UNIVERSITY OF CALGARY

Maximum Likelihood Differential Sequence Estimation of DS-UWB

by

Yinggan Huang

A THESIS

SUBMITTED TO THE FACULTY OF GRADUATE STUDIES

IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE

DEGREE OF MASTER OF SCIENCE

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

Calgary, Alberta

JANUARY 2009

UNIVERSITY OF CALGARY

FACULTY OF GRADUATE STUDIES

The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies for acceptance, a thesis entitled "Maximum Likelihood Differential Sequence Estimation of DS-UWB" submitted by Yinggan Huang in partial fulfillment of the requirements for the degree of Master of Science.

Supervisor, Dr. John Nielsen
Department of Electrical & Computer Engineering

Dr. Ed Nowicki
Department of Electrical & Computer Engineering

Mr. Norm Bartley
Department of Electrical & Computer Engineering

Dr. Kyle O' Keefe
Department of Geomatics Engineering

Dec 3/08

Date

Abstract

Direct Sequence Ultra-wideband (DS-UWB) technology is a high potential candidate for the next generation of wireless short range high speed communications. In the IEEE 802.15.3a DS-UWB proposal, the classical Rake receiver will suffer severe performance degradation due to the abundant inter-symbol interference (ISI) in the wireless UWB channels. A novel sequence detection technology referred as *maximum likelihood differential sequence estimation* (MLDSE) algorithm is proposed to be an alternative to the classical Rake receive for the DS-UWB scheme in the thesis. The MLDSE algorithm performs both functions of the *maximum likelihood sequence estimation* (MLSE) equalization and *multiple symbols differential detection* (MSDD) reception by a modified *Viterbi algorithm* (VA). Therefore the MLDSE receiver has the simplicity of the MSDD reception and the optimal performance of the MLSE equalization in the ISI and AWGN channels. The performance bound of the MLDSE algorithm is derived for the *M*-DPSK signals in the thesis. The feasibility of the proposal is also considered and analyzed in the thesis. The resource needed by Viterbi algorithm can be reduced greatly by integrating the decoding process of direct sequence encoding into the MLDSE algorithm. An adaptive joint data and channel estimation strategy is proposed to simulate the MLDSE receiver performance in the real UWB channels. The simulations demonstrate the superiority of the MLDSE receiver to the ideal Rake receiver in the UWB channels.

Acknowledgements

The thesis is dedicated to my wife, Jin Wu, who always been there for me and filled my life with love and happiness. The thesis can not come through without the supports from her. I also wish to express my deep appreciation to my supervisor, Dr. John Nielsen, for his patient guidance and valuable advices.

Table of Contents

APPROVAL PAGE II

ABSTRACT II

ACKNOWLEDGEMENTS III

TABLE OF CONTENTS IV

LIST OF TABLES VII

LIST OF FIGURES AND ILLUSTRATIONS VIII

LIST OF ABBREVIATIONS X

ABSTRACT II

CHAPTER 1. INTRODUCTION 1

1.1 History of UWB Technology 1

1.2 Advantages of UWB Communication Technology 4

1.2.1 License exempt 4

1.2.2 Low Channel Fading Margin 5

1.2.3 High Data Rate 6

1.3 Main UWB Communication Schemes in Research 9

1.4 Thesis Objectives 11

1.5 Thesis Contributions 16

1.6 Thesis Outline 17

CHAPTER 2. BASIC SIGNAL DETECTION TECHNOLOGIES 18

2.1 Matched Filter 18

2.2 Differential Detection of *M*-DPSK 20

2.2.1 The Classical Coherent Detection for M-DPSK 22

2.2.2 The Classical Differential Detection of DPSK Signal 23

2.3 Rake Receiver of the DSSS signals 27

2.4 Summary 30

CHAPTER 3. LITERATURE REVIEW 31

3.1 Maximum Likelihood Sequence Estimation of digital sequences in the presence of ISI 32

3.1.1 The Minimum-Distance Criterion 32

3.1.2 The Viterbi Algorithm 34

3.1.3 Performance of Maximum-Likelihood Sequence Estimation 42

3.2 Multiple-symbol differential detection of *M*-PSK 49

3.2.1 Maximum-Likelihood Differential Detection of M-PSK based on a larger than two symbol observation interval 50

3.2.2 Performance of Multiple-Symbol Differential Detection of M-PSK 56

3.3 Summary 61

CHAPTER 4. MAXIMUM LIKELIHOOD DIFFERENTIAL SEQUENCE ESTIMATION OF DS-UWB SIGNALS 62

4.1 The Modulation and Channel Model 63

4.2 The MLDSE receiver scheme 65

4.3 The Adaptive Channel Estimator 75

4.4 Performance of the MLDSE Scheme 77

4.5 The Drawbacks of Rake Receiver in DS-UWB Scheme 82

4.6 Summary 86

CHAPTER 5. SIMULATIONS AND COMPARISONS 91

5.1 The UWB channel responses used in the simulation 93

5.2 The performance of the MLDSE scheme simulation 95

5.2.1 The error rate of the MLDSE scheme in the CM1 channel................. 95

5.2.2 The error rate of the MLDSE scheme in the CM2 channel................. 96

5.2.3 The error rate of the MLDSE scheme in the CM3 channel................. 97

5.2.4 The error rate of the MLDSE scheme in the CM4 channel................. 98

5.3 Summary.. 99

CHAPTER 6. CONCLUSIONS AND FUTURE WORKS................................ 101

6.1 Conclusions 101

6.2 Future works 107

APPENDIX A: DERIVATION OF PAIRWISE PROBABILITY 109

APPENDIX B: SOURCE CODE OF MLDSE FUNCTION............................... 112

REFERENCES 133

List of Tables

Table 1-1: US spectrum allocation for unlicensed use 5

Table 1-2: The comparison of UWB and other PHY transmission technologies 8

Table 4-1: Multipath channel characteristics of IEEE 802.15.3a Channel Models 64

Table 4-2: Diversity gain loss due to the partial energy collection 84

Table 5-1: General parameters used in simulations 92

List of Figures and Illustrations

Figure 1-1: FCC spectral mask in terms of dBm/MHz of EIRP for indoor commercial systems 3

Figure 2-1: Classical matched filter block diagram 19

Figure 2-2: Classical Coherent Detection of *M*-DPSK signal 22

Figure 2-3: Classical Differential Detection of DPSK Signal 23

Figure 2-4: Bit error rates of coherent and differential detection of DPSK 26

Figure 2-5: The implementation of the Rake receiver 28

Figure 3-1: (a) The ISI signal generator. (b) The state transition diagram 36

Figure 3-2: The trellis diagrams of the possible state transition 37

Figure 3-3: A two-state trellis with the transitions branch metrics marked 38

Figure 3-4: An illustrated iterative procedure of Viterbi algorithm. 39

Figure 3-5: Direct implementation of MSDD of *M*-PSK 56

Figure 3-6: Bit error probability versus Eb/No for MSDD of DPSK 60

Figure 4-1: Modulation and channel model of DS-UWB *M*-DPSK signals 63

Figure 4-2: The block diagram of the MLDSE receiver scheme 65

Figure 4-3: (a) The channel model. (b) The state transition diagram 69

Figure 4-4: The trellis diagrams of the possible state transition 70

Figure 4-5: The branch metrics of the MLDSE algorithm 70

Figure 4-6: An illustrated iterative procedure of MLDSE algorithm 72

Figure 4-7: The Adaptive channel estimator 76

Figure 4-8: The upper BER bounds of MLDSE algorithm for DPSK signal 81

Figure 4-9: The discrete-time Rake receiver for the DS-UWB scheme 83

Figure 4-10: The auto-correlation property of a M=12 ternary code 85

Figure 5-1: The channel responses of the four UWB channel models 93

Figure 5-2: The performance of the MLDSE scheme in CM1 channel 95
Figure 5-3: The performance of the MLDSE scheme in CM2 channel 96
Figure 5-4: The performance of the MLDSE scheme in CM3 channel 97
Figure 5-5: The performance of the MLDSE scheme in CM4 channel 99

List of Abbreviations

Symbol	Definition
AWGN	additive white Gaussian noise
BPSK	binary phase shift keying
CDMA	code division multiple access
DPSK	differential binary phase shift keying
DSSS	direct sequence spread spectrum
DS-UWB	Direct Sequence Ultra-Wideband
FCC	Federal Communications Commission
GPS	global position system
ICI	inter-chip interference
I&D	integration and dump
IR	impulse radio
IR-UWB	Impulse Radio Ultra-Wideband
ISI	inter-symbol interference
ITU	International Telecommunications Union
LMS	least mean-squared
LOS	line of sight

LPD	Low probability of detection
MB-OFDM	Multi Band Orthogonal Frequency Division Multiplexing
ML	maximum likelihood
MLDSE	maximal likelihood differential sequence estimation
MLSE	maximal likelihood sequence estimation
MRC	maximal ratio combining
MSDD	multiple symbols differential detection
NLOS	non-line of sight
OFDM	Orthogonal Frequency Division Multiplexing
OOK	on-off keying
PAM	pulse amplitude modulation
PHY	physical
PPM	pulse position modulation
PSD	power spectral density
RLS	recursive least-squares
SNR	signal-to-noise ratio
TH	time hopping
UWB	Ultra-wideband
VA	Viterbi algorithm

WCDMA	wideband code division multiple access
WLAN	wireless local area network
WPAN	wireless personal area network

Chapter 1. Introduction

1.1 History of UWB Technology

Ultra-wideband (UWB) radio is a fast emerging technology with uniquely attractive features in wireless communications, networking, radar, imaging and position systems [1]. Despite its increasing acceptation during the past decade, the history of UWB radio communication can track back to the first experiment of spark-gap transmitter invented by Guglielmo Marconi at the end of the 19^{th} century [2]. In the late sixties, the introduction of UWB radar systems motivated by the high sensitivity to scatters and low power of ultra-short pulses usually in the order of nanoseconds or less was another milestone of UWB technology developments [3]. Low probability of detection (LPD) is an important requirement for military wireless communications systems. As information-bearing pulses with ultra-short duration have ultra wide spectral occupancy, the impulse radio (IR) is also a promising covert technique for tactical military communications due to its low power and duty cycle compared to alternative systems [4, 5]. The U.S. Department of Defense (DoD) coined the term “ultra wideband” for devices occupying at least 1.5 GHz, or a -20 dB fractional bandwidth exceeding 25% of the center frequency. Similar definitions were adopted by the Federal Communications Commission (FCC) in the United States, the resulting First Report and Order (R&O) defined UWB transmission systems with instantaneous spectral occupancy in excess of 500 MHz or a -10 dB fractional bandwidth of more than 20% of the center frequency [6].

The unique advantages of UWB radios have long been appreciated by the radar and communications communities: I) enhanced capability to penetrate through obstacles; II) ultra high precision ranging at the centimeter level; III) potentially for very high data rates along with a commensurate increase in user capacity; and IV) potentially small size and processing power [1];

In February 2002 the FCC essentially unleashed huge "new bandwidth" (3.1-10.6 GHz) and the power spectral density (PSD) is required to below -41.3 dBm/MHz unintentional radiation limit (As a comparison, receiver noise floor is about -174 dBm/Hz), which allows UWB technology to overlay already available services such as the global position system (GPS) and the IEEE 802.11 wireless local area networks (WLANs) that coexist in the 3.1-10.6 GHz band [6].

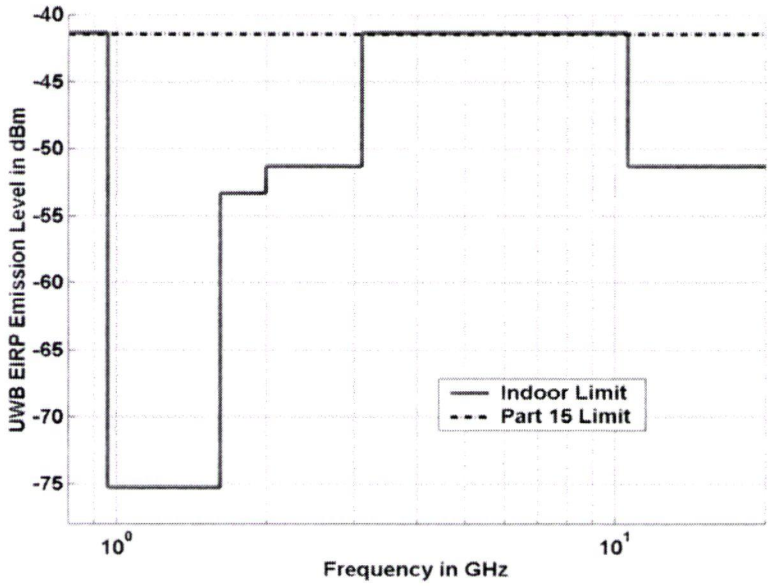

Figure 1-1: FCC spectral mask in terms of dBm/MHz of EIRP for indoor commercial systems

Unprecedented number of institutions responded to this FCC ruling with rapidly growing research efforts targeting a host of exciting UWB applications: short-range very high-speed broadband access to the Internet, covert communication links, localization at centimeter-level accuracy, high-resolution ground-penetrating radar, through-wall imaging, precision navigation and asset tracking, etc [5, 7, 8]. Several UWB radio techniques detailed in the following section 1.3 are competing to be the standards of IEEE802.11 and IEEE802.15, etc. But the commercial UWB deployment seems have to stall a bit due to the difficulty in implementation.

1.2 Advantages of UWB Communication Technology

When UWB radio is used as a communication carrier, some advantages of the technology are unique and attractive comparing with the existing communication schemes.

1.2.1 *License exempt*

The radio spectrum is an international limited resource. The use of this resource is regulated internationally by the Radio Communications Sector of the International Telecommunications Union (ITU), and nationally by the spectrum management authority of each country. To minimize harmful interference between spectrum users, the radio spectrum is segmented into several frequency bands. Each frequency band is allocated by the ITU-R to one or more wireless services on a worldwide or regional basis. The Radio Regulations (RR) of the ITU-R specifies the rules for the assignment and use of each band. The spectrum management authority in each country allocates frequency bands domestically based on its national needs and usually in harmony with the ITU-R frequency allocations.

Wireless systems are mostly licensed. However, some low-power systems are exempt from the requirement of obtaining a license. License-exempt systems have to comply with certain standards and specifications (e.g.; emission limits), and traditionally operate in specific frequency bands such as the Industrial, Scientific and Medical (ISM) bands [9].

The FCC's power requirement of -41.3dB/MHz puts the UWB radio in the category of unintentional radiators, such as TVs and computer monitors. Such power restriction allows UWB systems to operate below the noise floor of a typical narrow band receivers and enables UWB signals to coexist with current radio service with minimal or no interference [10]. This property makes the UWB radio a suitable candidate of license-exempt applications

The following table summarized the US spectrum allocation for unlicensed use [11].

Unlicensed Band	Frequency of Operation	Bandwidth
ISM at 2.4GHz	2.4000-2.4835 GHz	83.5 MHz
U-NII at 5 GHz	5.15-5.35 GHz 5.725-5.825 GHz	300 MHz
UWB	3.1-10.6 GHz	7500 MHz

Table 1-1: US spectrum allocation for unlicensed use

1.2.2 *Low Channel Fading Margin*

The phenomenon known as multipath is unavoidable in wireless communications channels. It is caused by multiple reflections of the transmitted signal from various surfaces such as building, barriers and people. The effect of multipath creates severe issues for narrow band communication. It can cause signal degradation up to -40 dB due to the out-of-phase addition of the line of sight (LOS) and non-line of sight (NLOS) continuous waveforms [10]. It is called frequency selective multipath channel fading.

On the other hand, the using of signals with gigahertz bandwidths means that signals reflected with path length differentials on the order of a foot or less can be resolved and combined, which offers UWB techniques high immunity to multipath cancellation[7]. UWB communication does not suffer intense multipath fading any more. The fading effect can be significantly reduced even in indoor environment [12]. The research results show that when measurement bandwidths of 10 MHz and 1 GHz were used, the fading margins were about 30~40 dB and 3 dB, respectively [13].

1.2.3 *High Data Rate*

One of the major advantages of the large bandwidth for UWB radio is the improved channel capacity. Channel capacity, or data rate, is defined as the maximum amount of data that can be transmitted per second over a communications channel. When following conditions are met: I) source transmitted signal is a Gaussian random process; II) the detection is soft; III) channel noise is AWGN, the Shannon's information capacity formula is given as:

$$C = B\log_2(1+SNR) \tag{1-1}$$

where C represents the maximum channel capacity in bits/Hz, B is the single sided bandwidth, and SNR is the signal-to-noise power ratio. The large channel capacity of UWB communications systems is evident from Shannon's capacity formula (1-1). As shown in above equation, channel capacity C linearly increase with bandwidth B. When source transmitted signal is a series of discrete signals, we can still find that channel capacity C can be increased by increasing the bandwidth B [14]. With the capabilities of several gigahertz of bandwidth available for UWB signals, a data rate of gigabits per second (Gbps) can be implemented [10]. Due to the FCC's power limitation on UWB transmissions, a high data rate is available only for short ranges (up to 10 meters). This makes UWB systems perfect candidates for short-range, high-data-rate wireless applications such as wireless personal area networks (WPANs).

