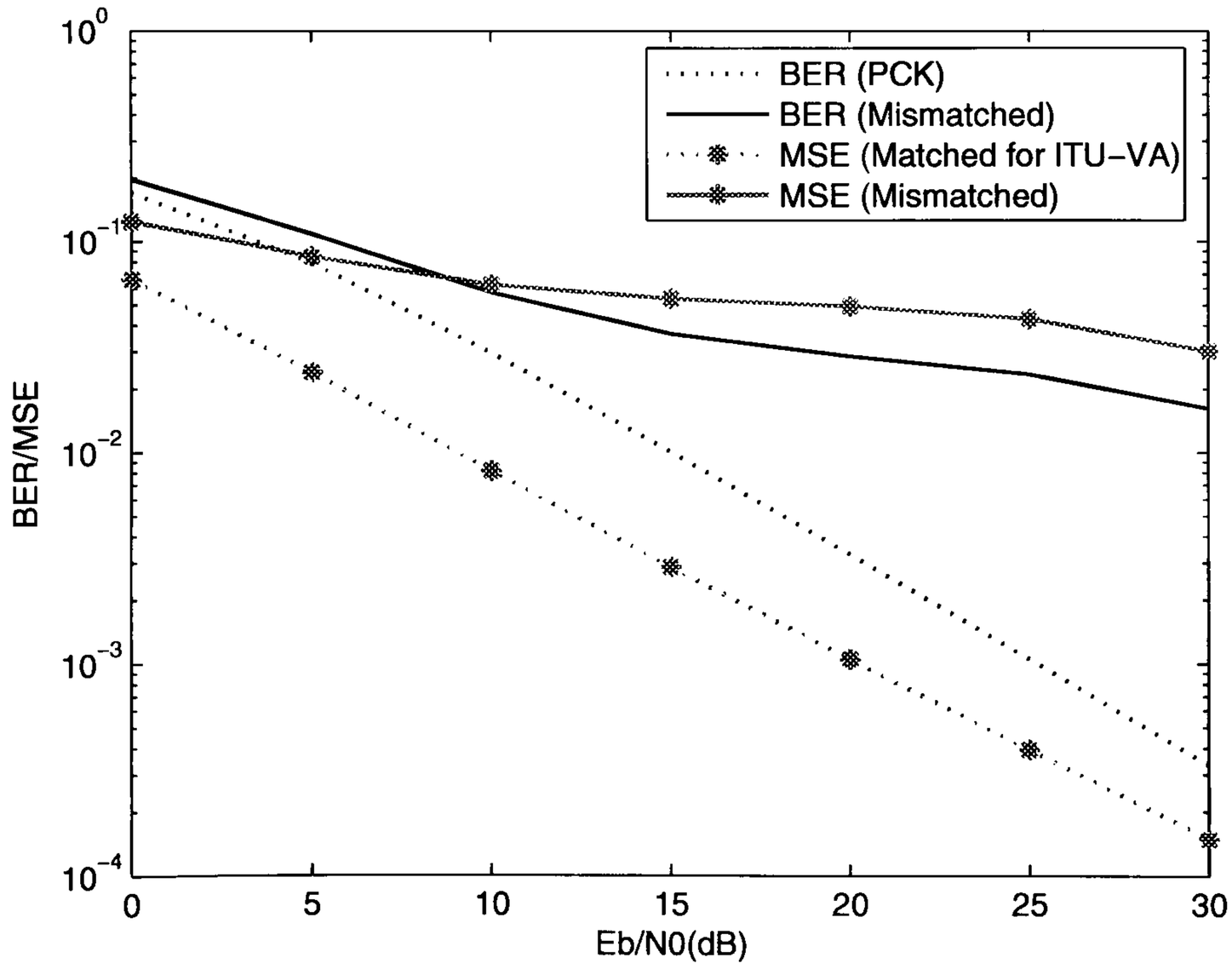


Figure 6.34: MSE and BER performance of mismatched channel estimation: users moving (ITU-VA); estimator optimized for stationary users (ITU-IB) (DL FUSC, uncoded)

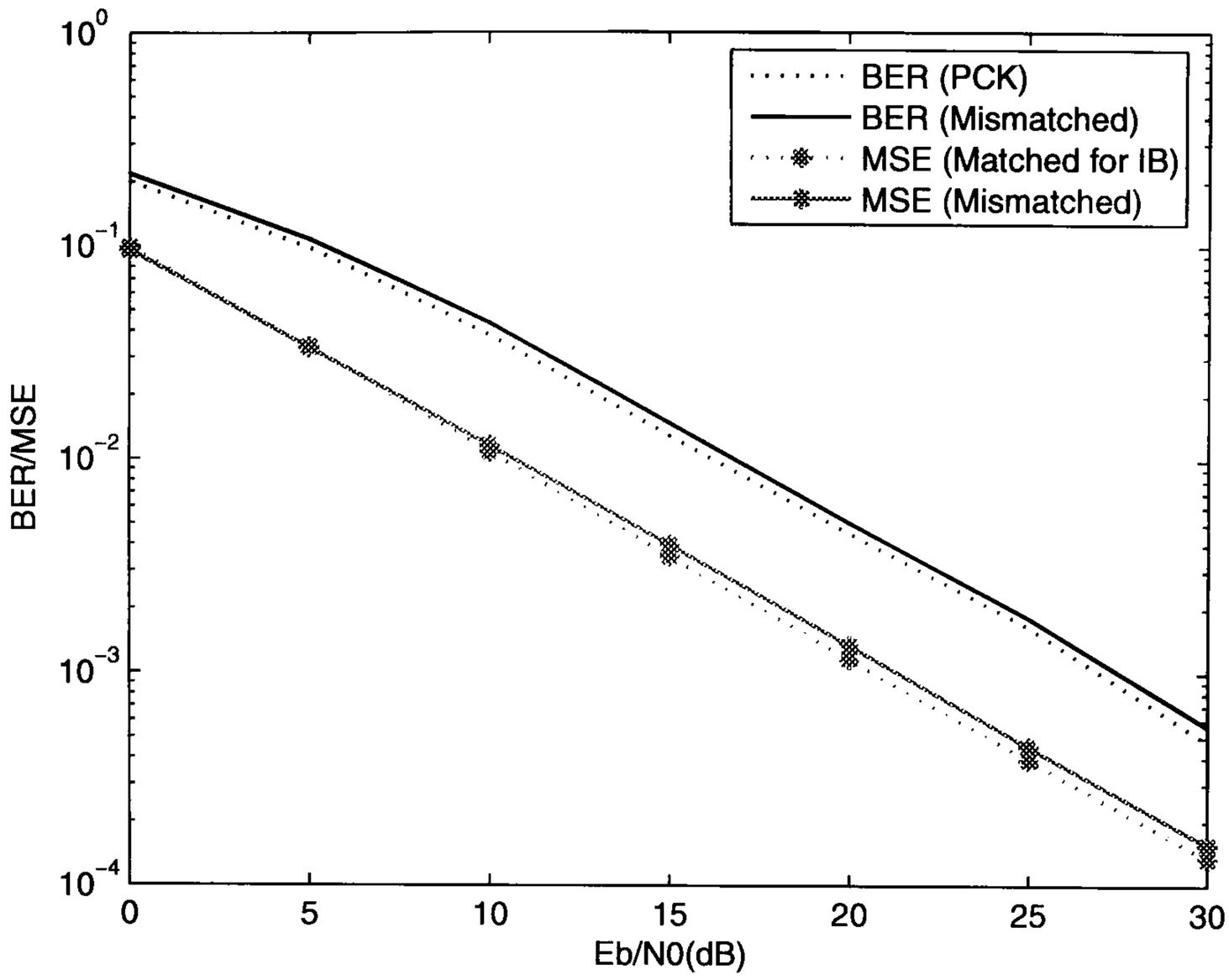


Figure 6.35: MSE and BER performance of mismatched channel estimation: users short delay (ITU-IB); estimator optimized for long delays (ITU-PB) (UL PUSC, uncoded)

(UL PUSC) and Fig. 6.36 (DL FUSC) respectively. It is shown that BER is not affected although there is a small degradation in MSE. The performance in the opposite situation is shown in Fig. 6.37 (UL PUSC) and Fig. 6.38 (DL FUSC), where both MSE and BER suffer from a large degradation, especially in DL.

III. Conclusion

Therefore, it is believed that if the delay spread or fading rate of the channel are unknown, the estimator should be optimized for the worst case that are likely to occur: fast fading rate (e.g., ITU-VA) or longest delay (e.g., ITU-PB). The performance in these cases will at least be as good as that of a matched estimator on a channel having the fastest fading rate or highest delay spread.

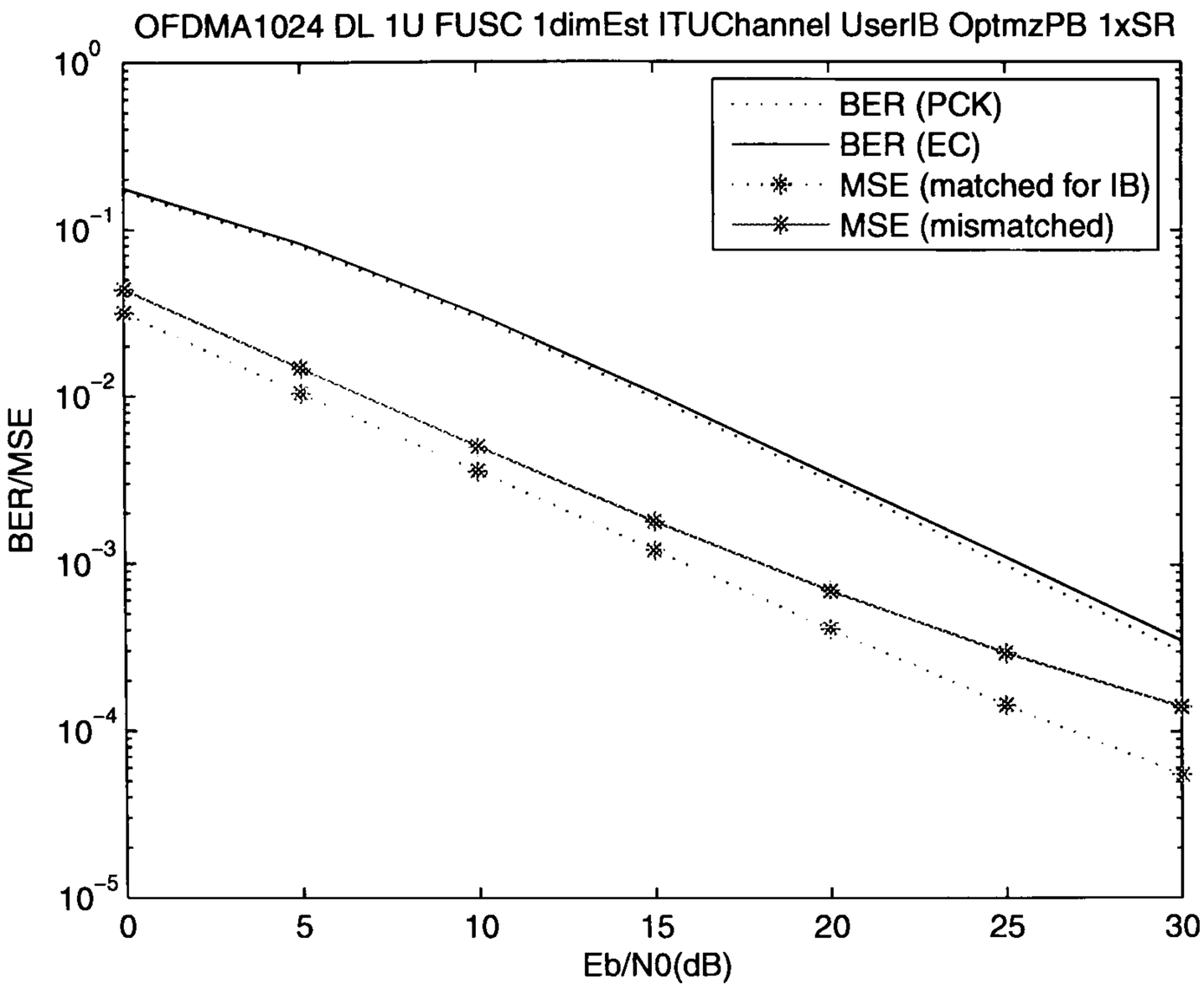


Figure 6.36: MSE and BER performance of mismatched channel estimation: users short delay (ITU-IB); estimator optimized for long delays (ITU-PB) (DL FUSC, uncoded)