UWB technology is highly anticipated because it provides WPAN connectivity of Bluetooth, but at speeds of up to 1000 times faster or possibly even more. The following table gives a detail about UWB and other PHY transmission technologies [15].

Technology	Data Rate	Range	Cost	Power	Spectrum	Issues
Bluetooth	0.8-1.0 Mb/s	30ft	Low	Low	2.4GHz	Speed and Interference issues
802.11a	54Mb/s	90-100ft	high	high	5.0 GHz	High power consumption, High cost, Bulky chipsets
802.11b	11Mb/s	250–300ft	Medium	Medium	2.4 GHz	Speed and signal strength issues for more range.
802.11g	54Mb/s	100 ft	High	High	2.4 GHz	Connectivity and range problems, High cost
HyperLan	25Mb/s	100 ft	High	High	2.4 GHz	Only European standard, High cost
Home RF	11Mb/s	150 ft	Medium	Medium	2.4 GHz	Speed Issues, does not have big players' support
Zigbee	0.02 – 0.2Mb/s	20 – 25ft	Low	Low	2.4 GHz	Standard still under consideration, very less communication range, law data-rate
UWB	> 1 Gb/s	30 ft	High	Low	3.1–10.6 GHz	High data rate for short range only, technology still not ratified

Table 1-2: The comparison of UWB and other PHY transmission technologies

Although UWB signals can propagate greater distances at higher power levels, current FCC regulations enable high-rate (above 110 Mb/s) data transmission over a short range (less than 15m) at very low power [15]. Major efforts are currently under way by the IEEE 802.15 wireless personal area networks (WPANs) Working Group for standardizing UWB radios for indoor (home and office) multimedia transmission.

1.3 Main UWB Communication Schemes in Research

According to the definition of UWB radio given by Federal Communications Commission (FCC), several communication schemes were discussed in vogue. There are three main communication schemes researched in depth, including Impulse Radio Ultra-Wideband (IR-UWB), Multi Band Orthogonal Frequency Division Multiplexing (MB-OFDM) and Direct Sequence Ultra-Wideband (DS-UWB) [16].

For IR-UWB scheme, the information is transmitted through a series of ultra-short pulses which are typically on the order of a nanosecond; therefore, enabling the energy burst spreading of the radio signal to a few gigahertzs. The ultra-short pulse trains are generally transmitted in the form of pulse position modulation (PPM), pulse amplitude modulation (PAM) or on-off keying (OOK) modulation. In the IR-UWB scheme, the time hopping (TH) modulation is a simple means for spreading the spectrum of these ultra-wide bandwidth low-duty-cycle pulse trains and is a principle multiple access method to accommodate many users [4, 17-25]. In MB-OFDM scheme, the available spectrum is divided into many sub-bands. Each sub-band occupies the frequency spectrum greater than 500 MHz. The multi-carrier modulation employs the frequency division multiplexing technology to split a train of symbols into several orthogonal sub-carriers. Symbols are sent over the sub-channels respectively and resemble at the receiver to recover each sub-carriers date. Signals are transmitted in each band with sufficient time interval with a pre-defined frequency hopping pattern between each sub-carrier to minimize the inter-symbol interference (ISI). The MB-OFDM scheme also exploits the frequency diversity technology which is inherent in the multiple channel transmission by sending a symbol across multiple sub-carries. Therefore the scheme provides more multipath resolvability and more flexibility. The transmission rate can be improved by simply multiplexing more bands at the expense of system complexity [15, 26-28].

The DS-UWB scheme uses the direct sequence spread spectrum (DSSS) technology which is already deployed in other wireless systems to transmit the information bits over several GHz bandwidths. The key property of this scheme is to spread each continuously transmission data with a pseudorandom sequence code C of length N_c to expand the signal over a large bandwidth, therefore each data symbol is multiplied by the specific spreading code to form a transmission sequence of N_c chips. The spreading gain compensates the low level of transmit power required to fit under the FCC power density spectrum mask. In the DS-UWB scheme in order to fully exploit this very large bandwidth frequency resource, the different length spreading codes are utilized according the transmission channel characteristics. So the DS-UWB offers great flexibility and promising ability for very high data rate wireless access [16, 29-40].

Each scheme has its own specific advantages and drawbacks. Due to their flexibility and maturation in technology, the DS-UWB and MB-OFDM schemes are being introduced by the IEEE 802.15.3a Standardization Committee as the principle WPAN PHY layer candidate standards for short range, high data rate wireless applications.

1.4 Thesis Objectives

Among these three UWB radio schemes, the DS-UWB scheme is the research interest of the thesis. The DS-UWB technology is becoming a highly potential candidate for the next generation of wireless short range high speed communications. DS-UWB scheme advantages are derived from the DSSS technique. DSSS technique can reduce the transmitted signal energy spectrum density which is limited by FCC regulations and achieve the maximum bandwidth efficiency among three mentioned UWB radio schemes [27]. To achieve higher data rates, the DS-UWB scheme needs to sacrifice its processing gain (or spreading ratio) by using the shorter spreading codes. For example, in the IEEE 802.15.3a DS-UWB proposal, variable length short spreading codes (1~24 chips/symbol and a chip interval is about 1ns) are proposed to achieve data rate 28~1320Mbps [41].

Conventionally, the RAKE receiver is a technique that can effectively mitigate the multipath impairment to obtain the diversity gain of time domain and improve the transmission performance. Numerous types of RAKE receivers have been suggested in the literature for wireless UWB communications [35, 42-44]. The Rake receivers can also be used to collect the multipath signals in the DSSS systems. The classical Rake receiver in the DSSS system resolves the mulipath components by utilizing the auto-correlation characteristic of spreading codes. The technology determines that only the multipath components referred as inter-chip interference (ICI) can be resolved and combined. The multipath components referred as inter-sysmbol interference (ISI) are treated as an additive noise in a

DSSS system [31]. The Rake receiver mentioned in the thesis is referred to this sort of Rake receiver in the DSSS system unless the other statement is given.

Many propagation measurements have been performed for indoor UWB channels and several models have been proposed in the literature [13, 19, 45-53]. To evaluate different developed receiver schemes in the UWB band, IEEE 802.15.3a Channel Model Committee suggests a UWB channel model of indoor transmission environments. The IEEE 802.15.3a channel model is based on a modified Saleh-Valenzuela model where multipath components arrive in clusters and each cluster could contain several components [53]. There are four indoor channel models (CM1~CM4) for different channel characteristics. CM1 denotes the propagation environment with line of sight (LOS) and 0~4m propagation distance. CM2~CM4 denote three non-line of sight (NLOS) propagation environments with different propagation distance or delay spread [53]. The typical root-mean-square (RMS) delay spread measured in the indoor UWB channels range from 17 to 40 ns for antenna separations from 5 to 30m [19]. The suggested RMS delay spread of IEEE 802.15.3a channel model ranges from 5 to 25 ns for CM1~CM4 channels. Thereby the multipath is spread over multiple symbols in the case of ultra-high speed communications of several hundreds Mbps, which causes severe *inter-symbol interference* (ISI) and *inter-chip interference* (ICI) to the DS-UWB scheme [13, 45, 47, 49, 54, 55].

Due to the UWB channel is a severe frequency selective channel in the ultra-high speed communications, it necessitates an equalizer to the multipath signal receptions. Numerous types of equalization technique have been researched in the literature for the wireless UWB communications [22, 23, 25, 35, 44, 56, 57]. Among those equalization schemes, the *maximum likelihood sequence estimation* (MLSE) equalization based on *Viterbi algorithm* (VA) is an attractive approach, since it can provide the optimal detection performance in the ISI cases when the channel noise is a Gaussian variable.

Differential detection, as an alternative to the coherent detection, has attracted lots of interest for its simplicity which comes from the differential detection does not require a carrier phase recovery circuit, and its robustness in the severe fading channels due to the accurate channel estimation is quite difficult in the environments and differential detection does not require a channel estimation. Many differential detection schemes, including the transmit-reference (TR) scheme, are suggested in literature for the wireless UWB communications [12, 20, 21, 23, 25, 34, 58-62]. Classical differential detections where the detections are applied to two adjoining symbols suffer a penalty in performance comparing to the coherent detection. To compensate the performance loss, numerous of improved differential detection schemes based on the observation of a sequence of symbols, referred as *multiple symbols differential detection* (MSDD), are proposed in [63-71].

The objective of the thesis is to overcome the limitation of Classical Rake receivers in the DS-UWB scheme. As an alternative to the Rake receiver, a novel UWB receiver scheme based on the IEEE 802.15.3a DS-UWB PHY standard is proposed in the thesis. Two sequence detection technologies are the foundation of the proposed scheme: *maximum likelihood sequence estimation* (MLSE) and *multiple symbols differential detection* (MSDD). The proposed detection scheme denoted as *maximum likelihood differential sequence estimation* (MLDSE) implements the two sequence detection functions via a modified Viterbi algorithm. The MLDSE implementation can use the same structure of MLSE scheme without increasing any hardware. The proposed MLDSE scheme owns both advantages: the simplicity of differential detection due to the elimination of carrier phase recovery circuits, the optimal performance of differential detection and the optimal performance in the multipath channels.

The objectives of the thesis include:

- To introduce the basic knowledge of several classical signal detection technologies (the matched filer, differential detection and Rake receiver) and two advanced sequence detection technologies (MLSE and MSDD) in the wireless communications. The MLSE and MSDD technologies are the foundation of the proposed MLDSE scheme, the detailed introduction of the MLSE and MSDD technologies can help readers to understand the MLDSE scheme better. In addition, an introduction of Viterbi algorithm gives readers an illustrative knowledge to the iterative implementation of MLSE equalization.
- To introduce the proposed MLDSE algorithm aimed to the IEEE802.15.3a DS-UWB scheme. The performance bound of the proposed MLDSE algorithm is derived by the analogous method in the MSDD performance bound derivation, and the iterative implementation of the MLDSE algorithm by a modified Viterbi algorithm is illustrated by an example. To illustrate one approach of multipath channel estimations in the MLDSE scheme

implementation, an adaptive joint data and channel estimation strategy is presented.

- To demonstrate the superiority of the proposed MLDSE scheme to the Classical Rake receiver in a DS-UWB system. Using the IEEE 802.15.3a UWB channel models, the bit error rates of the proposed MLDSE scheme and classical Rake receiver are compared via the computer simulations in the ideal channel estimation cases. The feasibility of the proposed MLDSE scheme is proved by the computer simulation using the adaptive joint channel and data estimation approach.

1.5 Thesis Contributions

The main contributions of the thesis are:

- The thesis proposes a novel DS-UWB receiver scheme referred as the *maximum likelihood differential sequence estimation* (MLDSE) reception scheme. The MLDSE algorithm can perform the MLSE equalization and MSDD reception functions in one algorithm. The performance bound of the proposed MLDSE scheme is derived theoretically by the analogous approach in the derivation of the MSDD reception performance bound.
- The thesis proposes a modified Viterbi algorithm to iteratively implement the MLDSE algorithm. The decision rule of the Viterbi algorithm is modified to enable the MLDSE algorithm to perform the MLSE equalization and MSDD reception functions at the same time. As consequence, the branch metrics and path metrics of the Viterbi algorithm are modified respectively.
- The thesis proposes a joint data and channel estimation strategy using adaptive algorithms for the MLDSE scheme. The joint estimation strategy greatly simplifies the estimation process of the UWB multipath channel impulse response. The feasibility of the joint estimation strategy is verified via the computer simulations.
- The thesis demonstrates the superiority of the proposed MLDSE receiver to the classical Rake receiver for the IEEE 802.15.3a DS-UWB scheme. Using the IEEE802.15.3a UWB channel models and PHY parameters of the DS-

UWB scheme, the performances of the MLDSE receiver and ideal Rake receiver in the DS-UWB scheme are compared via the computer simulations.

1.6 Thesis Outline

The thesis is organized as follow: Chapter 2 introduces three basic signal detection technologies: matched filter, classical differential detection and Rake receiver. Chapter 3 describes two sequence detection technologies relevant to the thesis: maximum likelihood sequence estimation (MLSE) equalization and multiple symbols differential detection (MSDD) reception. Chapter 4 covers the proposal of the maximum likelihood differential sequence estimation (MLDSE) reception scheme, its performance analysis and its implementation via a modified Viterbi algorithm, details the drawbacks of the Rake receiver in the DS-UWB scheme. Chapter 5 provides the results and comments of the computer simulations for the MLDSE scheme and Rake receiver in the IEEE802.15.3a channel models. Chapter 6 is the conclusion of the thesis.

Chapter 2. Basic Signal Detection Technologies

The transportation of information requires a communication system. Telecommunication system corrupts the information by the introduction of random background noise. The need to filter weak signals out of such noise has called for the development of the technologies of *Signal Detection*. In this chapter, a brief introduction of some basic signal detection technologies is given: matched filter, classical differential detection and Rake receiver. The matched filter reception performs the coherent detection to the received signal and achieves the optimal detection performance when the addition noise is a Gaussian random variable. The classical differential detection performs the non-coherent detection to the received signal on the observation of two symbols duration. The differential detection simplifies the receiver structure at the penalty of some performance loss comparing to the coherent detection. Rake receiver is one of the principle diversity reception techniques in the multipath channels. The classical Rake receiver in the DSSS system will be the interest of the thesis. These basic signal detection technologies are the base of some other advanced signal detection algorithms which will be introduced in the next chapter.

2.1 Matched Filter

The classical matched filter is a simple and optimal method for detecting a deterministic signal in additive white Gaussian noise (AWGN) based on the correlation process. As shown in Figure 2-1. Classical matched filters perform the correlation operation on the received signal $r(t)$, which comprises the transmitted signal $s(t)$ and channel noise $n(t)$. The correlation operation is achieved by multiplying the received signal with a predefined template (similar to the transmitted signal) $s(t)$, and then integrating over one symbol duration T_s.

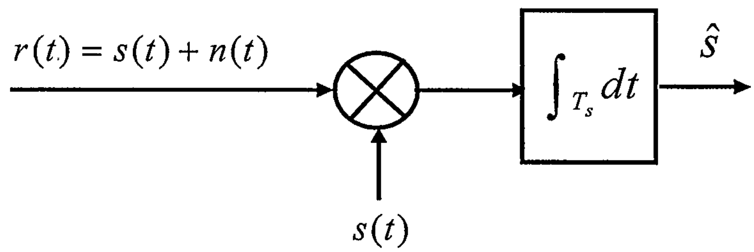

Figure 2-1: Classical matched filter block diagram

The following equations illustrate the matched filters operation mathematically.

$$\begin{aligned} \hat{s} &= \int_0^{T_s} [s(t) + n(t)] \cdot s(t)dt \\ &= \int_0^{T_s} s^2(t)dt + \int_0^{T_s} n(t) \cdot s(t)dt \end{aligned} \tag{2-1}$$

The first item in the detected output $\hat{s}$ is the expected signal with amplitude E_s, where E_s is defined as the energy per symbol, $E_s \triangleq \int_0^{T_s} s^2(t)dt$.

The second item results from the correlation of the signal with noise is a noise item defined as $w(t) \triangleq \int_0^{T} n(t) \cdot s(t)dt$. When $n(t)$ is assumed to AWGN with two-sided power spectrum density of $N_0/2$ and zero mean, $n(t)$ and $s(t)$ are uncorrelated with each other, the noise $w(t)$ is also a Gaussian random variable. We can conclude the mean and variance of $w(t)$ as

$$E[w(t)] = E[\int_0^T n(t)s(t)dt] = \int_0^T E[n(t)] \cdot E[s(t)]dt = 0 \tag{2-2}$$

$$\begin{aligned} E[w^2(t)] &= E[\int_0^T n(\alpha)s(\alpha)d\alpha \cdot \int_0^T n(\beta)s(\beta)d\beta] \\ &= \int_0^T \int_0^T E[n(\alpha)n(\beta)] \cdot s(\alpha) \cdot s(\beta) d\alpha d\beta \\ &= \frac{N_0}{2} \int_0^T s^2(\alpha)d\alpha = \frac{N_0}{2} \cdot E_s \end{aligned} \tag{2-3}$$

The transmitted signal and the random noise generally are assumed to be completely independent. If a filter is matched to the shape of the transmitted signal, the output of the matched filter provides sufficient statistics to detect the deterministic signals in the presence of noise. The output SNR of a matched filter is equal to

$$SNR_{MF} = \frac{E_s^2}{E[w^2(t)]} = \frac{E_s^2}{\frac{N_0}{2} \cdot E_s} = \frac{2E_s}{N_0} \tag{2-4}$$

If the channel noise is an additive Gaussian noise and no inter-symbol interference (ISI) exists, the classical matched filter is a coherent receiver which has been proved to be optimal in terms of the maximization of the received signal's signal-to-noise ratio (SNR) [72] . In the channel noise isn't a Gaussian noise or the ISI exists, the matched filter is no longer the optimal receiver.

2.2 Differential Detection of *M*-DPSK

In general, the implementation and operation of a coherent receiver is a complex process involving many ancillary functions associated with the carrier synchronization process, such as the signal acquisition, tracking, lock detection, false lock prevention functions, etc. In many applications, simplicity and robustness of implementation take precedence over achieving the best possible system performance. An attractive alternative to coherent detection, *differential detection*, can obviate the complex carrier synchronization implementation. Aside from implementation considerations, it is also possible that the transmission channels may be so degraded that accurate acquiring and tracking a coherent demodulation reference signal are difficult. Here again, differential detection is a possible solution that still provides better performance than the noncoherent detection scheme such as an energy detector [72]. In principle, differential detection can be used with many different modulation schemes, the most widespread applications pertain to constant envelop modulations. The interest will focus in this chapter on the most common modulation scheme known as multiple phase-shift-keying (*M*-PSK).

Traditionally differential detection of *M*-PSK has been accomplished by comparing the received phase in a given symbol interval (of duration T_s seconds) with the phase in the previous symbol interval and making a multilevel decision on the difference between these two phases. The assumption in this process is that the phase introduced by the channel is constant over two symbols intervals. Since the decision is equivalently being made on the difference between two adjacent transmitted phases, a suitable coding must be applied at the transmitter to allow this phase difference between two adjacent symbols to represent a transmitted data. The encoding is referred to as *differential encoding*. The differential encoding was included in the system solely as a means of resolving the phase ambiguity associated with demodulation reference signal.

2.2.1 *The Classical Coherent Detection for M*-DPSK

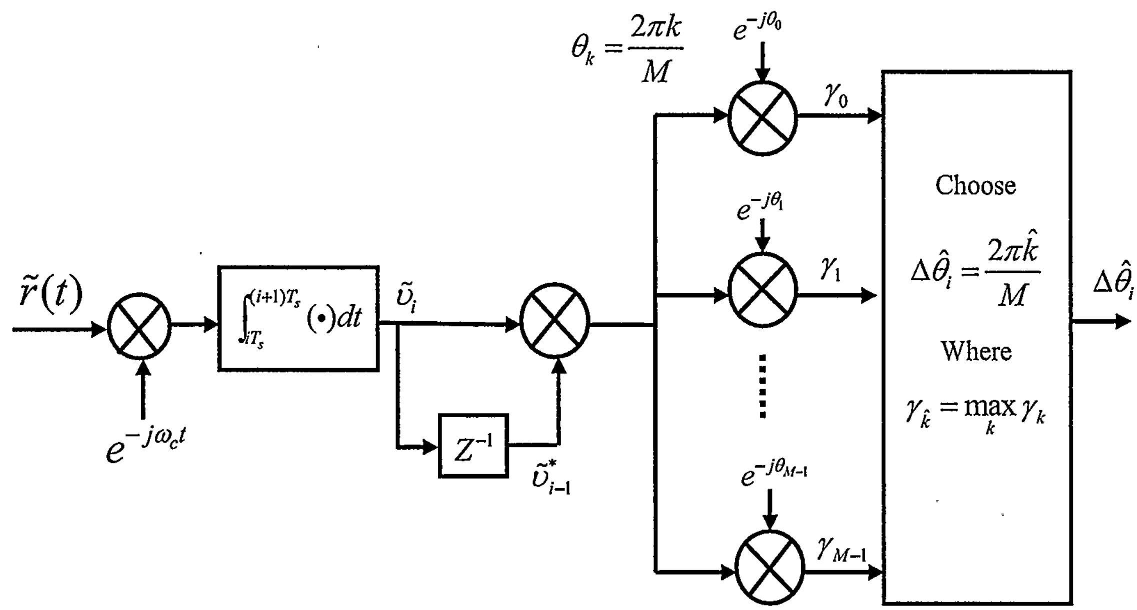

Figure 2-2: Classical Coherent Detection of *M*-DPSK signal

Figure 2-2 shows one classical implementation of the optimum receiver of *M*-DPSK signal. The derivation of the symbol error probability performance of this optimum *M*-DPSK receiver is quite complex. The details of the derivation for the differentially coherent detection are presented in [72]. The general symbol error probability expression of the optimum receiver for *M*-DPSK is given as [72]:

$$P_s(E) = \frac{\sin\frac{\pi}{M}}{2\pi} \int_{-\pi/2}^{\pi/2} \frac{\exp\left\{-\frac{E_s}{N_0}\left[1-\cos\frac{\pi}{M}\cos t\right]\right\}}{1-\cos\frac{\pi}{M}\cos t} dt \tag{2-5}$$

The special case of binary ($M = 2$) DPSK, often simply called DPSK, is a case where the error probability performance can be found in closed form (as opposed to an integration). Substituting $M = 2$ into (2-5) gives the simple result [72]:

$$P_s(E) = P_b(E) = \frac{1}{2}\exp\left(-\frac{E_b}{N_0}\right) \tag{2-6}$$

2.2.2 *The Classical Differential Detection of DPSK Signal*

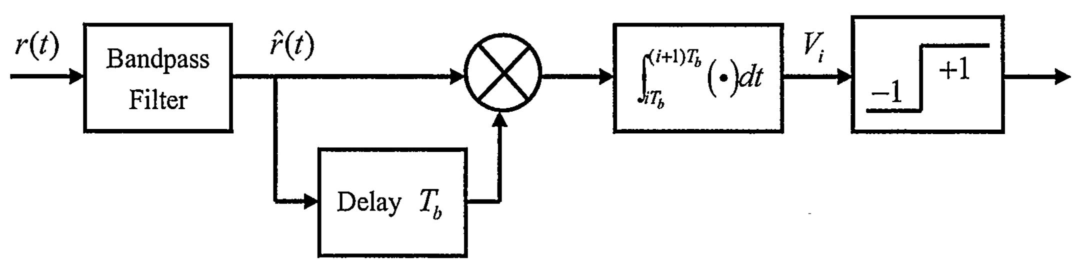

Figure 2-3: Classical Differential Detection of DPSK Signal

Figure 2-3 illustrates the classical differential detection of DPSK signal. The bandwidth W ($B = W/2$ is the equivalent lowpass bandwidth) of the bandpass filter is assumed to be sufficiently wide so that signal distortion and intersymbol interference (ISI) are not an issue. The received signal following the bandpass filter in the i^{th} transmission interval has the form:

$$\begin{aligned}\hat{r}(t) &= \sqrt{\frac{2E_b}{T_b}}\cos(\omega_c t+\theta_i+\theta)+\hat{n}(t) \\ &= \sqrt{\frac{2E_b}{T_b}}\cos(\omega_c t+\theta_i+\theta)+\frac{\sqrt{2}}{2}\left[n_c(t)\cos\left(\omega_c t+\theta\right)-n_s(t)\sin\left(\omega_c t+\theta\right)\right]\end{aligned} \tag{2-7}$$

where E_b denotes the bit energy. T_b denotes the bit duration. ω_c denotes the radian carrier frequence. θ_i denotes the transmitted phase in the i^{th} interval. The channel introduces the unknown phase θ, which in the absence of any information is assume to be uniformly distributed in the interval $(-\pi,\pi)$. $\hat{n}(t)$ is complex additive white Gaussian process with two-sided power spectrum density N_0.

The output of the suboptimal differential detection V_i is (ignoring second harmonic terms) given as:

$$\begin{aligned}V_i &= \int_{iT_b}^{(i+1)T_b}\hat{r}(t)\hat{r}(t-T_b)dt = E_b\cos\Delta\theta_i \\ &+\sqrt{\frac{E_b}{T_b}}\int_{iT_b}^{(i+1)T_b}\left[n_c(t)\cos\theta_{i-1}+n_c(t-T_b)\cos\theta_i\right]dt \\ &+\sqrt{\frac{E_b}{T_b}}\int_{iT_b}^{(i+1)T_b}\left[n_s(t)\sin\theta_{i-1}+n_s(t-T_b)\sin\theta_i\right]dt \\ &+\int_{iT_b}^{(i+1)T_b}\left[n_c(t)n_c(t-T_b)+n_s(t)n_s(t-T_b)\right]dt\end{aligned} \tag{2-8}$$

Assuming an ideal rectangular filter for the input bandpass filter, then $n_c(t)$ and $n_s(t)$ are statistically independent with autocorrelation function:

$$R_n(\tau)=E\{n_c(t)n_c(t+\tau)\}=E\{n_s(t)n_s(t+\tau)\}=N_0B\left(\frac{\sin 2\pi B\tau}{2\pi B\tau}\right) \tag{2-9}$$

In view of the assumption that B is an integer multiple of $1/2T_b$, then $R_n(T_b)=0$ and hence the first two moments of V_i are given by:

$$\overline{V_i}=\begin{cases} E_b & \Delta\theta_i=0 \\ -E_b & \Delta\theta_i=\pi \end{cases} \tag{2-10}$$

$$\begin{aligned}\sigma_{V_i}^2 &= 2\frac{E_b}{T_b}BN_0\int_{iT_b}^{(i+1)T_b}\int_{iT_b}^{(i+1)T_b}\left\{\left(\frac{\sin 2\pi B(t-t')}{2\pi B(t-t')}\right)+\left(\frac{\sin 2\pi B(t-t'-T_b)}{2\pi B(t-t'-T_b)}\right)\right\}dtdt' \\ &+2(BN_0)^2\int_{iT_b}^{(i+1)T_b}\int_{iT_b}^{(i+1)T_b}\left\{\left(\frac{\sin 2\pi B(t-t')}{2\pi B(t-t')}\right)^2+\left(\frac{\sin 2\pi B(t-t'-T_b)}{2\pi B(t-t'-T_b)}\right)\left(\frac{\sin 2\pi B(t-t'+T_b)}{2\pi B(t-t'+T_b)}\right)\right\}dtdt'\end{aligned} \tag{2-11}$$

In the condition of $B \gg 1/T_b$, the filtered noise approximately approach the Gaussian distribution, then the above integrals can be approximately evaluated, resulting in

$$\sigma_{V_i}^2 \approx E_bN_0+BT_bN_0^2 \tag{2-12}$$

The noise terms in (2-8) are zero mean. For sufficiently high SNR, the Gaussian first and second noise terms will dominate. Thus, the bit error probability is evaluated in the same manner as for an antipodal signal set on an AWGN channel.

$$P_b(E)=\frac{1}{2}erfc\sqrt{\frac{\left(\overline{V_i}\right)^2}{2\sigma_{V_i}^2}}\approx\frac{1}{2}erfc\sqrt{\frac{E_b}{2N_0\left(1+\frac{BT_bN_0}{E_b}\right)}} \tag{2-13}$$

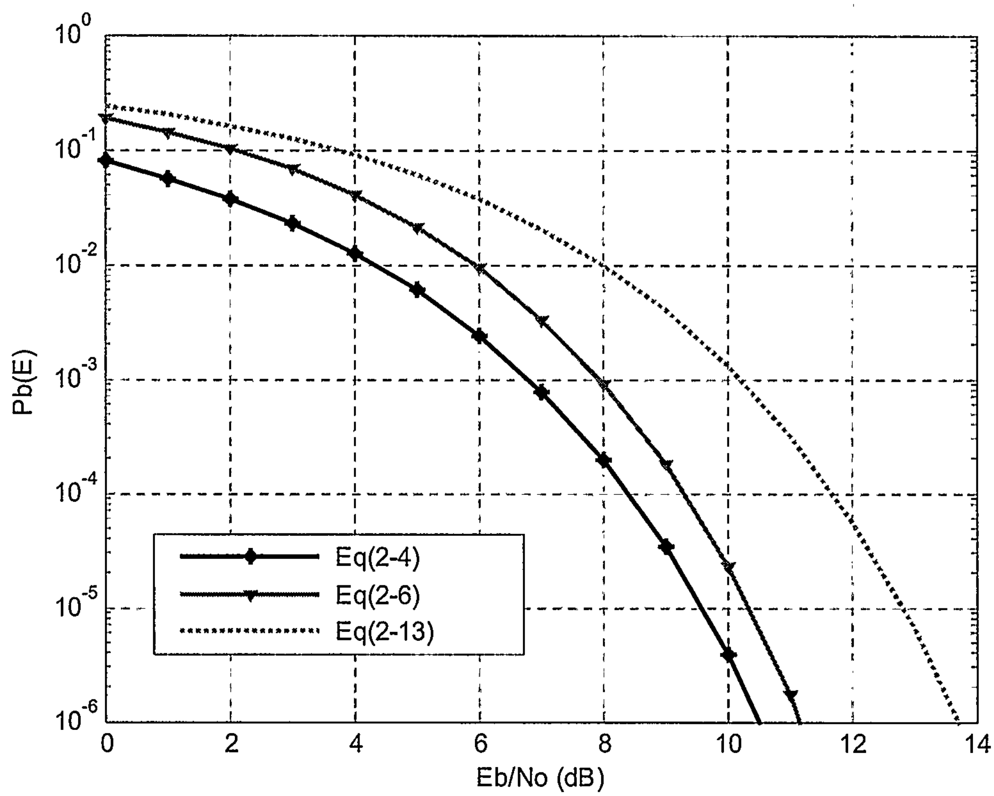

Figure 2-4: Bit error rates of coherent and differential detection of DPSK

Figure 2-4 illustrates the bit error rates of the coherent detection of DPSK signal (using Eq(2-6)) and the differential detection of the DPSK signal (using Eq(2-13)). The figure also gives the bit error rate of the coherent detection of BPSK signal as a benchmark (using Eq(2-4)). The performance loss between the coherent detections of BPSK and DPSK signal is caused by the error propagation effects of the differential coding. In the DPSK case, any single bit error caused by the channel noise will result in two bit errors in the differential detection. The reason for the performance loss in the figure between the coherent detection and the differential detection of DPSK signal is that a noisy reference signal is used in the differential detection.

2.3 Rake Receiver of the DSSS signals

The mobile radio channel is a dispersive channel. Signal cancellation occurs when the direct and reflected waves arrive with the inverse phases. *Direct Sequence Spread Spectrum* (DSSS) technology has particular property that makes it less vulnerable to the multipath interference. In the DSSS system Rake receiver is introduced to collect and combine the multipath signals over the dispersive channels. Rake receiver can be designed to have many branches. Each branch processes a multipath component respectively. Each branch in a RAKE receiver is called a RAKE receiver finger.

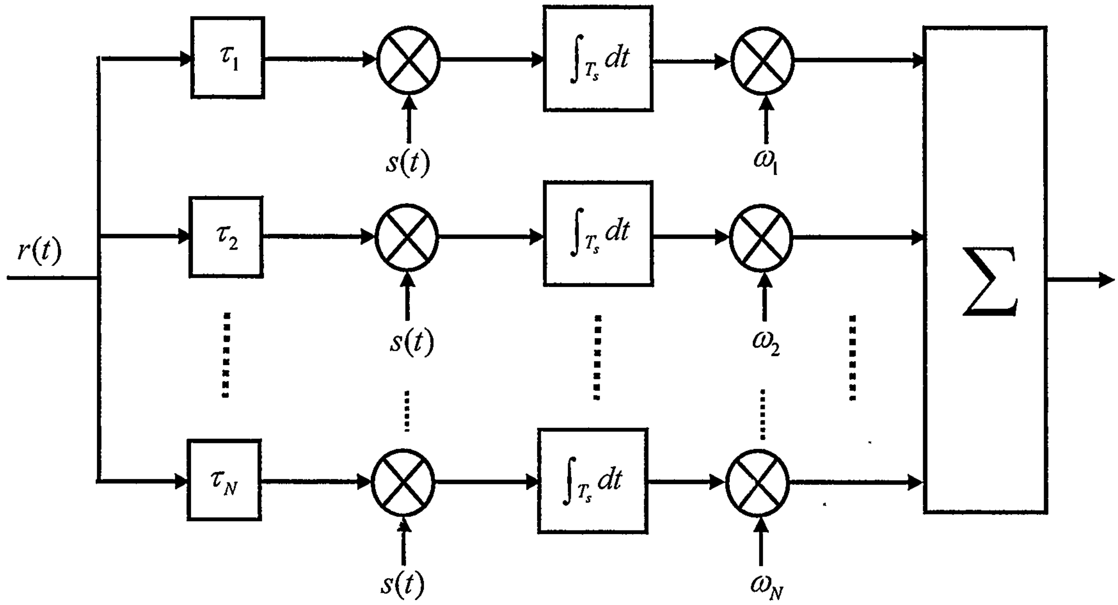

Figure 2-5: The implementation of the Rake receiver

Figure 2-5 illustrates one implementation of the Rake receiver. The Rake receiver consists of multiple correlators, in which the time-shifted version of the received signal is multiplied by a locally generated code sequence. The intention is to separate signals such that each finger only sees signals coming in over a single (resolvable) path. The spreading code is chosen to have a very small autocorrelation value for any nonzero time offset. This avoids crosstalk between fingers. Utilizing the special property of the spreading code, when the multipath delays fall in the duration of one symbol, the Rake receiver performs the diversity reception in the time domain.