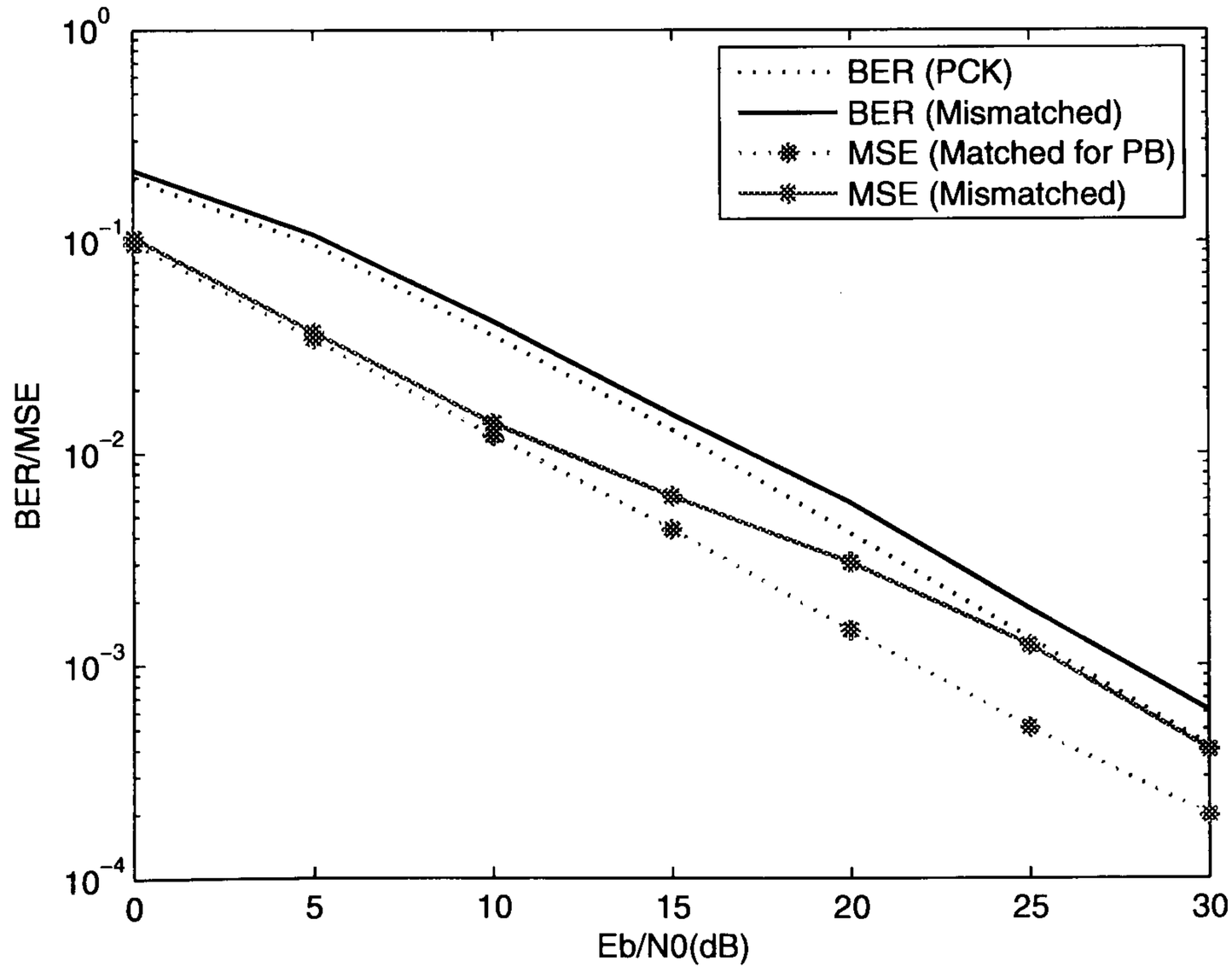


Figure 6.37: MSE and BER performance of mismatched channel estimation: users long delay (ITU-PB); estimator optimized for short delays (ITU-IB) (UL PUSC, uncoded)

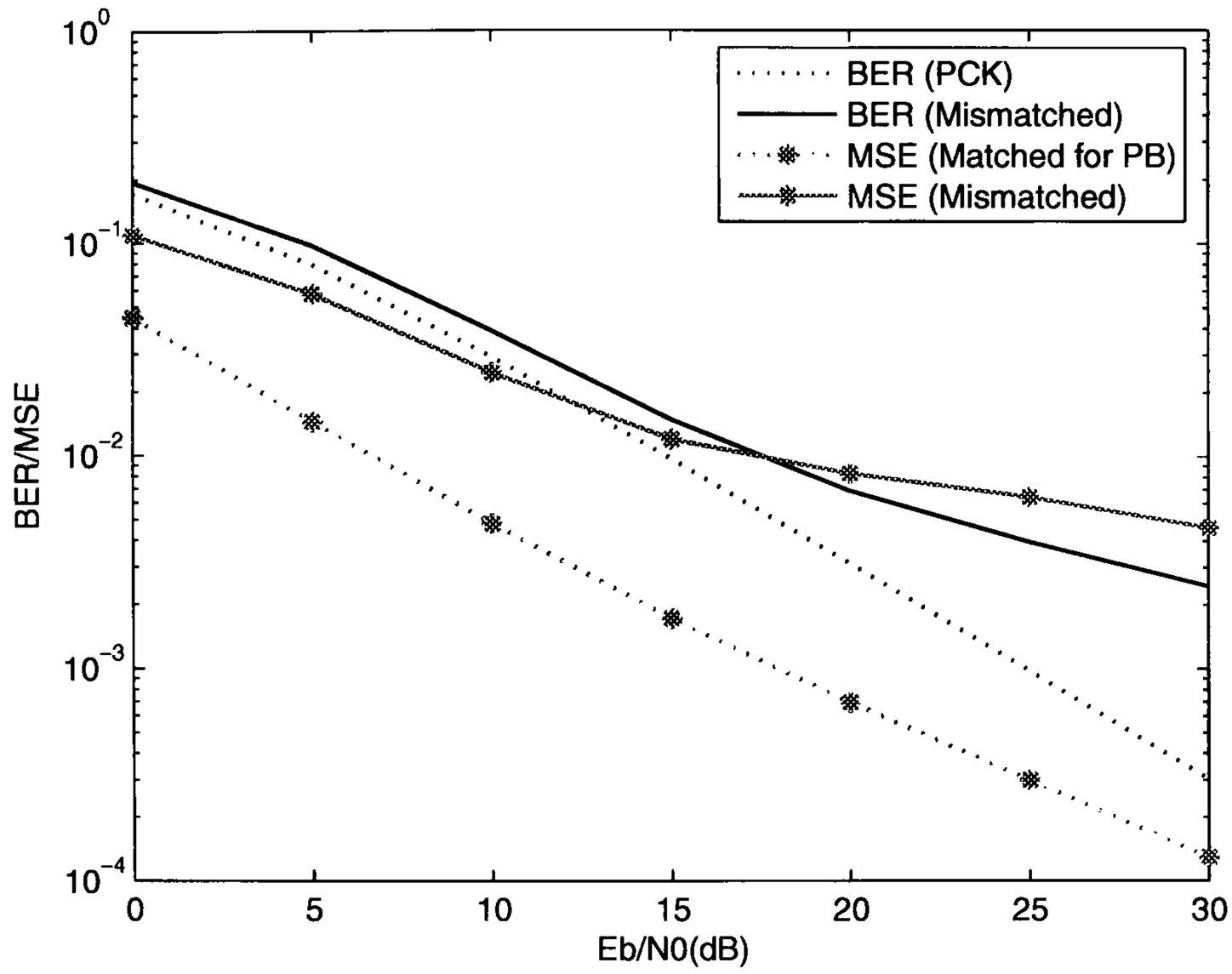


Figure 6.38: MSE and BER performance of mismatched channel estimation: users long delay (ITU-PB); estimator optimized for short delays (ITU-IB) (DL FUSC, uncoded)

6.3 Timing Errors and Frequency Offset in OFDMA

Before an OFDM receiver demodulates the subcarriers, it needs to complete at least two synchronization tasks. One is timing, which is to find out where the symbol boundaries. Optimal timing instants are to minimize the effects of ICI and ISI. The other is the carrier frequency offset of the received signal, which has to be estimated and corrected because it will introduce ICI. This section will investigate the performance of OFDMA with timing errors and frequency offsets. A timing recovery technique for DL OFDMA will also be introduced.

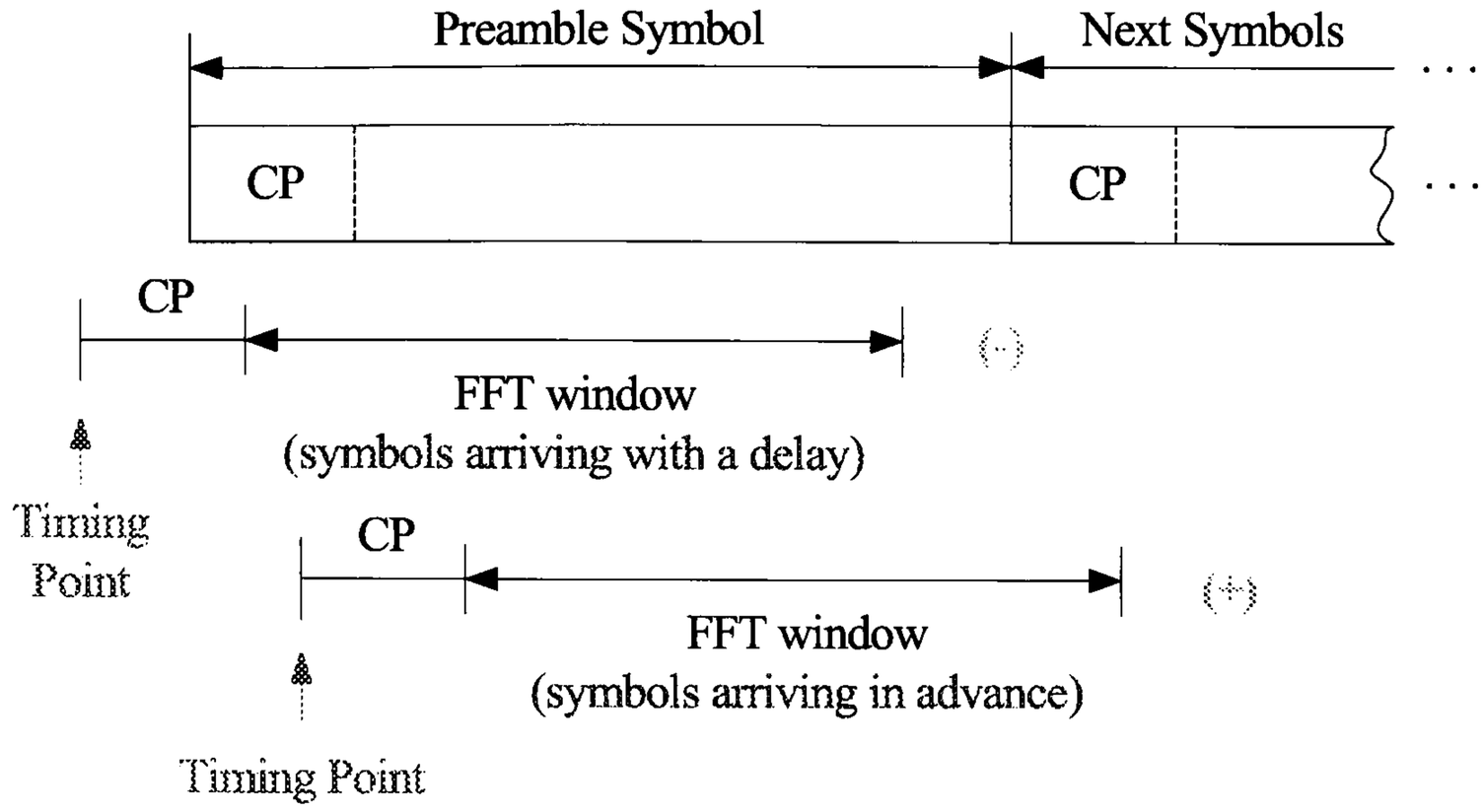


Figure 6.39: Timing offsets in DL OFDMA

6.3.1 Performance of OFDMA with Timing Errors

Generally speaking, OFDM is robust to timing errors, as long as the symbol timing offset varies within an interval equal to the guard time without causing ICI and ISI.

The timing of the OFDMA frame can be obtained by taking the advantage of the DL preamble symbol. There are two types of timing errors depending on the arriving time of the received symbol, which may either be earlier or later than the FFT window starting point (Fig. 6.39). If the preamble is delayed, timing will be earlier than the actual frame start, and hence the FFT window will move forward, either within the CP interval or further into the previous symbol. The other way round, if the preamble arrives earlier, the FFT window will definitely move backwards into the next symbol. ICI and ISI occur only when the FFT interval extends over the symbol boundary. Therefore, it is expected that timing would be worse in the earlier case than the delayed case. This is because even if the received symbols are delayed, as long as the delay is less than the length of CP, FFT is still integrated within a symbol period, but in the earlier case, the FFT will integrate over a period extended to the next symbol, where ICI and ISI may exist.

Fig. 6.40 and Fig. 6.41 show the BER performance of UL PUSC and DL FUSC with

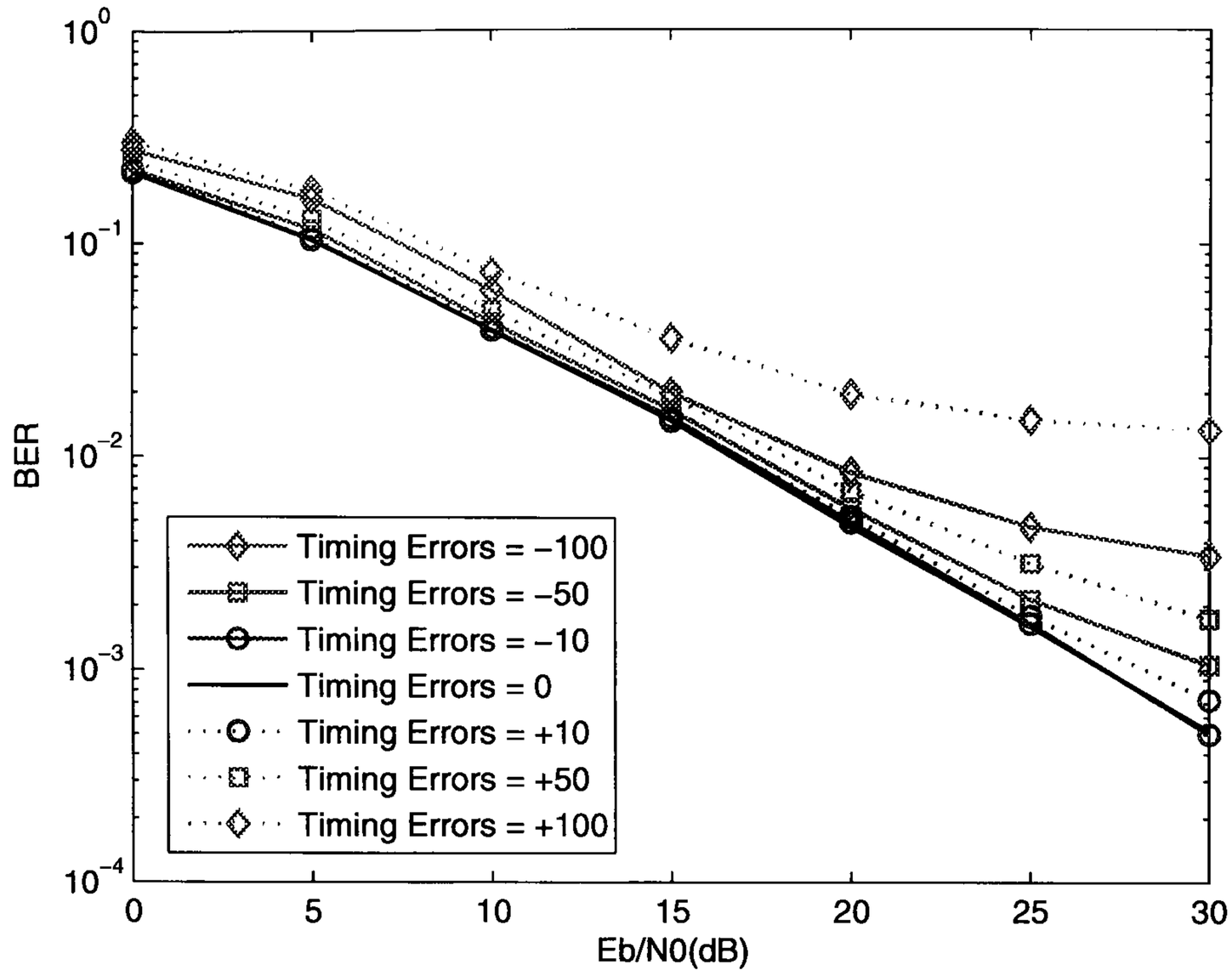


Figure 6.40: BER performance of timing errors in UL PUSC (ITU-PB)

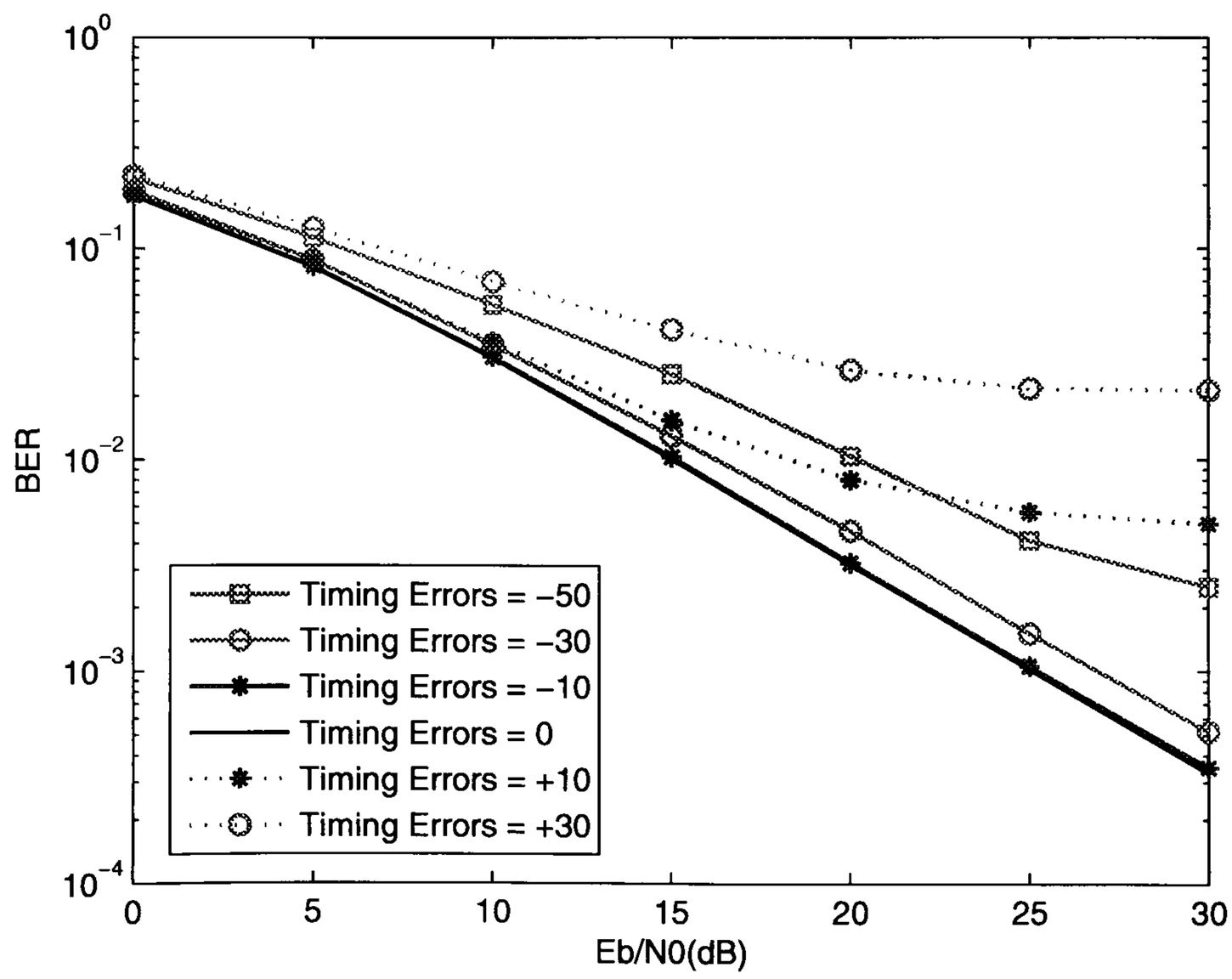


Figure 6.41: BER performance of timing errors in DL FUSC (ITU-PB)

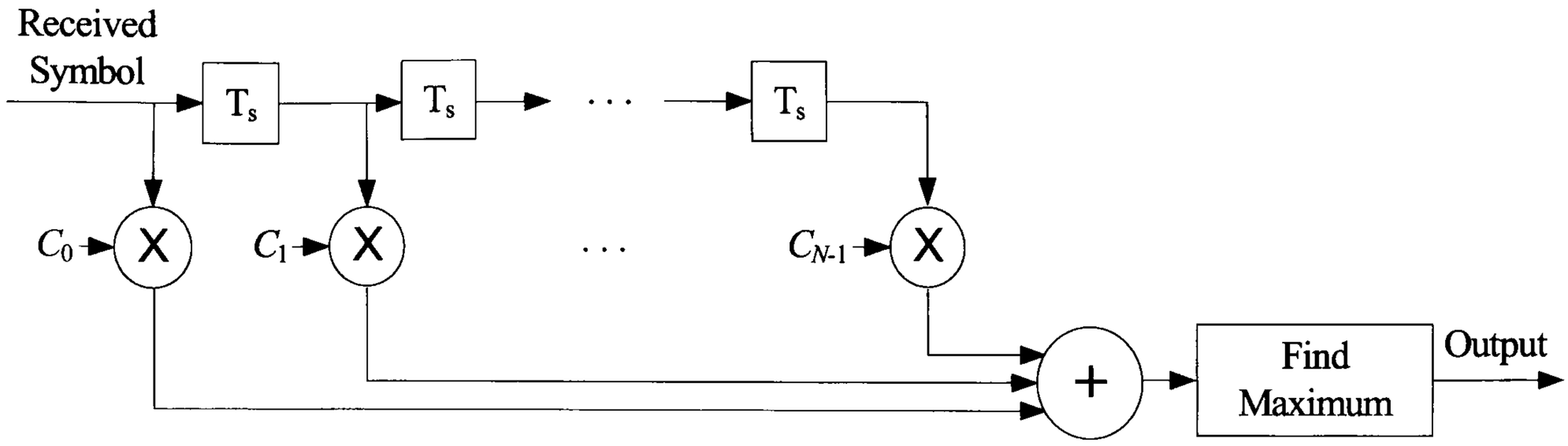


Figure 6.42: Matched filter [3]

various levels of timing errors in ITU-PB. '-' denotes the delayed case, while '+' refers to the earlier case. Timing errors are calculated as the number of samples. It is clear that in UL, the BER performance is not affected by a symbol delayed by within 10 samples, but is sensitive to an earlier arrival of 10 samples. DL is much more vulnerable to timing errors than UL, especially in the earlier case, where a high error floor occurs when the symbol arrives 10 samples in advance.