The outputs of all fingers are weighted and combined to achieve improved communications reliability and performance. The optimal linear combining technique is the *maximal ratio combining* (MRC) which yields the maximum output signal-to-noise ratio (SNR) [14]. For the rake receiver under consideration, the combiner appropriately weights the fingers' output according to their SNR prior to summing. The performance and optimality of MRC consequently depend upon the receiver's knowledge of the channel. Another suboptimal linear combining technique is referred as the *equal gain combining* (EGC) which weights the fingers' output with same gain without the requirement of the channel knowledge. Therefore the *maximal ratio combining* (MRC) can be used for the best performance or the *equal gain combining* (EGC) can be used for the simplicity.

For DS-UWB systems, rake reception is useful for capturing most of the multipath energy. The typical multipath number which accounts 85% multipath energy ranges from 20 to 120 for the channel CM1 to CM4 in IEEE 802.15.3a channel model [53]. As a result, the RAKE receiver needs a large number of fingers and the computational complexity of the RAKE receiver becomes quite high. Therefore, two kinds of suboptimal rake receptions have been proposed [31, 35, 42]. One is selective rake (S-rake) reception, which combines the largest N_s paths using maximum ratio combining (MRC) or equal gain combining (EGC). The other is called partial rake (P-rake), which combines the initial N_p paths using MRC or EGC. An S-rake can provide sufficient captured energy to compensate channel impairment, but its implementation requires many delay and tracking devices. On the other hand, a P-rake can capture less energy, but it can be implemented with low complexity.

2.4 Summary

In this chapter three basic signal detection technologies are introduced. Matched filter is the optimal detection in the Gaussian channel without ISI and makes decisions base on the observation over one symbol duration. The classical differential detection simplifies the receiver structure and makes decision base on the observations of the current symbol and the previous symbol. The Rake receiver in the DSSS system performs the diversity collection when the relative path delays of dispersive channel are less than one symbol duration (the ICI components). In the real mobile communication there are many applications in the severe dispersive radio channel where the multipath delays span over multiple symbols interval. These channels are referred to the inter-symbol interference (ISI) channels. These signal detection technologies based on one symbol observation are not optimal signal detection any more in the ISI environments. Some advanced signal detection technologies based on the multiple symbols observation are developed, namely *sequence detection*. In the next chapter, two of the sequence detection technologies are introduced.

Chapter 3. Literature Review

Some UWB applications need to work in the multipath channels with abundant of inter-symbol interference (ISI). The IEEE802.15.3a DS-UWB proposal which is the interest of the thesis is one of the cases. The parameters of IEEE802.15.3a channel models are given in the Chapter 4. The signal detection technologies described in the Chapter 2 are based on one symbol observation and are not optimal signal detection any more in the ISI environments. Some advanced signal detection technologies based on the multiple symbols observation are developed, namely *sequence detection*. In this chapter a brief review of two kinds of sequence detection technologies which are relevant to our proposed receiver scheme, including Forney's [73] *maximum-likelihood sequence estimation of digital sequences in the presence of inter-symbol interference* and Divsalar's [71] *Multiple-symbol differential detection of MPSK* will be given. The chapter is a detailed introduction to the works of Forney and Divsalar. Proakis' *Digital Communication* [14] and Simon's *Digital Communication Techniques* [72] are great reference books to the findings of Forney and Divsalar. The chapter is an extraction of those references. All derivations and results in the chapter belong to their works unless other statement is given. Readers should be careful that the variables are dealing in real quantities in Forney's works and the variables are dealing in complex quantities in Divsalar's works.

3.1 Maximum Likelihood Sequence Estimation of digital sequences in the presence of ISI

In reality, the communication technique is implemented over the band-limited channel. If the channel is not ideal over its bandwidth occupancy (W Hz), the signaling at a rate W symbols per second or higher results in a form of distortion known as *inter-symbol interference* (ISI). The current symbol will be interfered by the previous symbols, which degrades the SNR of signals. A class of technique referred as *equalization* introduces to mitigate the ISI distortion caused by the multipath channels, such as *zero-forcing linear equalizer* and *decision-feedback equalizer* [74-76], etc. Among those equalizers, a sort of equalization based on the notion of *maximum-likelihood sequence estimation* (MLSE) is optimum in the presence of both ISI and additive Gaussian noise in the sense of SNR loss. In MLSE the detection is based on a sequence of information symbols with the goal of minimizing the symbol error probability. The reason for estimating an information sequence rather than a single symbol at receiver is that ISI has the effect of introducing memory into the modulation in the sense that the detection of a given symbol depends on the knowledge of past data symbols.

3.1.1 *The Minimum-Distance Criterion*

Many of the communication systems can be modeled as a sequence of information corrupted by noise. Consider following AWGN channel:

$$\upsilon = s + \eta \tag{3-1}$$

where the signal vector $s = (s_1, s_2, \cdots, s_N)$ is an information sequence of N symbols observation interval, and where the noise vector $\eta = (\eta_1, \eta_2, \cdots, \eta_N)$ is Gaussian therefore the components are uncorrelated (and hence is independent sample to sample) with uniform variance σ^2. When the noise is independent, the conditional probability distribution of the observation given the signal vector s can be expressed as:

$$f_{\upsilon|s}(\upsilon|s) = \prod_{k=1}^{N} f_{\upsilon_k|s_k}(\upsilon_k|s_k) \tag{3-2}$$

The *maximum likelihood* (ML) detector chooses the estimated signal vector $\tilde{s}$ from among all the possibilities in order to maximize the conditional probability $f_{\upsilon|s}(\upsilon|\tilde{s})$. Since the received signal υ is a Gaussian vector with mean equal to s, the conditional probability density function follows:

$$f_{\upsilon|s}(\upsilon|s) = f_\eta(\upsilon - s) = \frac{1}{(2\pi\sigma^2)^{\frac{N}{2}}} \exp\left(-\frac{\|\upsilon - \hat{s}\|^2}{2\sigma^2}\right) \tag{3-3}$$

Since the exponential is a monotonic function of its exponent, maximizing $f_\eta(\upsilon - \hat{s})$ is equivalent to minimizing $\|\upsilon - \hat{s}\|^2$. In other words, the ML detector reduces to the *minimum-distance* detector for the special case of white Gaussian noise channel.

We extend the minimum-distance concept to the ISI case of the *M*-ary digital communication. The signal item of (3-1) is defined as:

$$s_i = \sum_{j=0}^{L} f_j I_{i-j} \tag{3-4}$$

where the sequence of transmitted symbols $\{I_i\}$ is drawn from an alphabet of size *M*, $\{f_j\}$ denotes the equivalent discrete sampled version of channel impulse response, the channel response $\{f_j\}$ spans $L+1$ symbols. A straightforward expression of minimum-distance N symbols sequence detector of the ISI case can be shown as:

$$\{\hat{I}_k\} = \arg \min_{\{I_k\} \in M^N} \sum_{k=1}^{N} \left| \upsilon_k - \sum_{j=0}^{L} f_j I_{k-j} \right|^2 \tag{3-5}$$

3.1.2 *The Viterbi Algorithm*

The minimum-distance sequence detection requires that (3-5) be calculated M^N times, once for each possible sequence of N symbols from an *M*-ary alphabet. The computation is therefore exponential in the sequence length. In this section we describe a dynamic programming algorithm known as the *Viterbi algorithm* (VA) which was originally proposed by Viterbi in 1967 [77]. The VA achieves a computational load that is linear in the sequence length, so the algorithm can be implemented with a fixed computational rate. The Viterbi algorithm is used in many applications beyond sequence detection such as the decoding of convolution codes and the equalizing of the ISI channel [18].

It is easily demonstrated that the metrics in (3-5) can be computed recursively. Since the ISI channel model has finite duration response that spans $L+1$ symbols, the signal item s_k is independent to s_{k-L-i} for $i=1,2,\cdots,k-L-1$. The conditional joint probability density function given the transmitted sequence $\boldsymbol{I}=(I_1,I_2,\cdots,I_N)$ can be expressed as a product of marginal densities:

$$\begin{aligned} f(\boldsymbol{\upsilon}|\boldsymbol{I}) &= f(\upsilon_N,\upsilon_{N-1},\cdots,\upsilon_1|I_N,I_{N-1},\cdots,I_1) \\ &= \prod_{k=1}^{N} f(\upsilon_k|I_k,I_{k-1},\cdots,I_{k-L}) \end{aligned} \tag{3-6}$$

where, by definition, $I_k=0$ for $k\le 0$. Then the metrics in (3-5) can be computed recursively by denoting the decision variable

$$Z_k = \left|\upsilon_k - \sum_{j=0}^{L} f_j I_{k-j}\right|^2 + Z_{k-1} \tag{3-7}$$

In each stage of the Viterbi algorithm, only M^L surviving sequences from the previous stage are needed to calculate the M probabilities of this stage, then total M^{L+1} probabilities are computed each stage. The M^{L+1} sequences are subdivided into M^L groups of M sequences. From each group of M sequences, the one having the minimum distance is selected (then the sequence is a survivor) and the remaining $M-1$ sequences are discarded. Hence, a total of NM^{L+1} probabilities are computed in the detection of N symbols sequence.

The best way to describe Viterbi algorithm is through an example using a method called *trellis diagram*. The trellis diagram is essentially a plot of all possible state progressions versus time.

Example of Viterbi algorithm: A simple example for illustrative purposes is the binary OOK alphabet $\lambda = \{0,1\}$ and the following ISI model ($f_0 = 1,\ \ f_1 = 0.5$) is shown in Figure 3-1(a). The channel memory is therefore $L = 1$, and the state is $\Psi_k = I_{k-1}$, the previous input. Since the alphabet is binary, there are only two possible states. The state transition diagram for this example is shown in Figure 3-1(b), where the arcs are labeled with the input/output pair (I_k, s_k).

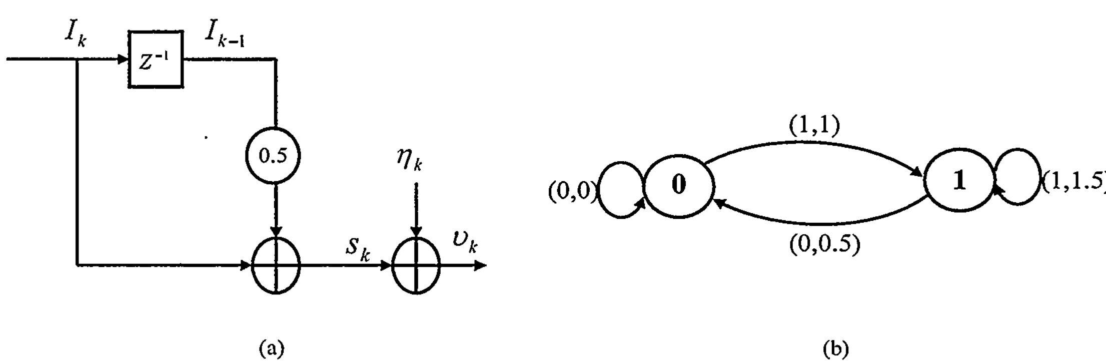

Figure 3-1: (a) The ISI signal generator. (b) The state transition diagram

The trellis diagram for this example is shown in Figure 3-2(a). The starting and ending conditions are $\Psi_0 = 0$ and $\Psi_{N+1} = 0$. Each small circle is a node of the trellis and corresponds to a particular state at a particular time. Each arc in the diagram is called a *branch*, and corresponds to a particular state transition at a particular time. Thus, the single node at the left indicates that state transition begins in state $\Psi_0 = 0$ at time $k = 0$. The next state can be either state $\Psi_1 = 0$ or state $\Psi_1 = 1$, depending on the value of I_1. The transitions to both states are shown in the diagram. From time $k = 1$, the state transition of this example may branch from any node (state) to any other node (state), until it reaches the terminal node of the trellis in state $\Psi_{N+1} = 0$. Each branch in the trellis corresponds to one state transition that is triggered by a particular input I_k and produces the output s_k, and thus there is a one-to-one correspondence between a branch (the state transition) and the input and output pairs. One stage of the trellis is shown in Figure 3-2(b) with the input and output pairs (I_k, s_k) labeled for each transition.

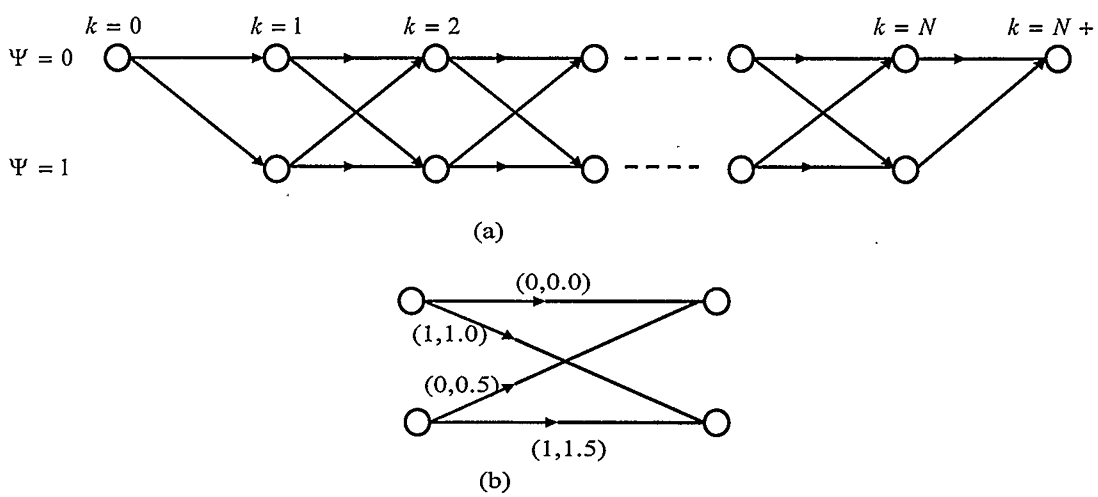

Figure 3-2: The trellis diagrams of the possible state transition

Figure 3-2 illustrates: (a) A two-state trellis illustrates the possible state transition, assuming the initial and final states are zero. (b) One stage of the trellis, where each branch is labeled with the input and output pair (I_k, s_k) corresponding to that state transition.

The trellis shown in Figure 3-3 is marked with branch metrics corresponding to the transmission of $N=3$ symbols and observing $\upsilon=\{0.2,0.6,0.9,0.1\}$ for this example. The branch metrics $|\upsilon_k - s_k|^2$ are labeled in corresponding branch. As an example, for $s_1=1$, the branch metric is achieved by $|\upsilon_1 - s_1|^2 = |0.2-1|^2 = 0.64$.

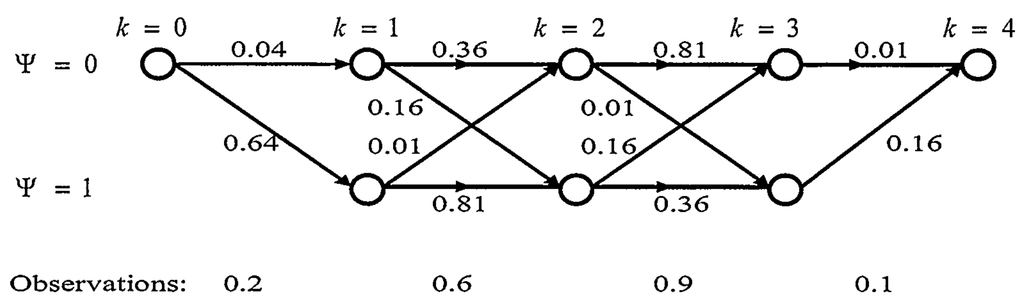

Figure 3-3: A two-state trellis with the transitions branch metrics marked

An iterative procedure for making this decision is illustrated in Figure 3-4. The survivor paths at each node and partial path metric of each surviving path are shown. Since the channel memory $L=1$, each stage of Viterbi algorithm will compute 4 path metrics to find the path with the minimum path metric to each node, and discard two branches with larger branch metrics.