6.3.2 Timing Recovery with Preamble in DL

As mentioned above, timing recovery of an OFDMA frame can be obtained by using the preamble symbol in DL which is known to the receiver. Fig. 6.42 shows a block diagram of a matched filter for correlating the received OFDMA signal with the known preamble symbol [3]. Note that the matched filter correlates with the OFDM time domain signal, before removing CP and performing FFT at the receiver. The known preamble symbol used for obtaining correlations is also in its time-domain expression, which is after IFFT and adding CP. Here, T_s is the OFDM sampling interval and C_i are the matched filter coefficients, which are the complex conjugates of the known preamble symbol; N is the number of samples in an CP extended OFDMA symbol.

In an AWGN channel, the correlation peak in the output of the matched filter is unique and

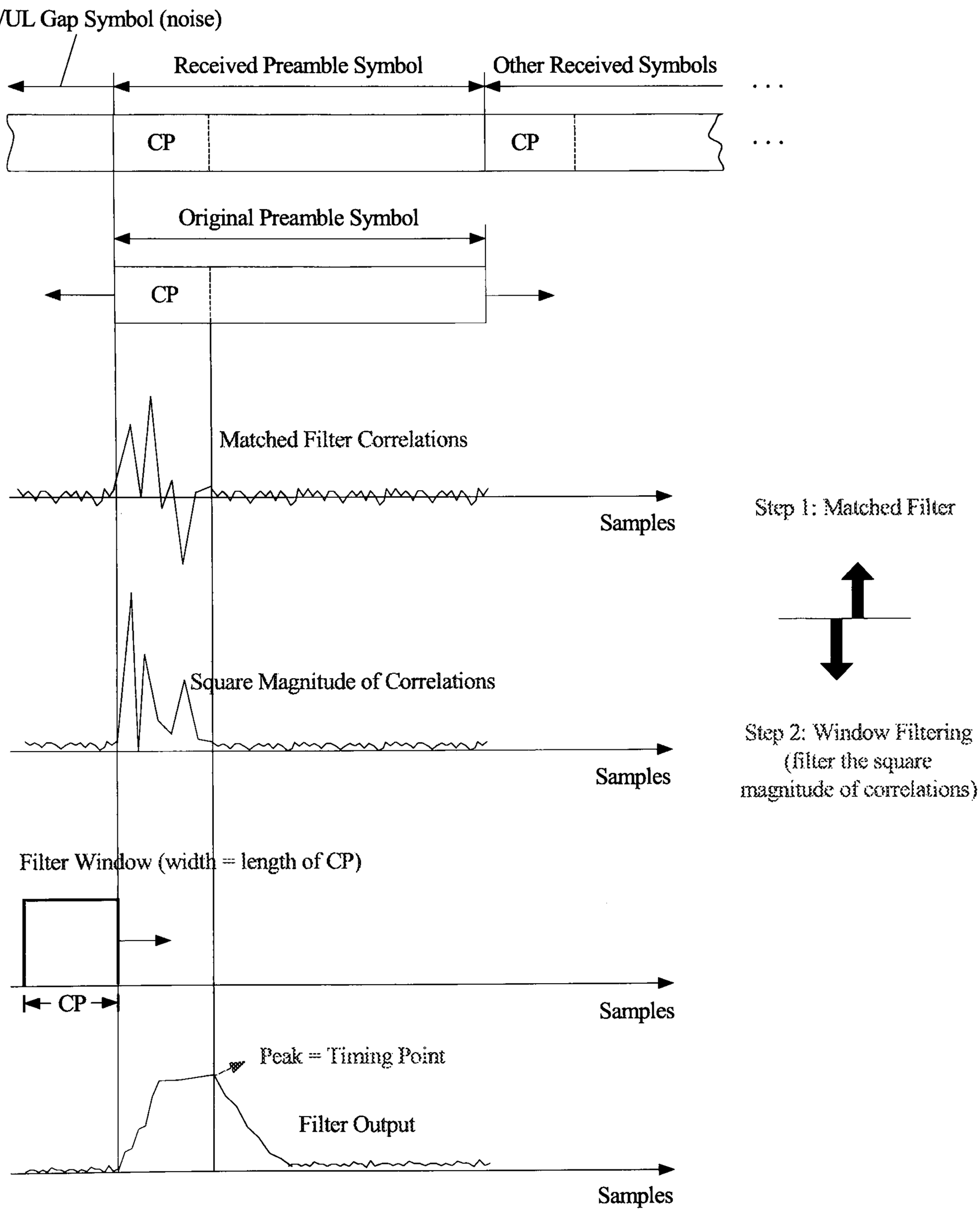


Figure 6.43: Timing with preamble in DL FUSC OFDMA sub-frame over multipath channels

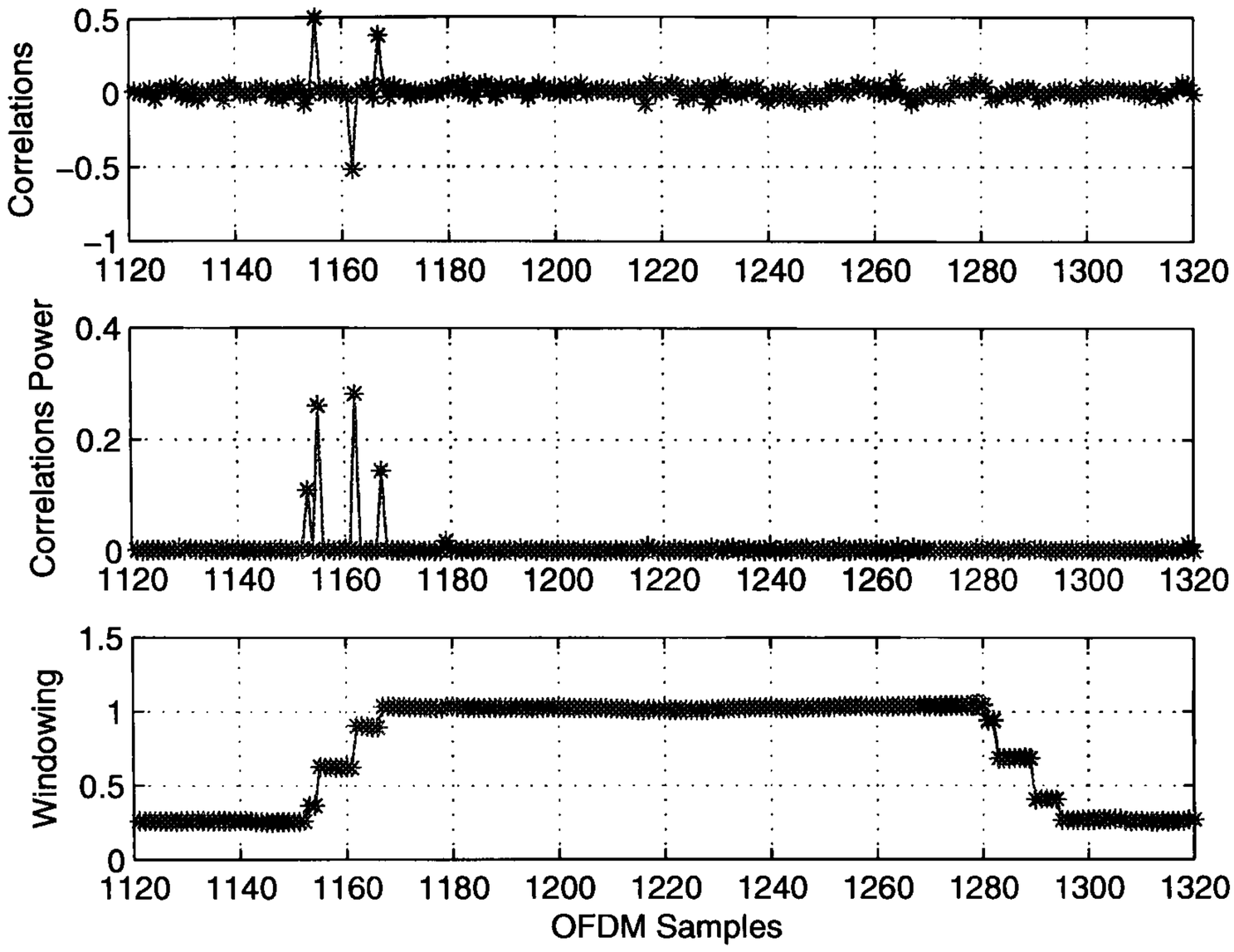


Figure 6.44: Timing in DL FUSC OFDMA at E_b/N_0=0 dB (ITU-PB)

hence indicates the estimated timing point, while other correlation outputs are like noise. However in a fading channel, more than one correlation peaks will appear in the output of the matched filter, the number of which depends on the number of multipaths. Therefore, instead of finding the maximum correlation output only, it is to find the maximum power in the whole guard interval. The process is illustrated in Fig. 6.43 and described as follows.

First of all, take the square magnitude of the output of the matched filter correlations. Then filter them by a window, the width of which is the same as CP duration. The peak position in the filter output is where the FFT window starts. Timing is obtained by left shifting from the peak position a number of samples that is equal to the length of CP. Note that a gap symbol exists between OFDMA frames, which means no information is transmitted before the preamble symbol and only noise at the receiver.

Fig. 6.44, 6.45 and 6.46 show the results of timing recovery in DL OFDMA over ITU-PB channel at different noise values (E_b/N_0=0 dB, 10 dB and 20 dB), from a random trial

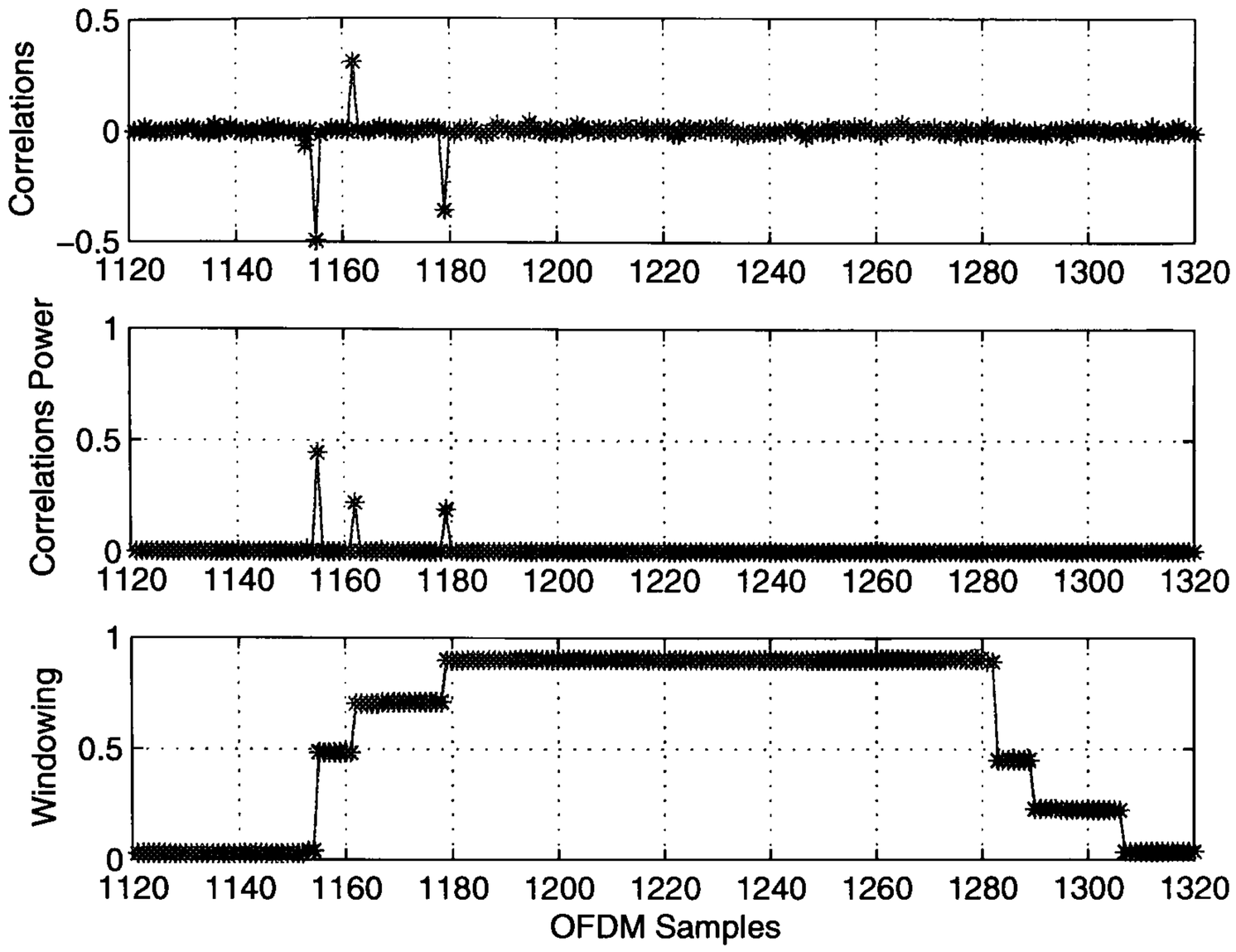


Figure 6.45: Timing in DL FUSC OFDMA at E_b/N_0=10 dB (ITU-PB)

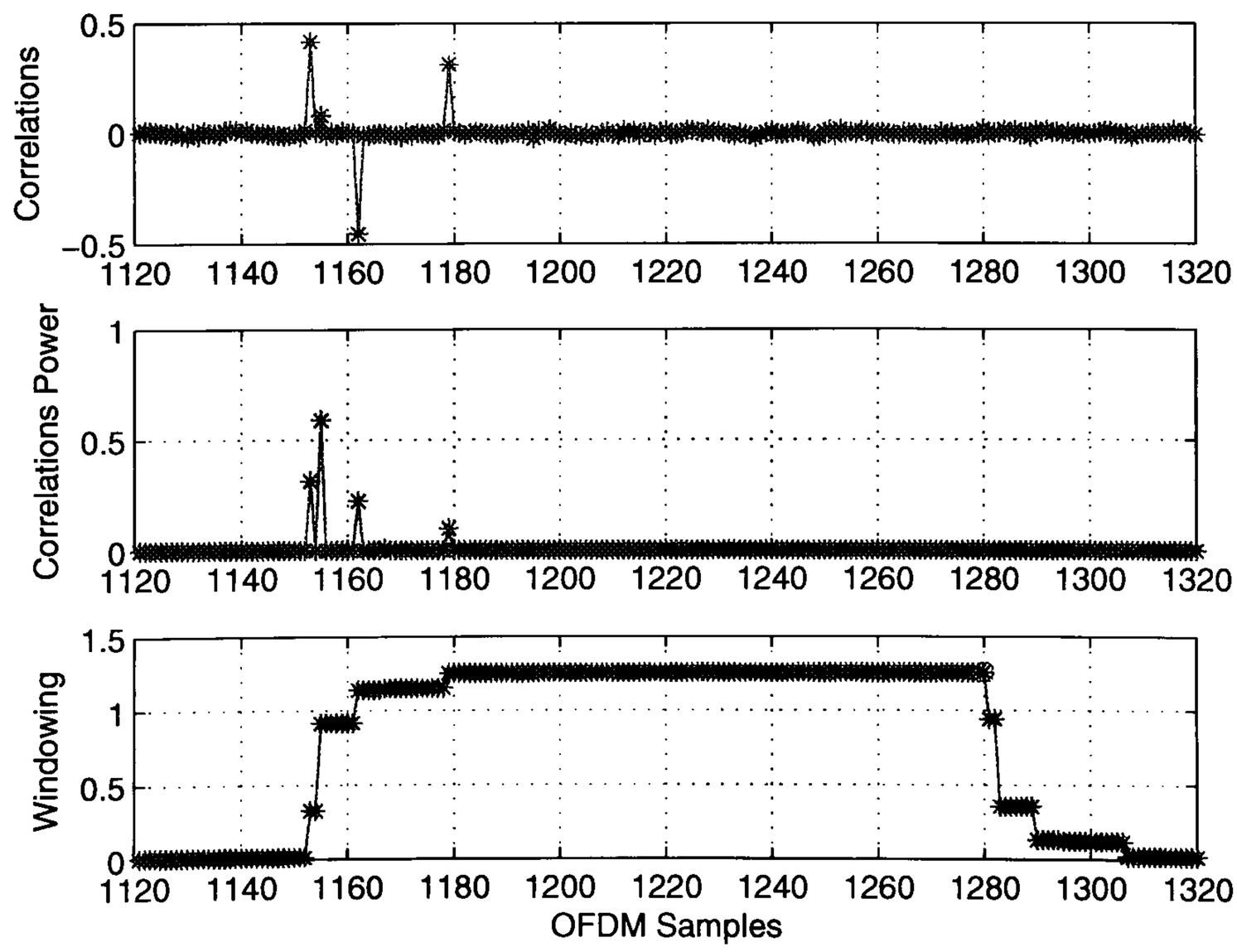


Figure 6.46: Timing in DL FUSC OFDMA at E_b/N_0=20 dB (ITU-PB)

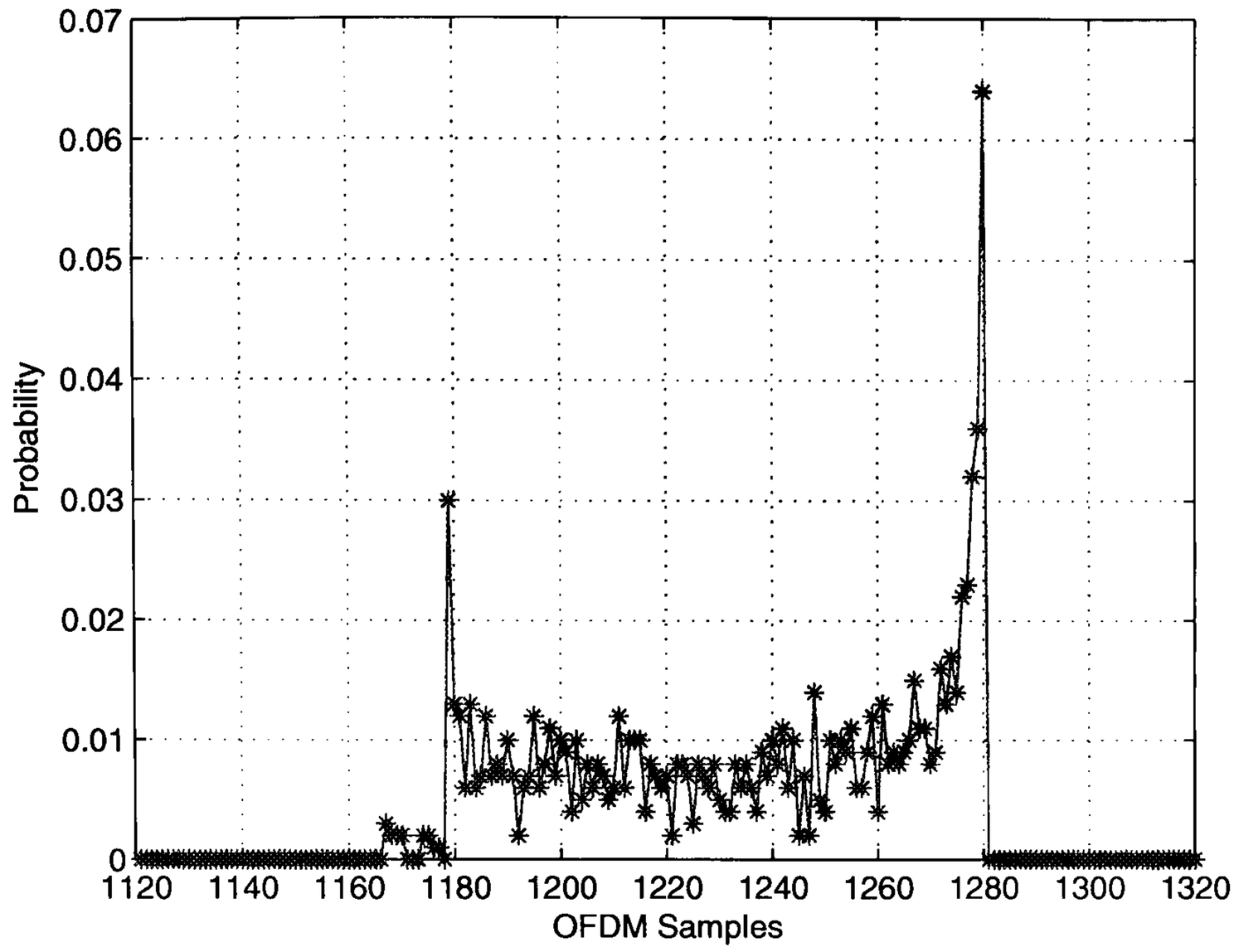


Figure 6.47: Probability of samples being timing points in DL FUSC OFDMA at E_b/N_0=0 dB (ITU-PB)

for one frame. According to the standard frame structure, the length of CP is 128 for a FFT-1024 OFDMA symbol. The simulation starts with the received gap symbol (1152 samples), the received preamble symbol (1152 samples) and then received data symbols. To make the results more clearly, the results only show 200 samples from 'sample 1120' to 'sample 1320'. The top sub-figures are the correlations between the transmitted preamble symbol and the received signal. The middle sub-figures are the square magnitudes of the above correlations. The bottom sub-figures are the output of the windowing over the power of correlations, the maximum values of which are marked with a circle. In practice, it is impossible to know beforehand where the timing point is, but the simulation is set up and the estimated maximum value is expected to appear at 'sample 1280' (1152+128), which is the same as seen from Fig. 6.44 and 6.46. Therefore, subtracting the length of CP, the FFT window starts from 'sample 1152'.