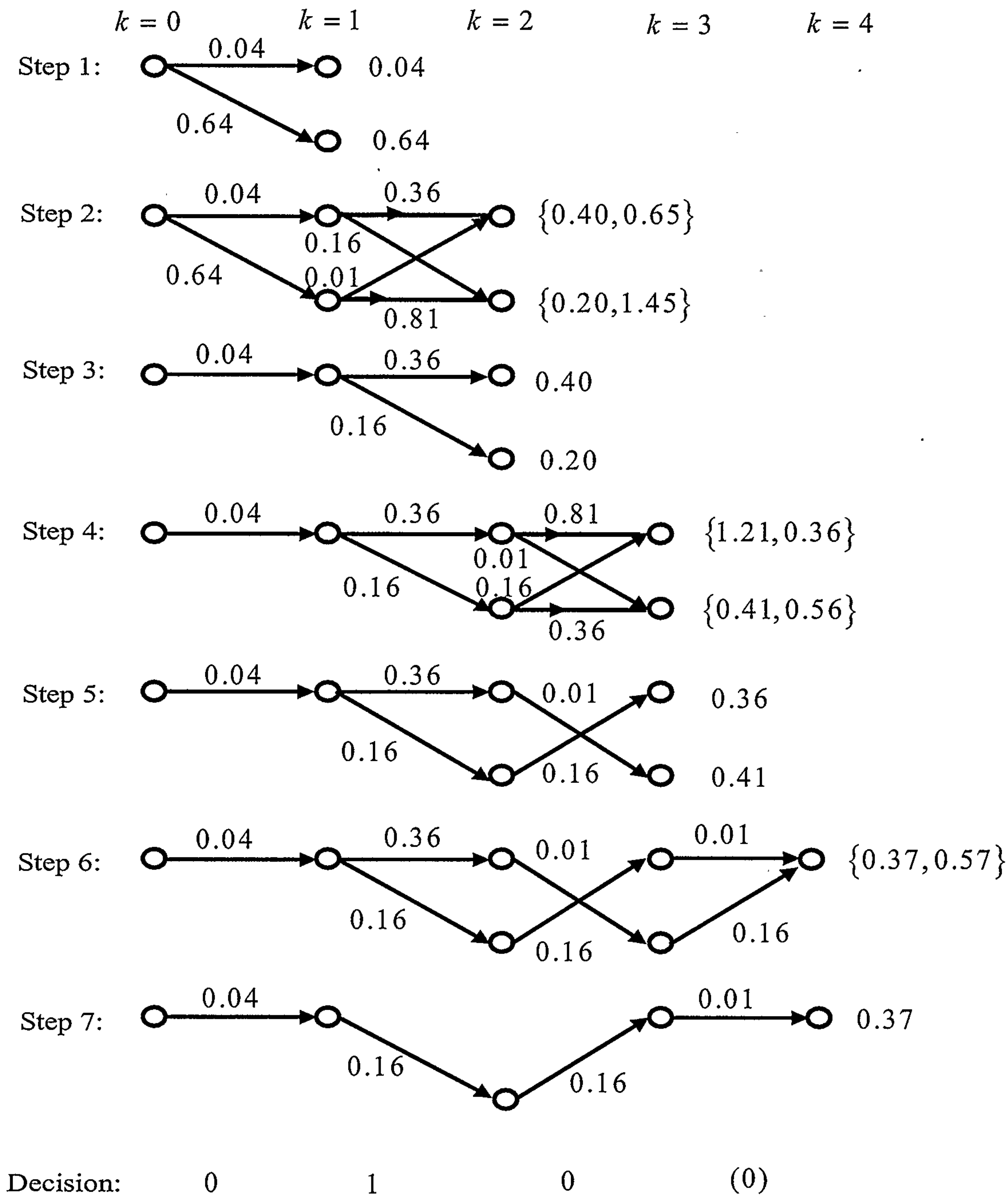

Figure 3-4: An illustrated iterative procedure of Viterbi algorithm.

In this example, the branch metrics are labeled with each branch, the partial path metrics which are the sum of the branch metrics in the path are labeled in the end of the path. The Viterbi algorithm iteratively finds the path with the minimum path metric to the each node and discards the others.

Step 1: Each node at $k=1$ has one incoming branch from a known node ($\Psi_0 = 0$), the two paths are survivor.

Step 2: Each node at $k=2$ have two incoming branches from previous nodes, computes the partial path metric of each path to that node. For the node $\Psi_2 = 0$, two partial path metrics are 0.40 and 0.65 respectively. For the node $\Psi_2 = 1$, two partial path metrics are 0.20 and 1.45 respectively.

Step 3: Select the path with the smaller path metric as a survivor. For the node $\Psi_2 = 0$, the path with path metrics of 0.40 is the survivor. For the node $\Psi_2 = 1$, the path with path metrics of 0.20 is the survivor. The surviving paths to each node are shown. The other two paths with relatively larger path metric are discarded.

Step 4: Each node at $k=3$ have two incoming branches from previous nodes, computes the partial path metric of each path to that node. For the node $\Psi_3 = 0$, two partial path metrics are 1.21 and 0.36 respectively. For the node $\Psi_3 = 1$, two partial path metrics are 0.41 and 0.56 respectively.

Step 5: Select the path with the smaller path metric as a survivor. For the node $\Psi_3 = 0$, the path with path metrics of 0.36 is the survivor. For the node $\Psi_3 = 1$, the path with path metrics of 0.41 is the survivor. The surviving paths to each node are shown. The other two paths with relatively larger path metric are discarded.

Step 6: All the paths will enter a known ending node ($\Psi_4 = 0$), compute all path metrics from the previous surviving paths. To the $\Psi_4 = 0$, the path metrics are 0.37 and 0.57 respectively.

Step 7: The path with path metric of 0.37 is used to make the final decision and the path with path metric of 0.57 will be discarded.

Result of the example: A simple instantaneous detector would decide that the transmitted bits were $\{0,1,1\}$, but the minimum-distance sequence detector takes into account knowledge of the ISI and selects $\{0,1,0\}$.

The computational complexity of Viterbi algorithm is same at each step except for the end effects at the originating and terminating nodes and the total computational complexity is proportional to the sequence length N. One practical problem still exists. The algorithm does not determine the optimal path until the terminal node of the trellis. It does not reach a conclusion on the entire sequence until the end of the sequence. In addition, although the computation at each step is the same, the memory required to store the survivor paths grows linearly with the sequence length. In digital communications systems, sequence may be very long, and we cannot afford the resulting long delay in making decisions and the very large memory that would be required. In practice the very long delay is avoided by truncating the surviving sequences to d most recent symbols where $d \gg L$ (d is called *trace back length*). This results the detector is a suboptimum detector. It is proved that the loss in performance resulting from this suboptimum decision procedure is negligible if $d \geq 5L$.

3.1.3 *Performance of Maximum-Likelihood Sequence Estimation*

Following the procedure developed by *Forney* [73] and *Proakis* [14], the derivation of the probability of error for the MLSE of the received information sequence will be detailed in the section when the information is transmitted via *M*-PAM and additive Gaussian noise. The received signals of the *M*-PAM digital communication subject to inter-symbol interference and white Gaussian noise can be denoted as

$$\upsilon_i = \sum_{j=0}^{L} f_j I_{i-j} + \eta_i \tag{3-8}$$

where the data sequence $\{I_i\}$ takes the values $\pm d, \pm 3d, \cdots, \pm(M-1)d$, and $2d$ is the distance between successive levels, the channel response $\{f_j\}$ spans $L+1$ symbols, and the $\{\eta_i\}$ is a real-valued white Gaussian noise sequence with one-sided power spectrum density N_0.

Let the transmitted symbols and the estimated symbols from the Viterbi algorithm be denoted as $\{I_i\}$ and $\{\tilde{I}_i\}$, respectively. The trellis has M^L states, defined at time k as

$$\Psi_k = \left(I_{k-1}, I_{k-2}, \cdots, I_{k-L}\right) \tag{3-9}$$

the corresponding estimated state at time k be denoted as

$$\tilde{\Psi}_k = \left(\tilde{I}_{k-1}, \tilde{I}_{k-2}, \cdots, \tilde{I}_{k-L}\right) \tag{3-10}$$

Now suppose that the estimated path through the trellis diverges from the correct path at time k and remerges with the correct path at time $k+l$. Thus $\tilde{S}_k = S_k$ and $\tilde{S}_{k+l} = S_{k+l}$, but $\tilde{S}_m \neq S_m$ for $k < m < k+l$. As in a convolution code, we call this an *error event*. Since the channel spans $L+1$ symbols, it follows that $l \geq L+1$.

For such an error event, we have $\tilde{I}_k \neq I_k$ and $\tilde{I}_{k+l-L-1} \neq I_{k+l-L-1}$, but $\tilde{I}_m = I_m$ for $k-L \leq m \leq k-1$ and $k+l-L \leq m \leq k+l-1$. It is convenient to define an error vector $\boldsymbol{\varepsilon}$ corresponding to this error event as

$$\boldsymbol{\varepsilon} = \left(\varepsilon_k, \varepsilon_{k+1}, \cdots, \varepsilon_{k+l-L-1}\right) \tag{3-11}$$

where the components of $\boldsymbol{\varepsilon}$ are defined as

$$\varepsilon_j = \frac{1}{2d}(I_j - \tilde{I}_j) \quad j = k, k+1, \cdots, k+l-L-1 \tag{3-12}$$

The normalization with a factor of $2d$ results in element ε_j takes on the value $\pm 1, \pm 2, \pm 3, \cdots, \pm(M-1)$. Moreover, the error vector is characterized by the properties that $\varepsilon_k \neq 0$, $\varepsilon_{k+l-L-1} \neq 0$, and there is no sequence of L consecutive elements that are zero.

For the error event $\boldsymbol{\varepsilon}$ to occur, the following three subevents ξ_1, ξ_2 and ξ_3 must occur:

ξ_1: At time k, $\tilde{S}_k = S_k$.

ξ_2: The information symbols $I_k, I_{k+1}, \cdots, I_{k+l-L-1}$ when added to the scaled error sequence $2d\left(\varepsilon_k, \varepsilon_{k+1}, \cdots, \varepsilon_{k+l-L-1}\right)$ must result in an allowable sequence, i.e., the sequence $\tilde{I}_k, \tilde{I}_{k+1}, \cdots, \tilde{I}_{k+l-L-1}$ must have values selected from $\pm d, \pm 3d, \cdots, \pm(M-1)d$.

ξ_3: For $k \le m \le k+l$, the sum of the branch metrics of the correct path exceeds the sum of the branch metrics of the estimated path.

The probability of occurrence of ξ_3 is

$$P(\xi_3) = \Pr\left[\sum_{i=k}^{k+l-1}\left(\upsilon_i - \sum_{j=0}^{L} f_j \tilde{I}_{i-j}\right)^2 < \sum_{i=k}^{k+l-1}\left(\upsilon_i - \sum_{j=0}^{L} f_j I_{i-j}\right)^2\right] \tag{3-13}$$

Substitution of (3-8) into (3-13) yields

$$\begin{aligned} P(\xi_3) &= \Pr\left[\sum_{i=k}^{k+l-1}\left(\eta_i + 2d\sum_{j=0}^{L} f_j \varepsilon_{i-j}\right)^2 < \sum_{i=k}^{k+l-1}\eta_i^2\right] \\ &= \Pr\left[4d\sum_{i=k}^{k+l-1}\eta_i\left(\sum_{j=0}^{L} f_j \varepsilon_{i-j}\right) < -4d^2\sum_{i=k}^{k+l-1}\left(\sum_{j=0}^{L} f_j \varepsilon_{i-j}\right)^2\right] \end{aligned} \tag{3-14}$$

where $\varepsilon_j = 0$ for $j < k$ and $j > k+l-L-1$. If we define

$$\alpha_i = \sum_{j=0}^{L} f_j \varepsilon_{i-j} \tag{3-15}$$

Then (3-14) may be expressed

$$P(\xi_3) = \Pr\left[\sum_{i=k}^{k+l-1} \alpha_i \eta_i < -d\sum_{i=k}^{k+l-1} \alpha_i^2\right] \tag{3-16}$$

where the factor of $4d$ common to both terms has been dropped. Now (3-16) is just the probability that a linear combination of statistically independent Gaussian random variables is less than some negative number. Thus

$$P(\xi_3) = \frac{1}{2} erfc\left(\sqrt{\frac{d^2}{N_0}\sum_{i=k}^{k+l-1} \alpha_i^2}\right) \tag{3-17}$$

For convenience we define

$$\delta^2(\boldsymbol{\varepsilon}) = \sum_{i=k}^{k+l-1} \alpha_i^2 = \sum_{i=k}^{k+l-1}\left(\sum_{j=0}^{L} f_j \varepsilon_{i-j}\right)^2 \tag{3-18}$$

Then (3-17) can be expressed as

$$P(\xi_3) = \frac{1}{2} erfc\left(\sqrt{\frac{d^2}{N_0}\delta^2(\boldsymbol{\varepsilon})}\right) \tag{3-19}$$

The probability of the subevent ξ_2 depends only on the statistical properties of the input sequence. We assume that the information symbols are equally probable and that the symbols in the transmitted sequence are statistically independent. Then, for an error of the form $|\varepsilon_i| = \Lambda_i, \Lambda_i = 1, 2, \cdots, (M-1)$, there are $M - \Lambda_i$ possible values of I_i such that

$$I_i = \tilde{I}_i + 2d\varepsilon_i \tag{3-20}$$

Hence

$$P(\xi_2) = \prod_{i=0}^{l-L-1} \frac{M - \Lambda_i}{M} \tag{3-21}$$

The probability of the subevent ξ_1 is much more difficult to compute exactly because of its dependence on the subevent ξ_3. That is, we must compute $P(\xi_1|\xi_3)$. However, $P(\xi_1|\xi_3) = 1 - P_M$, where P_M is the symbol error probability. Hence $P(\xi_1|\xi_3)$ is well approximated (and upper-bounded) by unity for reasonably low symbol error probabilities. Therefore the probability of the error event ε is well approximated and upper-bounded as [14]:

$$P(\varepsilon) \le \frac{1}{2} erfc\left[\sqrt{\frac{d^2}{N_0}\delta^2(\boldsymbol{\varepsilon})}\right] \prod_{i=0}^{l-L-1} \frac{M - \Lambda_i}{M} \tag{3-22}$$

Let E be the set of all error events $\boldsymbol{\varepsilon}$ starting at time k and let $w(\boldsymbol{\varepsilon})$ be the corresponding number of nonzero components (Hamming distance of symbol errors) in each error event $\boldsymbol{\varepsilon}$. Hamming distance is a measure of the difference or "distance" between two binary sequences of equal length; in particular, Hamming distance is the number of bits which differ between the sequences. Then the probability of a symbol error is upper-bounded as

$$\begin{aligned} P_M &\le \sum_{\varepsilon \in E} w(\boldsymbol{\varepsilon}) P(\boldsymbol{\varepsilon}) \\ &\le \frac{1}{2} \sum_{\varepsilon \in E} w(\varepsilon) erfc\left[\sqrt{\frac{d^2}{N_0}\delta^2(\boldsymbol{\varepsilon})}\right] \prod_{i=0}^{l-L-1} \frac{M-\Lambda_i}{M} \end{aligned} \tag{3-23}$$

Now let D be the set of all $\delta(\boldsymbol{\varepsilon})$. For each $\delta \in D$, let E_δ be the subset of error events for which $\delta(\boldsymbol{\varepsilon}) = \delta$. Then (3-23) may be expressed as

$$\begin{aligned} P_M &\le \frac{1}{2} \sum_{\delta \in D} erfc \sqrt{\frac{d^2}{N_0}\delta^2(\boldsymbol{\varepsilon})} \left[\sum_{\varepsilon \in E_\delta} w(\boldsymbol{\varepsilon}) \prod_{i=0}^{l-L-1} \frac{M-\Lambda_i}{M} \right] \\ &\le \frac{1}{2} \sum_{\delta \in D} K_\delta erfc \sqrt{\frac{d^2}{N_0}\delta^2(\boldsymbol{\varepsilon})} \end{aligned} \tag{3-24}$$

where

$$K_\delta = \sum_{\varepsilon \in E_\delta} w(\boldsymbol{\varepsilon}) \prod_{i=0}^{l-L-1} \frac{M-\Lambda_i}{M} \tag{3-25}$$

In general, however, the use of the error state diagram for computing P_M is tedious. Instead we may simplify the computation of P_M by focusing on the dominant term in the summation of (3-24). Due to the exponential dependence of each term in the sum, the expression P_M is dominated by the term corresponding to the minimum value of δ, denoted as $\delta_{\min}$. Hence the symbol error probability may be approximated as

$$P_M \approx \frac{1}{2} K_{\delta_{\min}} erfc \sqrt{\frac{d^2}{N_0} \delta_{\min}^2} \tag{3-26}$$

where

$$K_{\delta_{\min}} = \sum_{\varepsilon \in E_{\delta_{\min}}} w(\varepsilon) \prod_{i=0}^{l-L-1} \frac{M - \Lambda_i}{M} \tag{3-27}$$

In general, $\delta_{\min}^2 \leq 1$. Hence $10 \log \delta_{\min}^2$ represents the loss in SNR due to inter-symbol interference. It is proved that in many ISI cases the performance of MLSE can achieve or approach the performance of the cases where inter-symbol interference was absent.