The above results are obtained from one random run. The statistical probabilities can be used to measure the accuracy of the introduced timing recovery scheme at the lowest

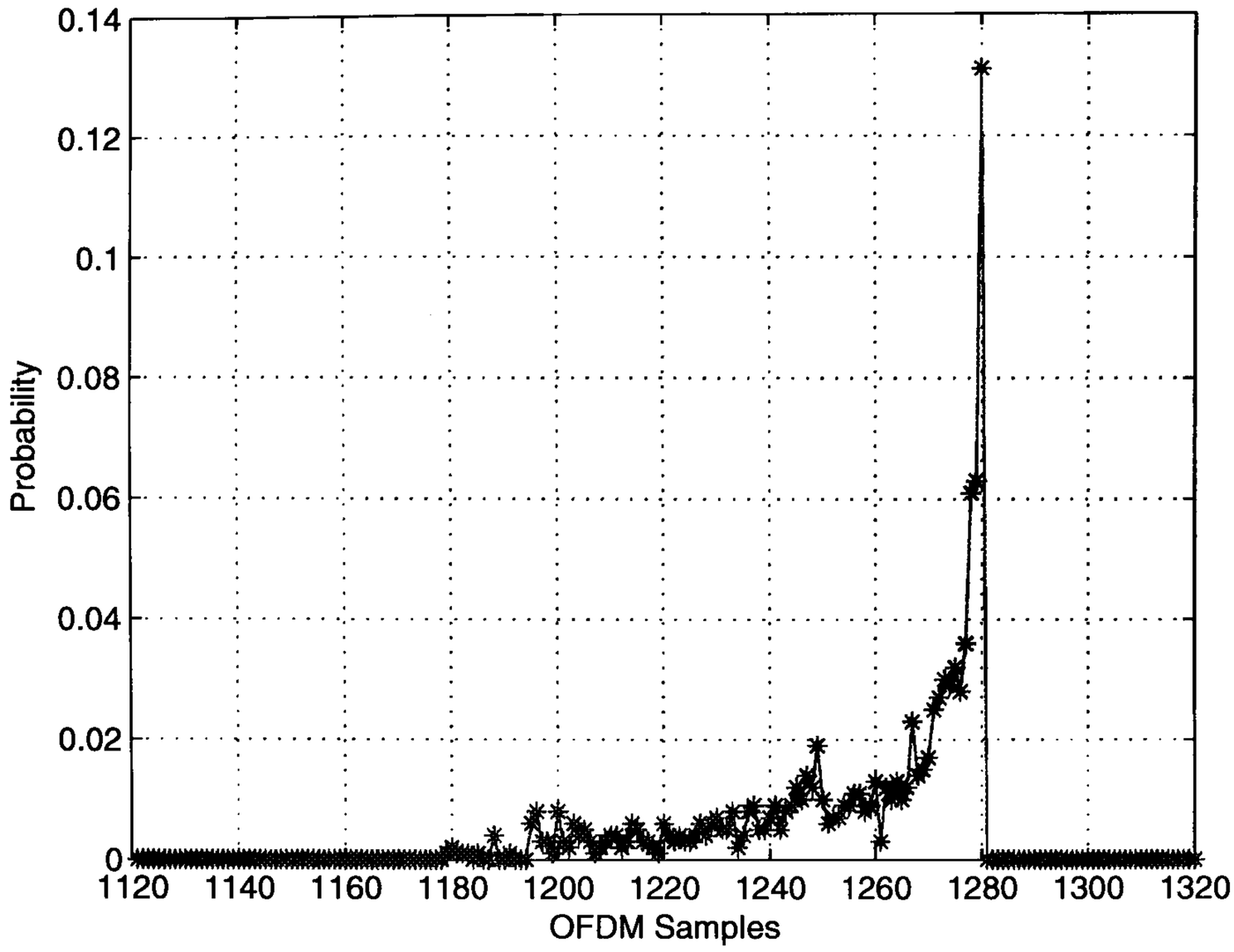


Figure 6.48: Probability of samples being timing points in DL FUSC OFDMA at E_b/N_0=10 dB (ITU-PB)

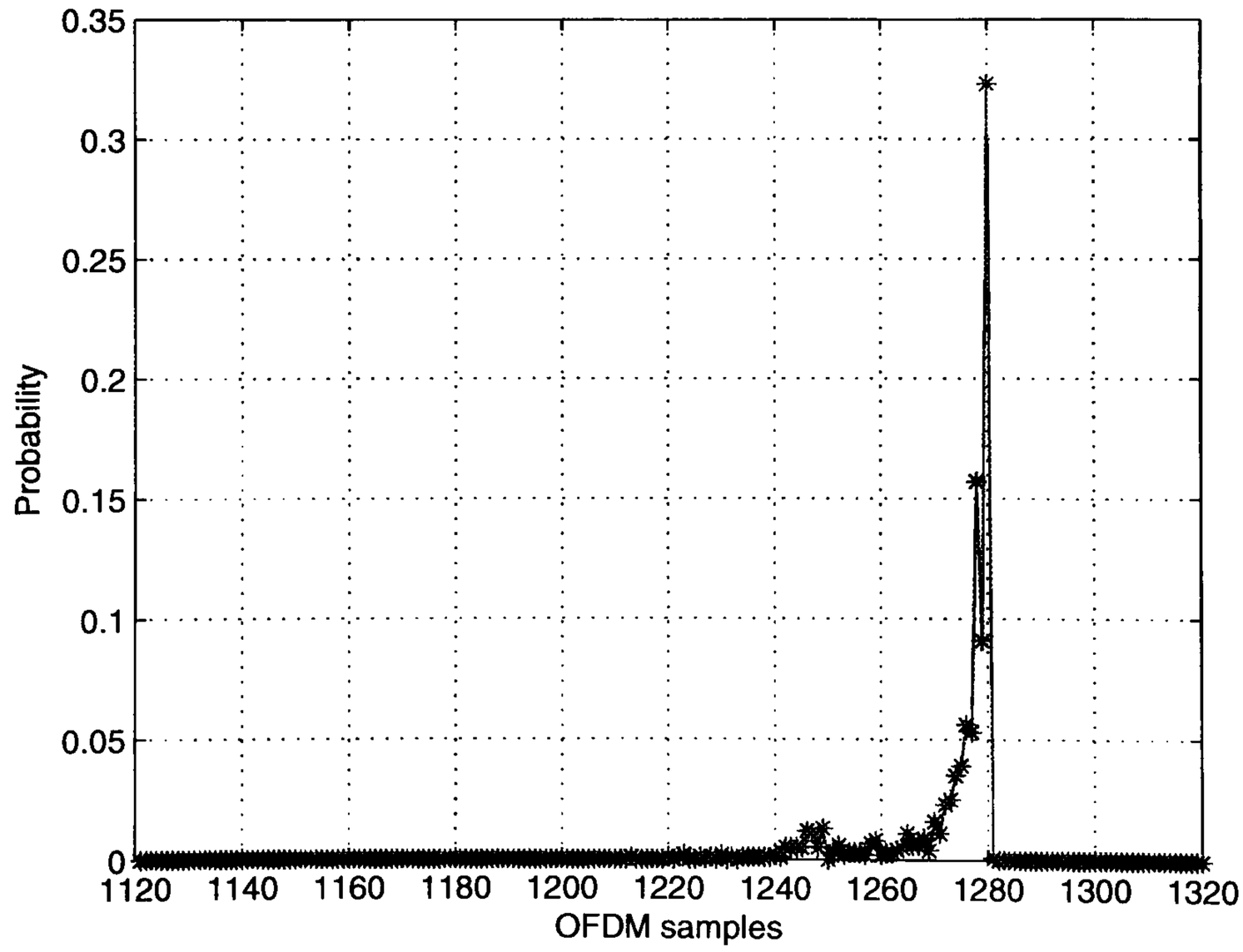


Figure 6.49: Probability of samples being timing points in DL FUSC OFDMA at E_b/N_0=20 dB (ITU-PB)

requirement that it does not affect the performance of UL PUSC or DL FUSC systems. Here, at each noise level, the probability of each symbol being the maximum is calculated over 1000 trials. The results are shown in Fig. 6.47 (E_b/N_0=0 dB), 6.48 (E_b/N_0=10 dB) and 6.49 (E_b/N_0=20 dB). The figures illustrate that at $E_b/N_0 = 0dB$, the probability of the peak appearing at sample index 1280 is 6.4%, and increases to 13.1% at $E_b/N_0 = 10dB$ and 32.3% at $E_b/N_0 = 20dB$. Note that in most cases, the maximum values appear no later than 'sample 1280' (the expected timing point). This means by using the above timing approach, errors occur mainly due to an early timing. According to the delayed cases in Fig. 6.40 and Fig. 6.41, errors within 30 samples will not affect the performance in both UL PUSC and DL FUSC. The probabilities of samples 1250-1280 being the maximum add up to 43.3%, 68.1%, 91.9% at $E_b/N_0 = 0dB$, $E_b/N_0 = 10dB$ and $E_b/N_0 = 20dB$, respectively. Therefore, this timing recovery scheme is more effective at higher E_b/N_0 values, and at around 20 dB, timing can be obtained, without degradation in the performance of DL FUSC and UL PUSC, at an accuracy of 91.9%.

6.3.3 Performance of OFDMA with Frequency Offset

In an OFDM link, the subcarriers are perfectly orthogonal only if the frequency remains the same at the transmitter and receiver. The OFDM system is sensitive to frequency offset, since the frequency offset will give rise to a degradation in the received SNR as well as ICI, which is one of the disadvantages of OFDM compared to single-carrier systems. Supposing the original sample value is a, the sample value with a frequency offset of f_o will be

$$a\exp(j2\pi f_o k T_s) = a\exp(j\varphi k T_s) \tag{6.40}$$

where kT_s ($k = 0, 1, ...N_s + N_{cp} - 1$) are the sampling intervals and φ is the common phase error.

The tolerance of carrier frequency offset specified in the IEEE 802.16 standard [2] is 2%