Example of MLSE: Consider a two-path channel ($L = 1$) with arbitrary coefficients f_0 and f_1 satisfying the constraint $f_0^2 + f_1^2 = 1$. Let $F(z), \varepsilon(z), \alpha(z)$ represent the Z-transform of sequence $\{f_i\}, \{\varepsilon_i\}, \{\alpha_i\}$, respectively. The channel characteristic is

$$F(z) = f_0 + f_1 z^{-1} \tag{3-28}$$

For an error event of length n,

$$\varepsilon(z) = \varepsilon_0 + \varepsilon_1 z^{-1} + \cdots + \varepsilon_{n-1} z^{-(n-1)} \quad n \geq 1 \tag{3-29}$$

The (3-15) may be expressed as the product $\alpha(z) = F(z)\varepsilon(z)$

$$\alpha(z) = \alpha_0 + \alpha_1 z^{-1} + \cdots + \alpha_n z^{-n} \tag{3-30}$$

where $\alpha_0 = \varepsilon_0 f_0$ and $\alpha_n = f_1 \varepsilon_{n-1}$. Since $\varepsilon_0 \neq 0$, $\varepsilon_{n-1} \neq 0$, and

$$\delta^2(\varepsilon) = \sum_{k=0}^{n} \alpha_k^2 \tag{3-31}$$

it follows that

$$\delta_{\min}^2 \geq f_0^2 + f_1^2 = 1 \tag{3-32}$$

Indeed, $\delta_{\min}^2 = 1$ when a single error occurs, i.e. $\varepsilon(z) = \varepsilon_0$. Thus we conclude that there is no loss in SNR in maximum-likelihood sequence estimation of the information symbols when the channel dispersion has length 1.

3.2 Multiple-symbol differential detection of *M*-PSK

Classical differential detection of *M*-PSK was based on observation of the received signal plus noise over an interval of two-symbol duration. Implicit in this process is the assumption that the phase introduced by the channel is constant over these two symbol intervals. This assumption is crucial to the analysis but is also realistic in many practical applications. Since the information is carried in the phase difference between adjacent phases, the information must be differentially encoded before transmission over the channel.

Although differential detection eliminates the need for carrier acquisition and tracking in the receiver, it suffers from a performance penalty (additional required

SNR at a given bit error rate) when compared to ideal (perfect carrier phase reference) coherent detection. Over the years, many researches are focused on enhancing the conventional (two symbol observation) differential detection technique so as to recover a portion of the performance lost relative to that of coherent detection and yet, still maintain a simple and robust implementation. *Divsalar* [71] proved that if the channel phase characteristic can be assumed constant over N_s symbols duration, $N_s > 2$, then by extending the observation interval to N_s symbols and applying an appropriate maximum-likelihood detection rule, one can further improve the performance of the differential detection system. In fact, by making N_s larger and larger, one can approach the performance of a coherent detection *M*-PSK system with differential encoding.

3.2.1 *Maximum-Likelihood Differential Detection of M-PSK based on a larger than two symbol observation interval*

Consider the transmission of *M*-PSK signals over the AWGN channel. The channel introduces the unknown phase θ, which in the absence of any information is assumed to be uniformly distributed in the interval $[-\pi, \pi]$. Assume that the phase θ is constant (independent of time) over a duration of N_s data symbols. The transmitted signal in the k^{th} transmission interval $kT_s \le t \le (k+1)T_s$ can be expressed in terms of the complex baseband signal

$$s_k = \sqrt{2E_s / T_s} \exp\left(j\theta_k\right) \tag{3-33}$$

where θ_k denotes the transmitted phase in the k^{th} interval which for *M*-PSK takes on values from the set $\beta_m = 2\pi m / M;\quad m = 0,1,2,\cdots M-1$. The bandpass representation of the received signal in this same time interval is given by [72]

$$r(t) = \mathrm{Re}\left\{\tilde{s}_k \exp\left(j\omega_c t + \theta\right) + \tilde{n}(t)\right\} = \mathrm{Re}\left\{\tilde{r}(t)\right\} \tag{3-34}$$

where $\tilde{n}(t)$ is a complex additive white Gaussian noise with single-sided PSD $2\sigma_n^2$. The receiver demodulates $\tilde{r}(t)$ with the complex reference signal $\exp\left\{-j\omega_c t\right\}$ to produce the complex baseband signal $\tilde{s}_k \exp\left(j\theta\right) + \tilde{n}(t)\exp\left(-j\omega_c t\right)$, which is then pass through a low pass filter, resulting in the complex detected sample

$$\tilde{\upsilon}_k = \tilde{s}_k \exp\left(j\theta\right) + \frac{1}{T_s}\int_{kT_s}^{(k+1)T_s} \tilde{n}(t)\exp\left(-j\omega_c t\right)dt = \tilde{s}_k \exp\left(j\theta\right) + \tilde{\eta}_k \tag{3-35}$$

Consider now an observation of the received signal over an interval of length $N_s T_s$ (N_s is referred to as the *block length*) and assume that the unknown phase θ is constant (independent of time) over this interval. In particular, for an observation of the received signal over the interval $(k - N_s + 1)T_s \le t \le (k+1)T_s$, the sequence $\tilde{\boldsymbol{\upsilon}} = \left(\tilde{\upsilon}_{k-N_s+1}, \tilde{\upsilon}_{k-N_s+2}, \cdots, \tilde{\upsilon}_{k-1}, \tilde{\upsilon}_k\right)$ conditioned on θ is a sufficient statistic for making a ML decision on the transmitted sequence $\tilde{\boldsymbol{s}} = \left(\tilde{s}_{k-N_s+1}, \tilde{s}_{k-N_s+2}, \cdots, \tilde{s}_{k-1}, \tilde{s}_k\right)$.

For the assumed AWGN model, the aposteriori probability of $\tilde{\upsilon}$ given $\tilde{s}$ and θ is

$$f(\tilde{\upsilon}|\tilde{s},\theta)=\frac{1}{(2\pi\sigma_n^2)^{N_s}}\exp\left\{-\frac{\left\|\tilde{\upsilon}-\tilde{s}\exp(j\theta)\right\|^2}{2\sigma_n^2}\right\}$$
$$=\frac{1}{(2\pi\sigma_n^2)^{N_s}}\exp\left\{-\frac{1}{2\sigma_n^2}\left[\sum_{i=0}^{N_s-1}\left(\left|\tilde{\upsilon}_{k-i}\right|^2+\left|\tilde{s}_{k-i}\right|^2\right)-2\left|\sum_{i=0}^{N_s-1}\tilde{\upsilon}_{k-i}\tilde{s}_{k-i}^*\right|\cos(\theta-\alpha)\right]\right\} \tag{3-36}$$

where

$$\alpha\triangleq\tan^{-1}\frac{\operatorname{Im}\left\{\sum_{i=0}^{N_s-1}\tilde{\upsilon}_{k-i}\tilde{s}_{k-i}^*\right\}}{\operatorname{Re}\left\{\sum_{i=0}^{N_s-1}\tilde{\upsilon}_{k-i}\tilde{s}_{k-i}^*\right\}} \tag{3-37}$$

Averaging (3-36) over the uniform PDF of θ and simplifying gives the conditional probability [72]:

$$f(\tilde{\upsilon}|\tilde{s})=\int_{-\pi}^{\pi}f(\tilde{\upsilon}|\tilde{s},\theta)f(\theta)d\theta$$
$$=\frac{1}{(2\pi\sigma_n^2)^{N_s}}\exp\left\{-\frac{1}{2\sigma_n^2}\sum_{i=0}^{N_s-1}\left(\left|\tilde{\upsilon}_{k-i}\right|^2+\left|\tilde{s}_{k-i}\right|^2\right)\right\}I_0\left(\frac{1}{\sigma_n^2}\left|\sum_{i=0}^{N_s-1}\tilde{\upsilon}_{k-i}\tilde{s}_{k-i}^*\right|\right) \tag{3-38}$$

where $I_0(x)$ is the zero-order modified Bessel function of the first kind. Noting that for *M*-PSK, $|\tilde{s}_k|^2$ is constant for all transmitted phases. Then, since $|\tilde{\upsilon}_k|^2$ is independent of the transmitted signal and since $I_0(x)$ is a monotonic function of x, maximizing $f(\tilde{\upsilon}|\tilde{s})$ over $\tilde{s}$ is equivalent to finding $\max_{\tilde{s}}\left|\sum_{i=0}^{N_s-1}\tilde{\upsilon}_{k-i}\tilde{s}_{k-i}^*\right|^2$. Defining the transmitted phase sequence $\boldsymbol{\theta}=\left(\theta_{k-N_s+1},\theta_{k-N_s+2},\cdots,\theta_{k-1},\theta_k\right)$, we obtain the decision rule:

$$\hat{\boldsymbol{\theta}}=\arg\max_{\boldsymbol{\theta}}\left|\sum_{i=0}^{N_s-1}\tilde{\upsilon}_{k-i}\exp\left(-j\theta_{k-i}\right)\right|^2 \tag{3-39}$$

where $\hat{\boldsymbol{\theta}}=\left(\hat{\theta}_{k-N_s+1},\hat{\theta}_{k-N_s+2},\cdots,\hat{\theta}_{k-1},\hat{\theta}_k\right)$ is the estimated sequence.

For differentially detect classical *M*-DPSK, we employ differentially encoding at the transmitter to resolve phase ambiguity. Letting reference phase $\theta_a=-\theta_{k-N_s+1}$, the transmitted information modulate in terms of $\Delta\theta_{k-i}=\theta_{k-i}-\theta_{k-i-1};\ \ i=0,1,\cdots,N_s-2$. Defining the information sequence $\Delta\hat{\boldsymbol{\theta}}=\left(\Delta\hat{\theta}_{k-N_s+2},\Delta\hat{\theta}_{k-N_s+3},\cdots,\Delta\hat{\theta}_{k-1},\Delta\hat{\theta}_k\right)$, then the above decision rule can be expressed as

$$\Delta\hat{\boldsymbol{\theta}}=\arg\max_{\Delta\boldsymbol{\theta}}\left|\sum_{i=0}^{N_s-1}\tilde{\upsilon}_{k-i}\exp\left\{-j\left(\sum_{m=0}^{N_s-i-2}\Delta\theta_{k-i-m}\right)\right\}\right|^2 \tag{3-40}$$

The decision variable can be rewritten as

$$Z=\left|\tilde{\upsilon}_{k-N_s+1}+\sum_{i=0}^{N_s-2}\tilde{\upsilon}_{k-i}\exp\left\{-j\left(\sum_{m=0}^{N_s-i-2}\Delta\theta_{k-i-m}\right)\right\}\right|^2 \tag{3-41}$$

Some special case of (3-41) are of interest. For $N_s=1$, that is , an observation of the received signal over one symbol interval, the decision variable simplifies to

$$Z=\left|\tilde{\upsilon}_k\right|^2 \tag{3-42}$$

which is completely independent of the input data phases and thus cannot be used for making decisions on differentially encoded *M*-PSK modulation.

Next, let $N_s=2$, in which case the decision variable becomes

$$Z=\left|\tilde{\upsilon}_{k-1}+\tilde{\upsilon}_k e^{-j\Delta\theta_k}\right|^2=\left|\tilde{\upsilon}_{k-1}\right|^2+\left|\tilde{\upsilon}_k\right|^2+2\,\mathrm{Re}\left\{\tilde{\upsilon}_k\tilde{\upsilon}_{k-1}^* e^{-j\Delta\theta_k}\right\} \tag{3-43}$$

This results in the equivalent decision rule

$$\Delta\hat{\boldsymbol{\theta}}=\arg\max_{\Delta\boldsymbol{\theta}}\mathrm{Re}\left\{\tilde{\upsilon}_k\tilde{\upsilon}_{k-1}^*\exp\left(-j\Delta\theta_k\right)\right\} \tag{3-44}$$

which is the classical differential detection of *M*-DPSK modulation over two symbols.

Next, we consider (3-41) for $N_s=3$. Here we have

$$\begin{aligned}Z&=\left|\tilde{\upsilon}_{k-2}+\tilde{\upsilon}_{k-1}\exp\left(-j\Delta\theta_{k-1}\right)+\tilde{\upsilon}_k\exp\left(-j\left(\Delta\theta_{k-1}+\Delta\theta_k\right)\right)\right|^2\\&=\left|\tilde{\upsilon}_{k-2}\right|^2+\left|\tilde{\upsilon}_{k-1}\right|^2+\left|\tilde{\upsilon}_k\right|^2+2\,\mathrm{Re}\left\{\tilde{\upsilon}_k\tilde{\upsilon}_{k-1}^*\exp\left(-j\Delta\theta_k\right)\right\}\\&+2\,\mathrm{Re}\left\{\tilde{\upsilon}_{k-1}\tilde{\upsilon}_{k-2}^*\exp\left(-j\Delta\theta_{k-1}\right)\right\}+2\,\mathrm{Re}\left\{\tilde{\upsilon}_k\tilde{\upsilon}_{k-2}^*\exp\left(-j\cdot\left(\Delta\theta_{k-1}+\Delta\theta_k\right)\right)\right\}\end{aligned} \tag{3-45}$$

Thus, the decision rule becomes

$$\Delta\hat{\boldsymbol{\theta}} = \arg\max_{\Delta\boldsymbol{\theta}} \mathrm{Re}\{\tilde{\upsilon}_k\tilde{\upsilon}_{k-1}^* \exp\left(-j\Delta\theta_k\right) + \tilde{\upsilon}_{k-1}\tilde{\upsilon}_{k-2}^* \exp\left(-j\Delta\theta_{k-1}\right) + \tilde{\upsilon}_k\tilde{\upsilon}_{k-2}^* \exp\left(-j\left(\Delta\theta_{k-1} + \Delta\theta_k\right)\right)\} \tag{3-46}$$

Note that the first and second terms of the metric used in the decision rule are identical to those used to make successive and independent decision on $\Delta\theta_k$ and $\Delta\theta_{k-1}$ in classical *M*-DPSK, respectively. The third term in the optimum metric is a combination of the first two and is required to make an optimum joint decision on $\Delta\theta_k$ and $\Delta\theta_{k-1}$. Clearly, a receiver implemented on the basis of (3-46) will outperform a classical *M*-DPSK receiver.

A receiver that implements the decision rule of (3-40) is illustrated in Figure 3-5.

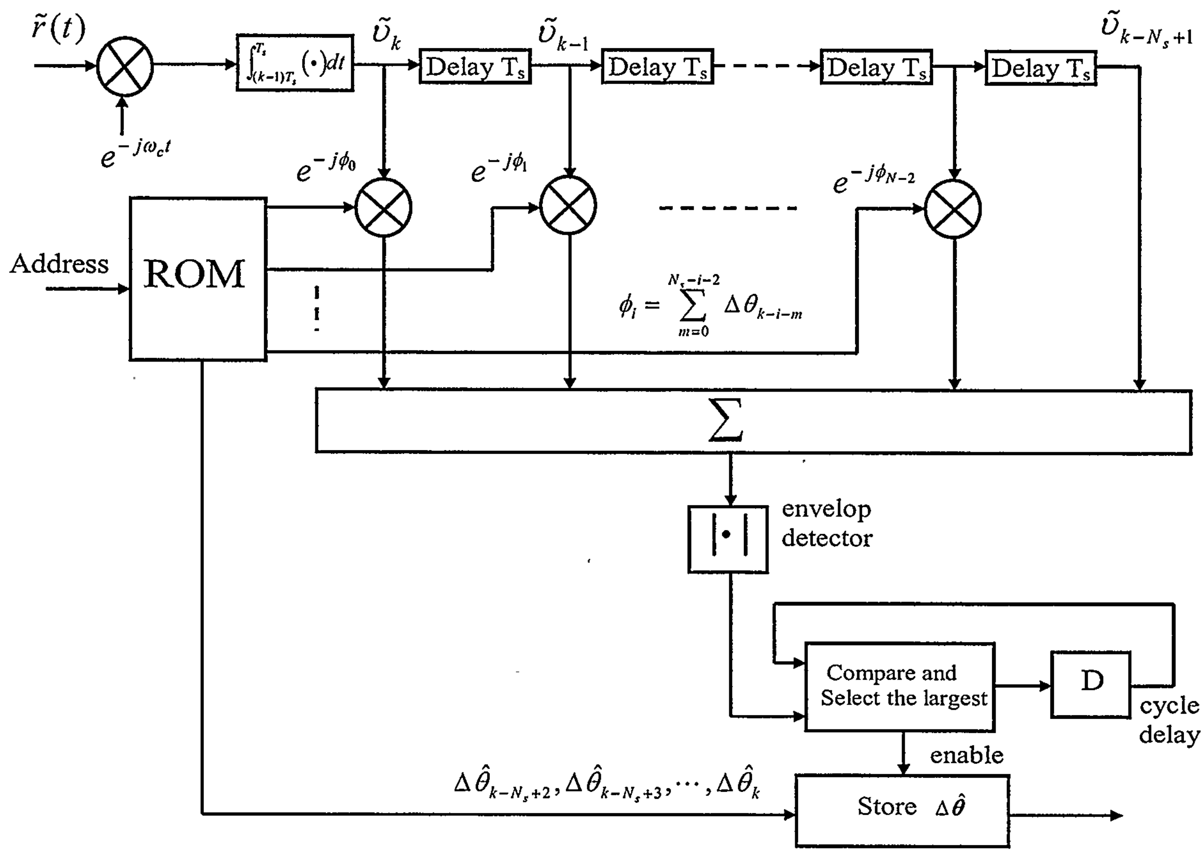

Figure 3-5: Direct implementation of MSDD of *M*-PSK

3.2.2 *Performance of Multiple-Symbol Differential Detection of M-PSK*

To obtain a simple upper bound on the average bit error probability, $P_b(E)$, of the proposed N_s symbols detection scheme, we use a union bound analogous to that used for upper bounding the performance of error correction coded systems. In particular, the upper bound on $P_b(E)$ is the sum of the pairwise error probabilities associated with each N_s-1 symbols error sequence. Each pairwise error probability is then either evaluated directly or itself upper bounded. Let

$\boldsymbol{\Delta\theta} = \left(\Delta\theta_{k-N_s+2}, \Delta\theta_{k-N_s+3}, \cdots, \Delta\theta_{k-1}, \Delta\theta_k\right)$ denote the sequence of $N_s - 1$ information phases and $\Delta\hat{\boldsymbol{\theta}} = \left(\Delta\hat{\theta}_{k-N_s+2}, \Delta\hat{\theta}_{k-N_s+3}, \cdots, \Delta\hat{\theta}_{k-1}, \Delta\hat{\theta}_k\right)$ be the corresponding sequence of estimated phases. Let $\boldsymbol{u}$ be the sequence of $b = (N_s - 1)\log_2 M$ information bits that produces $\boldsymbol{\Delta\theta}$ at the transmitter and $\hat{\boldsymbol{u}}$ be the sequence of b bits that result from the detection of $\Delta\hat{\boldsymbol{\theta}}$, then the bit error probability has the upper bound

$$P_b(E) \le \frac{1}{(N_s - 1)\log_2 M} \sum_{\boldsymbol{\Delta\theta} \ne \Delta\hat{\boldsymbol{\theta}}} w(\boldsymbol{u}, \hat{\boldsymbol{u}}) \Pr\left\{\hat{Z} > Z \middle| \boldsymbol{\Delta\theta}\right\} \tag{3-47}$$

where $w(\boldsymbol{u}, \hat{\boldsymbol{u}})$ denotes the Hamming distance between $\boldsymbol{u}$ and $\hat{\boldsymbol{u}}$ and $\Pr\left\{\hat{Z} > Z \middle| \boldsymbol{\Delta\theta}\right\}$ denotes the pairwise probability that $\Delta\hat{\boldsymbol{\theta}}$ is incorrectly chosen when indeed $\boldsymbol{\Delta\theta}$ was sent. The decision statistic Z is defined in (3-41) and the corresponding error statistic $\hat{Z}$ is identical to (3-41) with each $\Delta\theta_k$ replaced by $\Delta\hat{\theta}_k$.

From (3-41) we can conclude that decision variables $\hat{Z}$ and Z actually can be seemed as two independent Rician random variables with identical variance and arbitrary mean, respectively. Therefore, the pairwise probability of error $\Pr\left\{\hat{Z} > Z \middle| \boldsymbol{\Delta\theta}\right\}$ is given as [71]:

$$\Pr\left\{\hat{Z} > Z \middle| \boldsymbol{\Delta\theta}\right\} = \frac{1}{2}\left[1 - Q\left(\sqrt{b}, \sqrt{a}\right) + Q\left(\sqrt{a}, \sqrt{b}\right)\right] \tag{3-48}$$

where $Q(\alpha, \beta) \triangleq \int_\beta^\infty x \exp\left\{-\frac{x^2 + \alpha^2}{2}\right\} I_0(\alpha x)\, dx$ is referred to as the Marcum Q-function, whose arguments are evaluated in [71, 72] to be

$$\begin{Bmatrix} b \\ a \end{Bmatrix} = \frac{E_s}{2N_0}\left[N_s \pm \sqrt{N_s^2 - |\delta|^2}\right] \tag{3-49}$$

with

$$\delta \triangleq \sum_{i=0}^{N_s-1} \exp\left\{ j \sum_{m=0}^{N_s-j-2} \left(\Delta\theta_{k-j-m} - \Delta\hat{\theta}_{k-j-m}\right)\right\} \tag{3-50}$$

In (3-50), it is understood that the summation in the exponent evaluates to zero if the upper index is negative.

Special Case: Classical DPSK ($N_s = 2, M = 2$)

From (3-50), we immediately get $\delta = 0$ and thus (3-49) can be expressed as

$$\begin{Bmatrix} b \\ a \end{Bmatrix} = \begin{Bmatrix} \dfrac{2E_s}{N_0} \\ 0 \end{Bmatrix} \tag{3-51}$$

Substituting (3-51) into (3-48) gives

$$\Pr\left\{\hat{Z} > Z \middle| \boldsymbol{\Delta\theta}\right\} = \frac{1}{2}\left[1 - Q\left(\sqrt{\frac{2E_s}{N_0}}, 0\right) + Q\left(0, \sqrt{\frac{2E_s}{N_0}}\right)\right] \tag{3-52}$$

From the definition of the Marcum Q-function

$$Q(\alpha,0)=1; \quad Q(0,\beta)=\exp\left(-\frac{\beta^2}{2}\right) \tag{3-53}$$

Since for the binary case the pairwise error probability is indeed equal to the bit error probability, then we have from (3-52) and (3-53) that

$$P_b(E)=\frac{1}{2}\exp\left(-\frac{E_b}{N_0}\right) \tag{3-54}$$

which is identical with the bit error probability of classical DPSK in [72].

In the binary case for arbitrary N_s, the approximate upper bound asymptotic expression of the average bit error probability is given by [71]:

$$P_b(E)\leq\frac{2}{\sqrt{\pi\frac{E_s}{N_0}}}\left(\sqrt{\frac{N_s-1}{N_s-2}}\right)\left[\frac{1}{2}\exp\left\{-\frac{E_s}{N_0}\right\}\right] \tag{3-55}$$

The exact bit error probability performance for coherent detection of BPSK with differential encoding has an asymptotic expression [72]:

$$P_b=erfc\sqrt{\frac{E_s}{N_0}}\left[1-\frac{1}{2}erfc\sqrt{\frac{E_s}{N_0}}\right] \tag{3-56}$$

As one might expect, the performance of multiple symbol differentially detected BPSK approaches the limit of ideal coherent detection BPSK with differential encoding as the observation interval approaches infinity.

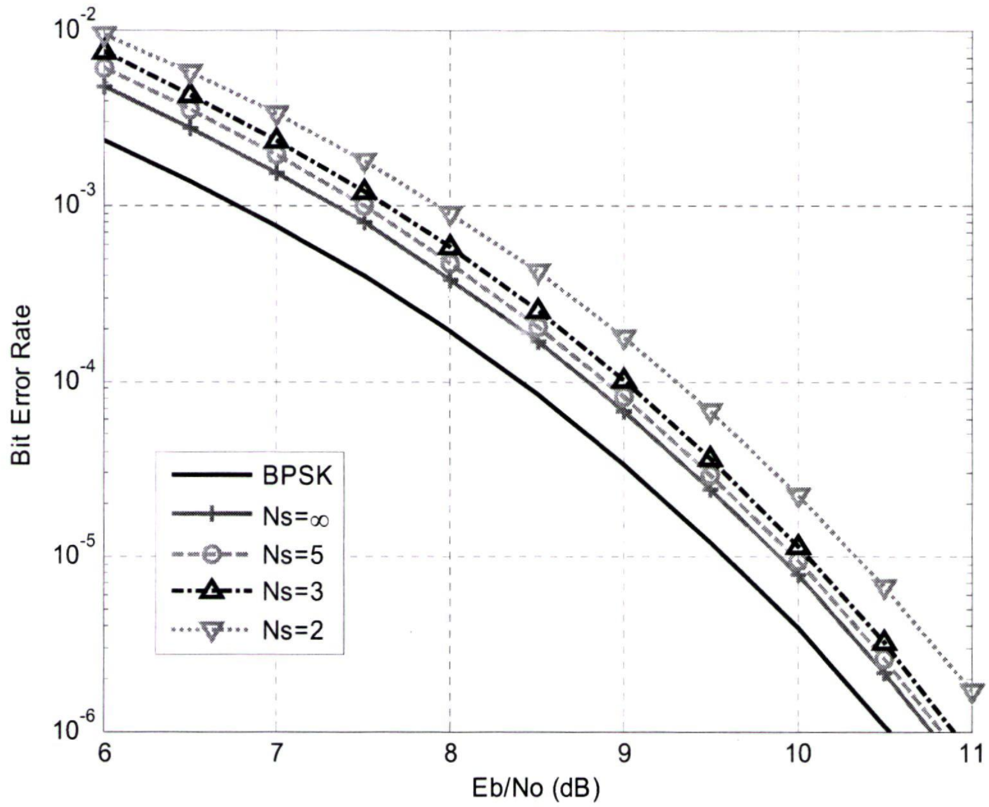

Figure 3-6: Bit error probability versus Eb/No for MSDD of DPSK

Figure 3-6 illustrates the upper bit error bounds of *multiple symbols differential detection* (MSDD) for the DPSK signals. In the figure, the length of the observation interval is a parameter varying from $N_s = 2$ (classical differential detection of DPSK signals), $N_s = 3,5$ (multiple symbols differential detection of DPSK signals) to $N_s = \infty$ (ideal coherent detection of BPSK signals with differential encoding). As a benchmark, the bit error rate of the coherent detection of BPSK signals without differential encoding is also shown in the figure. The differential encoding in BPSK signals has a performance penalty comparing to that without differential encoding, since a single symbol error always causes another symbol error happen in the differential decoding process. We observe from the figure that, for example, extending the observation interval from $N_s = 2$ to $N_s = 3$ recovers more than half of the E_b / N_0 loss of differential detection versus coherent detection of BPSK signals with differential encoding.

3.3 Summary

In the chapter two sequence detection techniques are introduced in depth. The maximum-likelihood sequence estimation (MLSE) of digital sequences is the optimal equalizer in the presence of inter-symbol interference (ISI) and additive Gaussian noise. The MLSE algorithm can be implemented recursively by the Viterbi algorithm (VA) in the fixed computational rate. The Viterbi algorithm (VA) is illustrated via the trellis diagrams in the chapter. The derivation process of *M*-PAM symbol error probability upper-bound for the MLSE algorithm in the ISI and AWGN channel is demonstrated also. The multiple symbols differential detection (MSDD) algorithm is invented to improve the performance of the classical differential detection system. The derivation process of the *M*-PSK symbol error probability upper-bound for the MSDD algorithm in the AWGN channel is introduced in the chapter. The performance of MSDD algorithm approaches the limit of ideal differential coherent detection as the observation interval approaches infinity. Stemmed from the MLSE and MSDD algorithms, a novel sequence detection technique named as maximum likelihood differential sequence estimation (MLDSE) is introduced in the next chapter.

Chapter 4. Maximum Likelihood Differential Sequence Estimation of DS-UWB Signals

The chapter describes the proposed maximum likelihood differential sequence estimation (MLDSE) receiver for the IEEE802.15.3a DS-UWB scheme. The MLDSE receiver stems from the maximum likelihood sequence estimation (MLSE) and the multiple symbols differential detection (MSDD) technologies. The innovation of the MLDSE receiver is to perform the MLSE equalization and the MSDD reception in one algorithm without increasing any hardware. The MLDSE algorithm can be implemented recursively by a modified Viterbi algorithm (VA). Therefore the MLDSE receiver owns both merits of the MLSE and the MSDD technologies. The demodulation function of DSSS signals also can be incorporated in the modified Viterbi algorithm (VA). The differential M-PSK symbol error probability upper-bound of MLDSE algorithm in the ISI and AWGN channel is derived using the similar approaches in the MLSE and MSDD derivations. To implement the MLDSE detection the full knowledge of multipath channel response parameters is required. To achieve the channel parameters, an adaptive joint data and channel estimation scheme is proposed for the MLDSE receiver scheme. The drawbacks of Rake receiver in the DS-UWB scheme of IEEE802.15.3a are also detailed in the chapter.

4.1 The Modulation and Channel Model

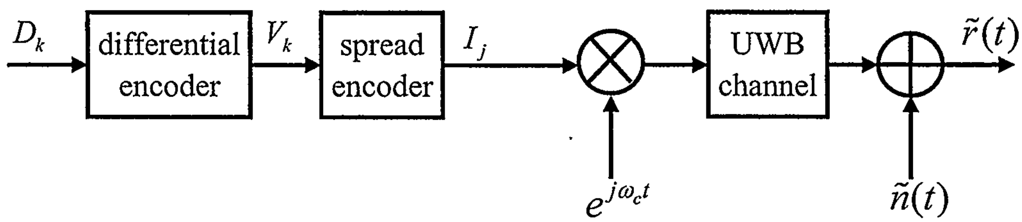

Figure 4-1: Modulation and channel model of DS-UWB *M*-DPSK signals

Figure 4-1 illustrates the block diagram of the differential MPSK encoding and DSSS modulation scheme as the transmission over the UWB channel. The differentially encoded MPSK signal in the k^{th} symbol transmission interval $kT_s \le t \le (k+1)T_s$ is denoted in terms of the complex baseband signal $V_k = \exp(j\theta_k)$, where $\theta_k = 2\pi m / M,\quad m = 0,1,2,\cdots,M-1$. The differential phase $\Delta\theta_k = \theta_k - \theta_{k-1}$ represents the input MPSK data corresponding to the k^{th} symbol interval, $D_k = \exp(j\Delta\theta_k)$. The spread encoder implements N_c folds spectrum spreading by a direct-sequence spread code $\{c_1, c_2, \cdots, c_{N_c}\}$ within one symbol duration T_s, thus one symbol has N_c chips ($T_s = N_c T_c$). In the IEEE802.15.3a DS-UWB scheme, the spreading code is the ternary sequence with value $c_i \in \{-1,0,1\},\quad i \in \{1,2,\cdots,N_c\}$. I_j defined as the output signal of spread encoder in the j^{th} transmission chip interval is given as:

$$I_j = c_i \cdot V_k = c_i \exp\left(j\theta_k\right), \quad j = N_c(k-1)+i \tag{4-1}$$

The carrier $\exp\left(j\omega_c t\right)$ modulates the baseband spread spectrum signal to radio frequency. The UWB channel is a severe multipath channel and $\tilde{n}(t)$ is a complex additive white Gaussian process.

To evaluate different receiver development schemes in the UWB band, IEEE 802.15.3a Channel Model Committee provides four channel models of indoor transmission environment for the distinct applications. The principle channel parameters based on the measurement statistics for different indoor channel are shown in the Table 4-1 [53].

Model Channel Characteristics	CM 1	CM 2	CM 3	CM 4
Mean excess delay (nsec) (τ_m)	5.0	9.9	15.9	30.1
RMS delay (nsec) (τ_{rms})	5	8	15	25
NP_{10dB}	12.5	15.3	24.9	41.2
NP (85%)	20.8	33.9	64.7	123.3
Channel energy mean (dB)	-0.4	-0.5	0.0	0.3
Channel energy std (dB)	2.9	3.1	3.1	2.7

Table 4-1: Multipath channel characteristics of IEEE 802.15.3a Channel Models

CM1: 0 to 4 meters, line of sight

CM2: 0 to 4 meters, non-line of sight

CM3: 4 to 10 meters, non-line of sight

CM4: 4 to 10 meters, extreme non-line of sight

NP_{10dB}: Defined as the number of statistically independent multipath components that are within 10 dB of the peak multipath arrival.

NP(85%): Defined as the number of statistically independent multipath components that are accounted for 85% of total multipath energy.

4.2 The MLDSE receiver scheme

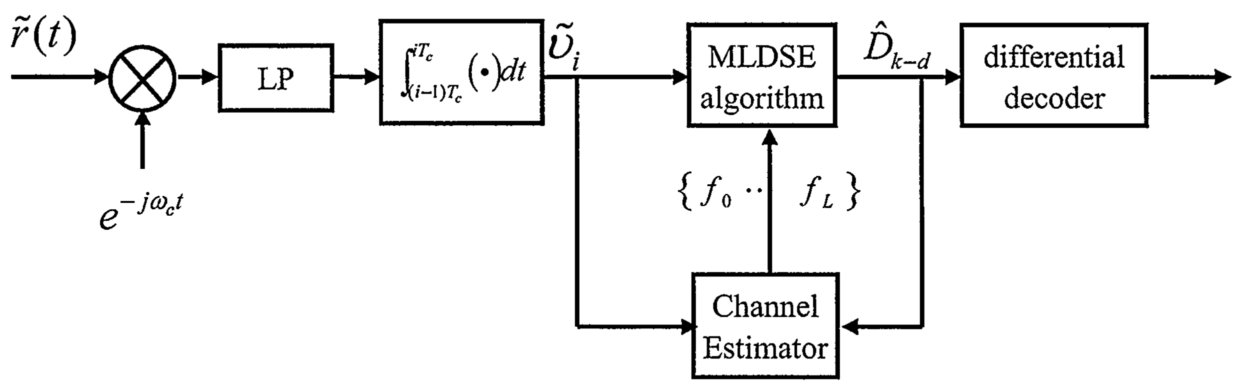

Figure 4-2: The block diagram of the MLDSE receiver scheme

Figure 4-2 illustrates the basic blocks of the maximum likelihood differential sequence estimation receiver for the differential encoding DS-UWB scheme. The receiver demodulates complex received signal $\tilde{r}(t)$ with a complex reference signal $\exp(-j\omega_c t)$ to produce the complex baseband signal. The complex baseband signal is then passed through a chip interval integration and dump (I&D), resulting in the complex detected chip samples:

$$\tilde{\upsilon}_i = \exp(j\theta)\left(\sum_{j=0}^{L} f_j \tilde{I}_{i-j}\right) + \tilde{\eta}_i \tag{4-2}$$

where $\tilde{I}_i$ represents the transmitted complex signal in the i^{th} transmission chip interval. $\tilde{\eta}_i$ denotes the complex noise component in the i^{th} chip sample after the I&D process. $\{f_j\}$ defines the equivalent chip interval discrete-time version of the UWB channel response and $\{f_j\}$ spans $L+1$ chips. The UWB channel introduces the unknown channel phase delay, θ, to the transmitted signals. In the absence of any information the channel phase delay θ is assumed to be uniformly distributed in the interval $(-\pi,\pi)$. The noise samples $\tilde{\eta}_i$ still conforms to the complex white Gaussian distribution after the I&D process. Implication in the reception process is that the chip boundary synchronization and spreading code synchronization are achieved.