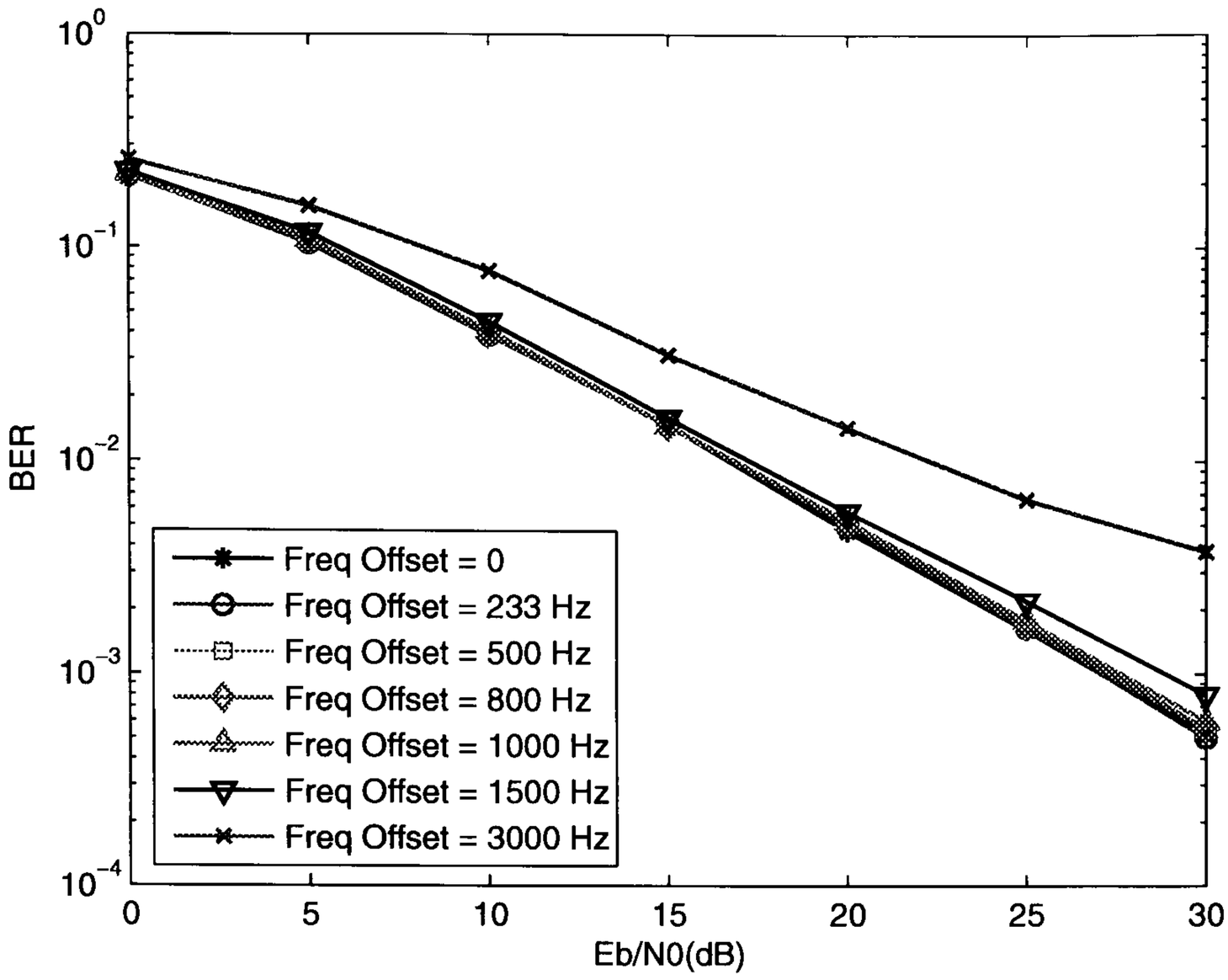


Figure 6.50: BER performance of UL PUSC (ITU-PB) with frequency offset

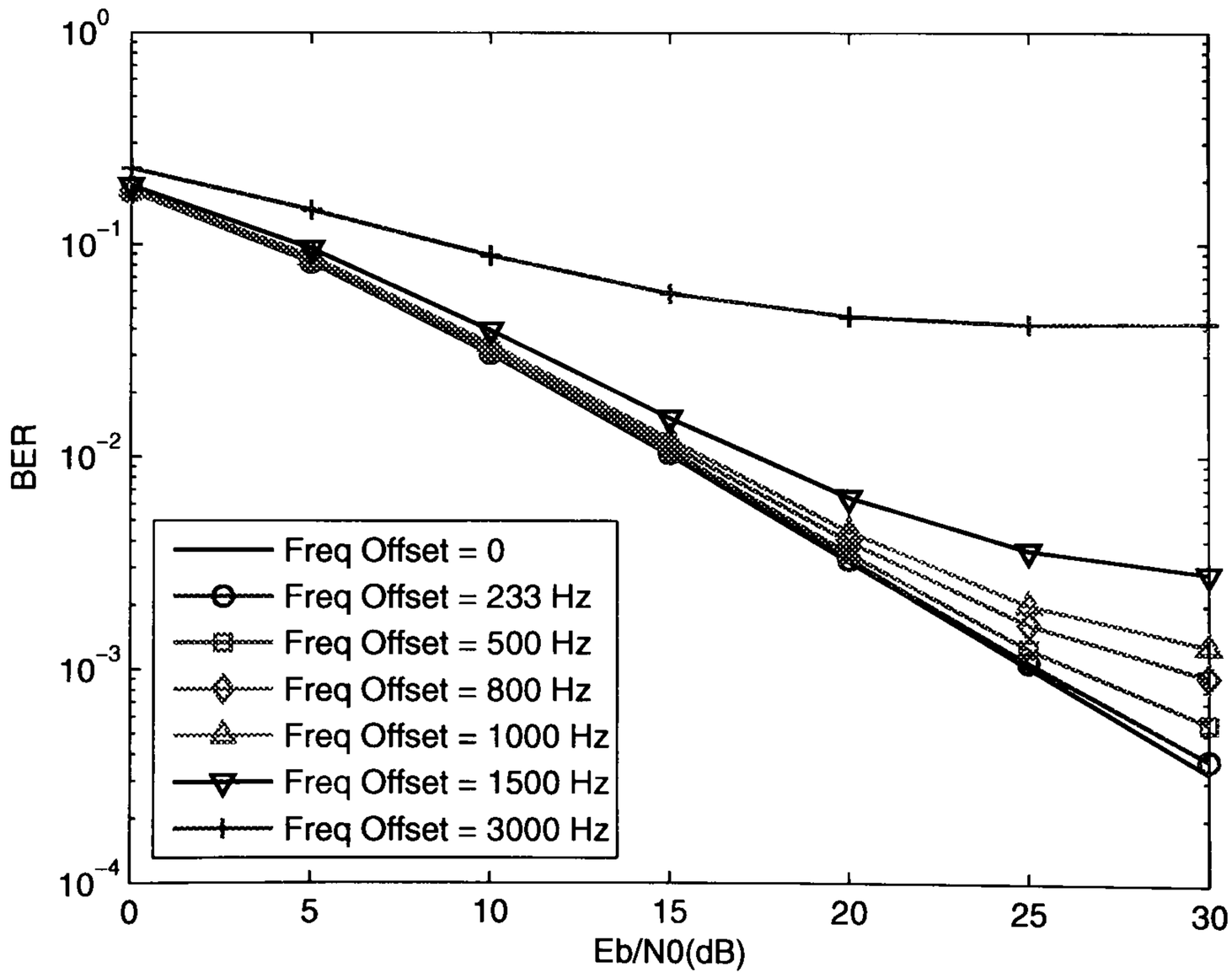


Figure 6.51: BER performance of DL FUSC (ITU-PB) with frequency offset

of the subcarrier spacing, which is

$$11.16(kHz) \times 2\% = 233Hz \tag{6.41}$$

Fig. 6.50 and Fig. 6.51 show the results of OFDMA UL PUSC and DL FUSC in ITU-PB with various carrier frequency offset including the specified value of 233 Hz. The results show that carrier frequency offset within the specified 233Hz does not cause any additional degradation, but an error floor will occur when the frequency offset is over 1500 Hz in UL PUSC and 500 Hz in DL FUSC. Therefore, frequency synchronization is not necessary within the standard defined tolerance value.

6.4 Summary

This chapter has provided the practical performance of channel estimation in IEEE 802.16 scalable OFDMA physical layer over typical ITU-R channels. To improve the performance of channel estimation in OFDMA, the code-aided iterative channel estimation technique has been applied. In both UL and DL, the soft information from the Log-MAP decoder is exploited and used for the channel estimation or channel tracking in the next iteration. Then the renewed channel estimates are sent back for the decoder to obtain more accurate output. In this way, the decoder and the estimator can help each other in an iterative manner. Besides, the BER degradation due to timing errors and frequency offsets are measured. Since timing error is a source of BER deterioration, a timing approach is described and its accuracy at different noise levels are shown.

Chapter 7

Conclusions and Future Work

Contents

7.1 Summary of Work

This thesis discussed two issues regarding the application of code-aided iterative techniques to OFDM systems.

One issue is about canceling the ICI existing in the decoding process of G_4 SFBC-OFDM due to channel frequency variations within a block. Our solution applies a soft PIC approach based on the conventional MF decoding to iteratively subtract the interference from the initial output of the MF decoder. The E_b/N_0 gain from the first iteration is about 5 dB at the BER of 10^{-3} over an uniformly distributed multipath channel with a

channel order of 12, compared to the conventional decoding. Noticing that an error floor still exists at the high E_b/N_0 region due to the non-optimal initial output of the MF decoder, an outer convolutional code is added and the soft information from the Log-MAP decoder is fed back to improve the initial values, so as to make the inner soft PIC work more efficiently. The inner and outer iterations are repeated as many times as needed until improvement from previous iteration is negligible. Considering the tradeoff between the performance and the complexity, 4 inner iterations and 3 outer iterations are chosen. Although most gain comes from the outer coding, with 4 inner iterations, 3 outer soft iterations can still provide about 0.7 dB gain at the BER of 10^{-6} compared to the case without outer iterations.

The other issue is to investigate the practical performance of channel estimation in IEEE 802.16 scalable OFDMA physical layer in various typical ITU channels, which also involves measuring the BER deterioration due to timing errors and frequency offsets. For conventional pilot based channel estimations, the degradation in BER is around 1.9 dB in UL PUSC (2×1D), 1.2 dB in UL PUSC (2D), 2.1 dB in OPUSC (2D), and 0.5 dB in DL FUSC (2×1D) at the BER of 10^{-4}. The code-aided iterative estimation approach can provide about 1 dB, 0.4 dB and 1.3 dB gain at the BER of 10^{-4} after 4 iterations for UL PUSC (2×1D), UL PUSC (2D) and OPUSC (2D) respectively. As the non-iterative estimation in DL FUSC can already provide enough accuracy, it is not necessary to apply iterations with extra complexity. In DL PUSC, a combined code-aided iterative channel tracking and estimation scheme is applied according to its special pilot structure. For the FCH/MAP symbol, the improvement from 3 iterations is more than 4 dB at the BER of 10^{-3}, giving an estimation degradation less than 1 dB compared to PCK. For user data symbols, a gain of about 2 dB is obtained at the BER of 10^{-4} after just one iteration, and the BER performance of estimation is less than 0.5 dB away from that of PCK. In the cases of mismatched channel estimations, the conclusion is that as long as the estimator is optimized for the worst cases that are likely to occur, the performance is at least as good as that of the matched estimator for those worst cases.

Regarding timing errors, it is demonstrated that DL FUSC is in general more vulnerable than UL PUSC. In the worst case for DL FUSC, a high error floor occurs even with an timing error of less than 10 samples. A timing recovery scheme using preamble symbol in DL is approved to be effective, and it is possible to obtain timing at an accuracy of 91.9% at $E_b/N_0 = 20dB$, without degradation in the BER performance of both DL and UL. As for frequency offsets, the BER performance is sensitive to frequency offset over 1500 Hz in UL PUSC and over 500 Hz in DL FUSC, but does not degrade within an offset of 233 Hz, which is specified in the standard.

7.2 Summary of Contributions

This section lists the original work contributed by this thesis.

1. In Chapter 5, the performance of STBC-OFDM and SFBC-OFDM over sample-by-sample time variant and frequency selective fading channels is compared. It is concluded that STBC-OFDM is relatively sensitive to channel time variations between OFDM symbols, but it is resistant to the frequency response variations between OFDM sub-channels. On the other hand, SFBC-OFDM is easily affected by both time and frequency response variations. However, the total influence to SFBC-OFDM by both variations is less than that to STBC-OFDM by a single time variations under the same fading rate of the channel.

2. In Chapter 5, a MF based 'soft PIC' approach is proposed for ICI cancelation in G_4 SFBC-OFDM over quasi-static frequency selective fading channels. The interference existing in the output of the MF decoder is stepwise removed by iterations. In our simulations, 'soft PIC' obtains 2 dB more gain than that from 'hard PIC' at the BER of 10^{-3}. In addition, 'soft PIC' shows much lower error floors than 'hard PIC' at high E_b/N_0 regions.

3. The performance of 'soft PIC' is affected by the sub-optimal initial output of the MF decoder, which explains why error floors still exist after several iterations. Therefore Chapter 5 proposes a code-aided outer iterative PIC for G_4 SFBC-OFDM. In this way, the inner PIC works more efficiently based on the corrected initial values from the outer coding. In the process of outer iterations,the SISO Log-MAP decoder is deployed to provide soft information, which can obtain 0.2 dB more gain after 3 iterations at the BER of 10^{-6}, compared to that from the conventional SIHO Viterbi decoding.

4. Chpater 6 investigates the practical performance of the the physical layer of WiMAX technology, according to the IEEE 802.16 OFDMA standard with scalability. specifically, the degradation from channel estimation that affects BER is measured, and code-aided iterative channel estimations and tracking is proposed for UL and DL receivers. Also, the estimators should be optimized for the worst cases if the channel property is unknown. Finally, the performance of the system with timing errors and frequency offsets is tested. For imperfect timing, an optimal timing recovery technique is described.

7.3 Future Work

Some suggestions on possible future work based on this thesis are given below:

1. Chapter 5 proposes an interference cancelation scheme for SFBC-OFDM over frequency selective channels that remain static during one OFDM symbol period. Channel estimation can be considered in such systems. Moreover, channels may also vary fast even within a symbol, i.e., samples by samples, which is called a double-selective fading channel. Not only interference cancelation in such channels is more difficult, but the channel estimation also becomes much more challenging.

2. The iterative PIC in Chapter 5 may not be efficient enough by fixing the number of iterations. The convergence behavior of PIC needs to be investigated, by which means, an optimum number of iterations can be chosen.

3. The simulation of WiMAX in our thesis does not include standard MIMO antenna configurations. It is worthwhile to further investigate the practical performance of the standard scalable OFDMA over MIMO channels, e.g., channel estimation in SFBC-OFDMA. Of course, ICI will again exist and interference cancelation based on channel estimates in such systems will become a complex issue. Using the iterative approach proposed in this thesis, there could be three types of iterations around the SFBC-OFDMA detector: inner PIC iterations, outer PIC iterations and channel estimation iterations. It may arouse some interest to find out which type of iteration affects the performance the most, and a tradeoff between the performance and the complexity needs to be determined.

Publications

- Yu Zhang, A. G. Burr, "Iterative Channel Estimation in IEEE 802.16 OFDMA Uplink PUSC and OPUSC Transmission Systems", accepted by IEEE Mobile WiMAX'07, Orlando, Florida, USA, March 2007.

- Yu Zhang, Ji Zhang, A. G. Burr, "Turbo-Based Interference Cancelation for SFBC-OFDM Systems", presented at IEEE International Conference on Communications (ICC'06), Istanbul, Turkey, June 2006.

- Yu. Zhang and A. G. Burr, "Parallel Interference Cancelation Detection with Inner and Outer Iterations for Space-frequency Block Coded OFDM Systems", presented at IEEE Personal, Indoor, and Mobile Radio Communications Conference (PIMRC'05), Berlin, Germany, September 2005.

- Yu. Zhang and A. G. Burr, "Outer Iterative Parallel Interference Cancelation Detection for Orthogonal Space-time Block Coding over Time-selective Fading Channels", presented at IEEE Vehicular Technology Conference (VTC'05-Fall), Dallas, Texas, USA, September 2005.

- Yu. Zhang and A. G. Burr, "Diversity realization of orthogonal space-frequency block coded OFDM system over time and frequency selective fading channels", presented at IEEE Wireless Personal Multimedia Communications Conference (WPMC'05), Alborg, Denmark, September 2005.

- Yu. Zhang and A. G. Burr, "Performance improvement of space-frequency block

coded OFDM system in time and frequency selective fading channels", presented at the Postgraduate Symposium on the Convergence of Telecommunications, Networking and Broadcasting (PGNET2005), Liverpool, UK, June 2005.

Bibliography

[1] A.G. Burr, *Modulation and coding for wireless communications*, Prentice Hall, 2001.

[2] "Air interface for fixed broadband wireless access systems", in *IEEE STD 802.16-2004*, October 2004.

[3] R.V. Nee and R. Prasad, *OFDM for Wireless Multimedia Communications*, Artech House, 2000.

[4] Hassan Yagoobi, "Scalable OFDMA Physical Layer in IEEE 802.16 Wireless-MAN", in *Intel Technology Journal*, August 2004, vol. 08.

[5] "Guidelines for Evaluation of Radio Transmission Technology for IMT-2000", in *Recmmendation ITU-R M.1225*, 1997.

[6] R.R. Mosier and R.G. Clabaugh, "KINEPLEX, a bandwidth-efficient binary transmission system", *AIEE Transactions*, vol. 76.

[7] J. Guey, M. Fitz, M. Bell, and W. Kuo, "Signal design for transmitter diversity wireless communication systems over Rayleigh fading channels", in *IEEE VTC*, 1996, pp. 136–140.

[8] Bolcskei H. and A.J. Paulraj, *Multiple-input multiple-output (MIMO) wireless systems*, chapter in The Communications Handbook, pp. 90.1–90.14, CRC Press, 2002.

[9] B. Ozbek, D. Le Ruyet, and Bellanger M., "On Space-Frequency Block Codes for Unequal Channels", in *2nd COST273 Workshop on Broadband Wireless Local Access*, Paris, May 2003.

[10] Alamouti S.M., "A simple transmit diversity technique for wireless communications", in *IEEE Journal on Selected Areas in Communications*, October 1998, vol. 16, pp. 1451–1458.

[11] V. Tarokh, H. Jafarkhani, and A.R. Calderbank, "Space-time block coding for wireless communications: Performance results", in *IEEE Journal on Selected Areas in Communications*, March 1999, vol. 17, pp. 451–460.

[12] R.H. Clarke, "A statistical theory of mobile radio reception", *Bell System Technical Journal*, vol. 47, pp. 957–1000, 1968.

[13] P.A. Bello, "Characterization of randomly time-variant linear channels", *IEEE Transactions on Communication Systems*, vol. 11, no. 4, pp. 360–393, 1963.

[14] A.G. Burr, "Narrow band modulation and coding schemes for radio LAN", in *Proceedings of 4th European Conference on Radio Relay Systems*, Edinburgh, October 1993, pp. 51–56.

[15] W.C. Jakes, *Mobile microwave communication*, Wiley, 1973.

[16] M. Patzold, "On the statistical properties of deterministic simulation models for mobile fading channels", *IEEE Transaction on Vehicular Technology*, vol. 47, no. 1, February 1998.

[17] A.J. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm", *IEEE Transaction on Information Theory*, vol. 13, no. 2, pp. 260–269, 1967.

[18] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding: turbo codes", in *Proceedings of IEEE International Conference on Communications*, Geneva, Switzerland, 1993, pp. 1064–1070.

[19] G. White, *Optimised turbo codes for wireless channels*, Ph.D thesis, University of York, 2001.

[20] C. Berrou, "The ten-year-old turbo codes are entering into service", *IEEE Communications Magzine*, vol. 41, pp. 110–116, August 2003.

[21] A.G. Burr, "Turbo-codes: the ultimate error control codes?", *IEEE Electronics and Communication Engineering Journal*, vol. 13, no. 4, pp. 155–165, August 2001.

[22] D. Divsalar and F. Pollara, "Turbo codes for PCS applications", in *Proceedings of International Conference on Communications*, Seatle, USA, June 1995.

[23] L.R. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimising symbol error rate", *IEEE Transactions on Information Theory*, vol. 20, pp. 284–287.

[24] P. Robertson and P. Hoeher, "Optimal and sub-optimal maximum a posteriori algorithms suitable for turbo decoding", *European Transactions on Telecommunications*, vol. 8, no. 2, pp. 119–125.

[25] B. Le Floch, R. Lassalle, and D. Castelain, "Digital sound broadcasting to mobile receivers", *IEEE Transactions on Consumer Electronics*, vol. 3.

[26] U. Reimers, "The European perspective on digital television broadcasting, part I: conclusions of the working group on digital television broadcasting (WGDTB)", in *Proceedings of HDTV World '93*, Las vegas, USA, April 1993.

[27] J. Barsellotti and B. Langlais, "Paddling upstream: overcoming problems in the CATV return path", in *19th Montreausx Television Symposium*, June 1995.

[28] I.K. Czajkowski, "High-speed copper access: a tutorial overview", in *Electronics and Communication Engineering Journal*, 1999, vol. 11, pp. 125–148.

[29] IEEE802 standards, "Local and metropolitan area networksłSpecific requirementsłPart 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specificationsłAmendment 1: High-speed Physical Layer in the 5 GHz band", 1999.

[30] IEEE802 standards, "Local and metropolitan area networksłSpecific requirementsł-Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specificationsłAmendment 4: Further Higher-Speed Physical Layer Extension in the 2.4 GHz Band", 2003.

[31] C. Eklund, R.B. Marks, K.L. Stanwood, and S. Wang, "IEEE standard 802.16: a technical overview of the wirelessMAN air interface for broadband wireless access", in *IEEE Communications Magazine*, June 2002, pp. 98–107.

[32] S.J. Vaughan-Nichols, "OFDM: Old Technology for New Market", http://www.wi-fiplanet.com/tutorials/article.php/1500641, November 2002.

[33] J.C-I Chuang, "The effects of time delay spread on portable radio communication channels with digital modulation", in *IEEE Selected Areas in Communications*, June 1987, pp. 879–889.

[34] A.G. Burr, "Performance Analysis of COFDM for Broadband Transmission on a Fading Multipath Channel", in *Wireless Personal Communications*, June 1999, vol. 10, pp. 3–17.

[35] J.G. Proakis, *Digital Communications*, McGraw-Hill, 1995.

[36] R.W. Chang, "Synthesis of band-limited orthogonal signals for multichannel data transmission", in *Bell System Technical Journal*, 1966, vol. 45, pp. 1775–1796.

[37] B.R. Slatzberg, "Performance of an efficient parallel data transmission system", in *IEEE Transactions on Communication Technology*, 1967, vol. COM-15, p. 805.

[38] N. Weste and D.J. Skellern, "VLSI for OFDM", in *IEEE Communications Magazine*, 1998, vol. 36, pp. 127–131.

[39] V. Tarokh, H. Jafarkhani, and A.R. Calderbank, "Space-time codes for high data rate wireless communication: Performance criterion and code construction", in *IEEE Transactions on Informaiton Theory*, March 1998, vol. 44, pp. 744–765.

[40] N. Seshadri and J. Winters, "Two signaling schemes for improving the error performance of frequency-division-duplex (FDD) transmission systems using transmitter antenna diversity", in *Int. J. Wireless Information Networks*, 1994, vol. 1, pp. 49–60.

[41] V. Tarokh, H. Jafarkhani, and A.R. Calderbank, "Space-time block codes from orthogonal designs", in *IEEE Transactions on Information Theory*, July 1999, vol. 45, pp. 1456–1467.

[42] H. Jafarkhani, "A quasi-orthogonal spaceCtime block code", in *IEEE Transactions on Communications*, January 2001, vol. 49, pp. 1–4.

[43] D. Divsalar, M.K. Simon, and Raphaeli D., "Improved parallel interference cancellation for CDMA", in *IEEE Transactions on Communications*, February 1998, vol. 46, pp. 258–268.

[44] "White Paper: Mobile WiMAX - Part I: a technical overview and performance evaluation", in *WiMAX Forum*, February 2006.

[45] "Air interface for fixed and mobile broadband wireless access systems", in *IEEE P802.16e/D12*, February 2005.

[46] O. Edfors, M. Sandell, J. Beek, S.K. Wilson, and P.O. Borjesson, "OFDM Channel estimation by singular value decomposition", in *IEEE Transactions on Communications*, July 1998, vol. 46.

[47] H. Wymeersch, F. Simoens, H. Steendam, and M. Moeneclaey, "Code-aided channel tracking for OFDM", in *International Symposium on Turbo Codes and Related Topics*, Munich, Germany, April 2006.

[48] J.K. Cavers, "An analysis of pilot symbol assisted modulation for Rayleigh fading channels", in *IEEE Transaction on Vehicular Technology*, November 1991, vol. 40, pp. 686–693.

[49] L.L. Scharf, "Statistical signal processing: detection, estimation and time series analysis", 1991, Addison-Wesley.

[50] S.M. Kay, *Fundamentals of statistical signal processing: estimation theory*, Prentice Hall, 1993.

[51] M.C. Valenti and B.D. Woerner, "Iterative channel estimation and decoding of pilot symbol assisted turbo codes over falat-fading channels", in *IEEE Selected Areas in Communications*, September 2001, vol. 19, pp. 1697–1705.

[52] A.P. Dempster, N.M. Laird, and D.B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm", in *Journal of the Royal Statistical Society*, 1977, vol. 39, pp. 1–38.