To perform the multiple symbols differential detection and the maximum likelihood sequence estimation in the multipath channel at the same time, we modify the decision rule (3-39) of the multiple symbols differential detection to achieve a new algorithm named as the *Maximum Likelihood Differential Sequence Estimation* (MLDSE) algorithm. Defining that the transmitted N chips sequence is $\boldsymbol{I} = (I_1, I_2, \cdots, I_{N-1}, I_N)$ and the detected chips sequence is

$\hat{\boldsymbol{I}}=\left(\hat{I}_1,\hat{I}_2,\cdots,\hat{I}_{N-1},\hat{I}_N\right)$. The MLDSE decision rule of a N chips sequence is given as:

$$\hat{\boldsymbol{I}}=\arg\max_{I}\left|\sum_{i=1}^{N}\tilde{\upsilon}_i\left(\sum_{j=0}^{L}f_j\tilde{I}_{i-j}^{*}\right)\right| \tag{4-3}$$

Where superscript * denotes the conjugation of a complex signal.

Defining a possible chip sequence as $\boldsymbol{\Lambda}=\left(\Lambda_1,\Lambda_2,\cdots,\Lambda_{N-1},\Lambda_N\right)$ and substituting (4-2) into (4-3), the decision variable Z of the MLDSE algorithm can be expressed as:

$$\begin{aligned}
Z&=\left|\sum_{i=1}^{N}\tilde{\upsilon}_i\left(\sum_{j=0}^{L}f_j\tilde{\Lambda}_{i-j}^{*}\right)\right| \\
&=\left|\exp(j\theta)\sum_{i=1}^{N}\sum_{j=0}^{L}f_j^2\left(\tilde{I}_{i-j}\tilde{\Lambda}_{i-j}^{*}\right)+\sum_{i=1}^{N}\eta_i\left(\sum_{j=0}^{L}f_j\tilde{\Lambda}_{i-j}^{*}\right)\right| \\
&=\left|\exp(j\theta)\right|\left|\sum_{i=1}^{N}\sum_{j=0}^{L}f_j^2\left(\tilde{I}_{i-j}\tilde{\Lambda}_{i-j}^{*}\right)+\exp(-j\theta)\sum_{i=1}^{N}\eta_i\left(\sum_{j=0}^{L}f_j\tilde{\Lambda}_{i-j}^{*}\right)\right| \\
&=\left|\sum_{i=1}^{N}\sum_{j=0}^{L}f_j^2\left(\tilde{I}_{i-j}\tilde{\Lambda}_{i-j}^{*}\right)+\exp(-j\theta)\sum_{i=1}^{N}\eta_i\left(\sum_{j=0}^{L}f_j\tilde{\Lambda}_{i-j}^{*}\right)\right|
\end{aligned} \tag{4-4}$$

The first and second items in (4-4) represent the signal component and the noise component in the decision variable respectively. We can find the channel phase delay θ has no impact on the result of (4-3), therefore the implementation of the MLDSE algorithm eliminates the need of the carrier synchronization process. Because the decision variable can achieve the maximum while $\boldsymbol{\Lambda}=\pm\boldsymbol{I}$, the transmitted information sequence $\boldsymbol{I}$ need be differentially encoded to avoid the ambiguity. For the simplicity, the MLDSE algorithm of (4-3) excludes the differential decoding process. Therefore a differential decoder following the MLDSE algorithm is needed to restore the transmitted data.

The *Viterbi algorithm* (VA) used in the maximum likelihood sequence estimation of digital sequence in the presence of ISI can also be modified to implement MLDSE algorithm by computing the decision variable recursively. Analogous to (3-7) of the MLSE algorithm, the recursive decision variable Z_k of the k^{th} chip intervals in the MLDSE algorithm can expressed as

$$Z_k = Z_{k-1} + \tilde{\upsilon}_k \left(\sum_{j=0}^{L} f_j \tilde{I}^*_{k-j} \right) \tag{4-5}$$

In each stage of the Viterbi algorithm, only M^L surviving sequences from the previous stage are needed to calculate the M probabilities of the current stage, therefore total M^{L+1} probabilities are computed each stage. Some modifications in the Viterbi algorithm for the MLDSE detection include: I) The branch metrics are denoted as $\tilde{\upsilon}_k \left(\sum_{j=0}^{L} f_j \tilde{I}^*_{k-j} \right)$ and the path metrics are denoted as $|Z_k|$. II) The Viterbi algorithm iteratively finds the path with the relatively larger path metric to each node and discards the others. A simply modified VA example for illustrative purposes is given here.

Example of MLDSE algorithm: The binary transmitted data $I_k \in \{0,1\}$ is modulated to the DPSK signal $V_k = \exp(j\theta_k)$ with

$$\Delta\theta_k = \theta_k - \theta_{k-1} = \begin{cases} 0 & I_k = 0 \\ \pi & I_k = 1 \end{cases} \tag{4-6}$$

The multipath channel model ($f_0 = 1$, $f_1 = 0.5$) is shown in Figure 4-3 (a). The channel phase delay θ is uniformly distributed in $(-\pi, \pi)$. The channel memory is therefore $L = 1$, and the state is $\Psi_k = V_{k-1}$. There are two possible states $V_{k-1} = \pm 1$ for the DPSK modulation. The state transition diagram for this example is shown in Figure 4-3 (b), where the arcs are label with the input/output pair (V_k, S_k).

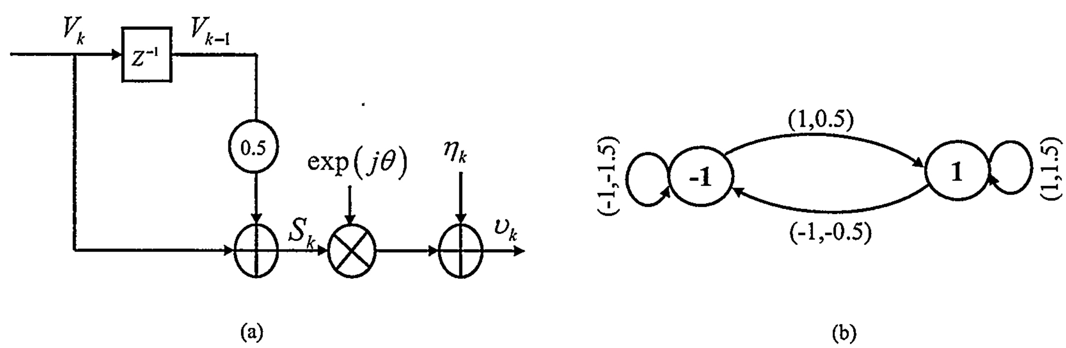

Figure 4-3: (a) The channel model. (b) The state transition diagram

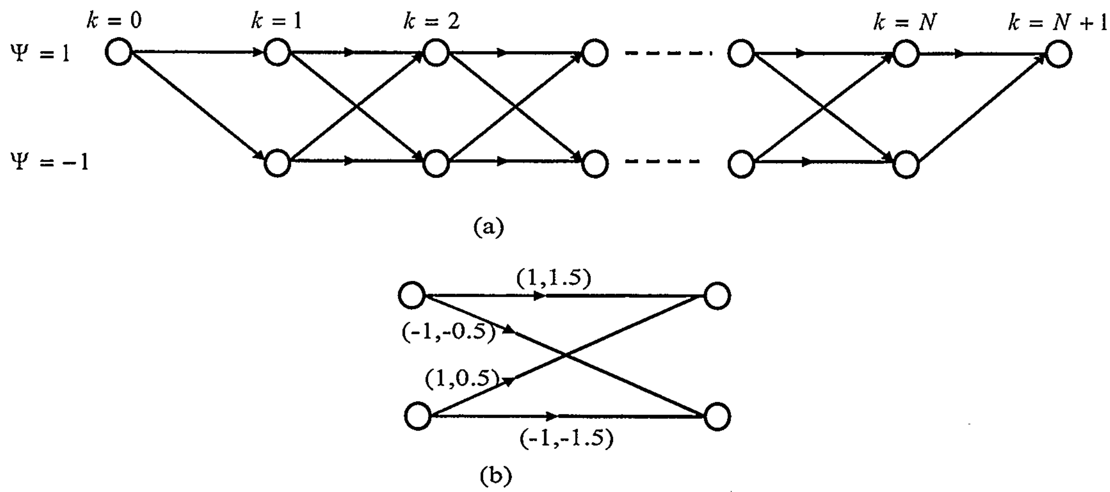

Figure 4-4: The trellis diagrams of the possible state transition

Figure 4-4 illustrates: (a) a two-state trellis diagram of the possible state transition, assuming the initial and final state are $\Psi = 1$. (b) one stage of the trellis diagram where each branch is labeled with the input and output pair (V_k, S_k) corresponding to that state transition.

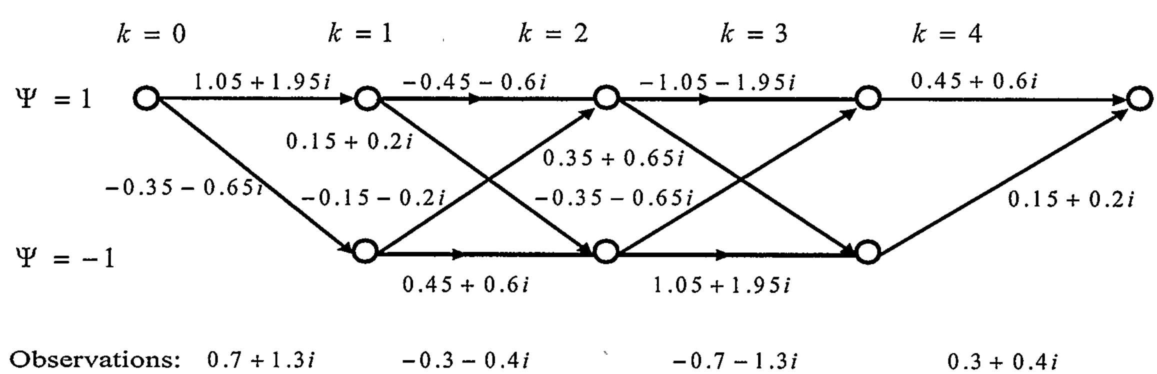

Figure 4-5: The branch metrics of the MLDSE algorithm

The trellis diagram shown in Figure 4-5 is marked with branch metrics corresponding to the transmission of $N = 3$ symbols and observing

$\tilde{\upsilon} = \{0.7+1.3i, -0.3-0.4i, -0.7-1.3i, 0.3+0.4i\}$ for this example. The branch metrics denoted as $\tilde{\upsilon}_k \sum_{i=0}^{1} f_i \tilde{V}^*_{k-i} = \tilde{\upsilon}_k \tilde{S}^*_k$ are labeled in corresponding branch. For an instance, the branch metric of the branch from the node $\Psi_0 = 1$ to the node $\Psi_1 = -1$ is given as $(0.7+1.3i)\times(-0.5) = -0.35-0.65i$.

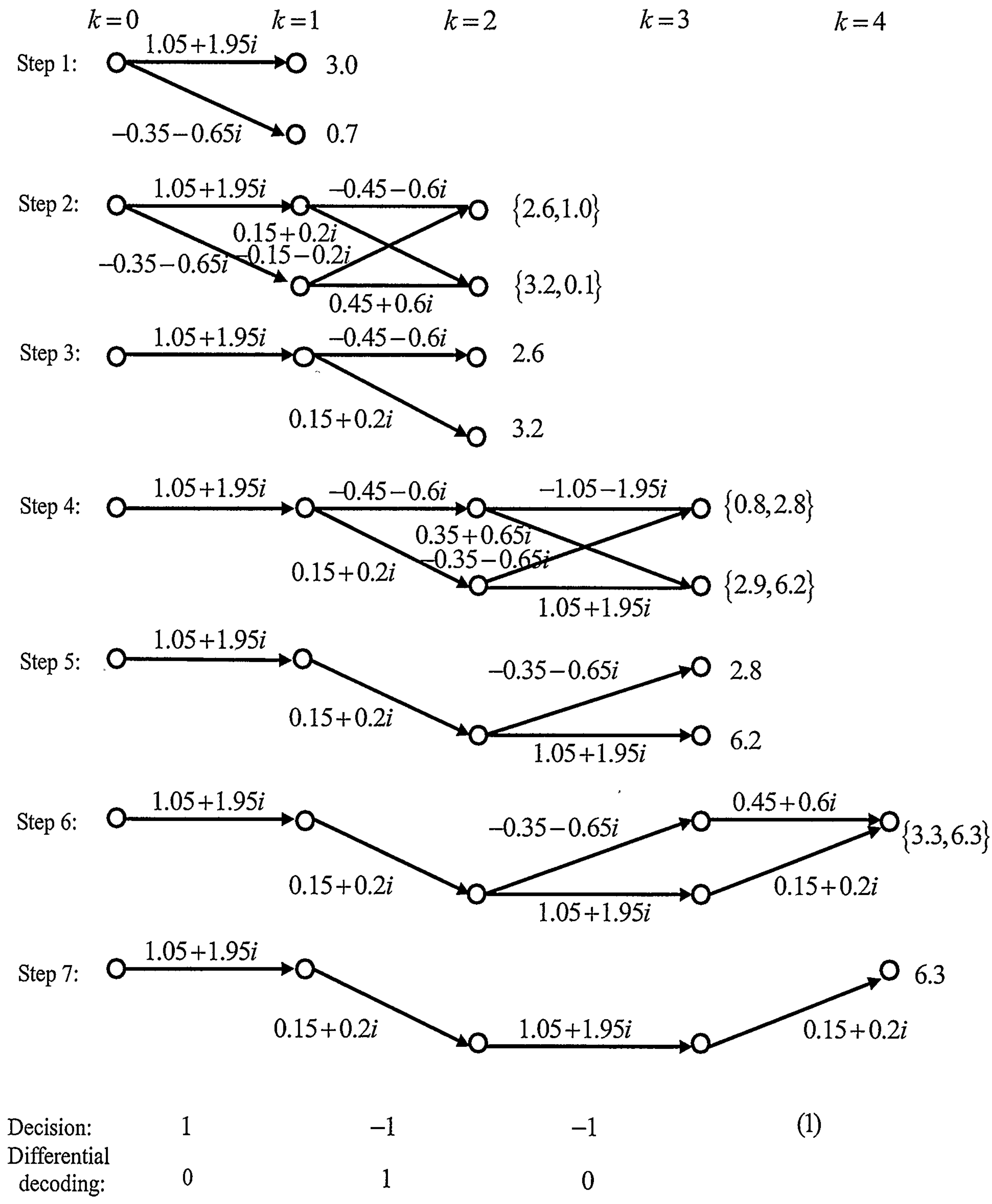

Figure 4-6: An illustrated iterative procedure of MLDSE algorithm

In this example, the branch metrics are labeled with each branch. The partial path metrics which are the absolute values of the sum of the branch metrics in the path are labeled in the end of the path. The MLDSE algorithm iteratively finds the path with the maximum path metric to the each node and discards the others.

Step 1: Each node at $k=1$ has one incoming branch from a known node ($\Psi_0=1$), the two paths are survivor. The partial path metrics are computed by $|1.05+1.95i|=3$ and $|-0.35-0.65i|=0.7$ respectively.

Step 2: Each node at $k=2$ have two incoming branches from previous nodes, computes the partial path metrics of each path to that node. For the node $\Psi_2=1$, two partial path metrics are computed by $|(1.05+1.95j)+(-0.45-0.6j)|=2.6$ and $|(-0.35-0.65i)+(-0.15-0.2i)|=1.0$ respectively. For the node $\Psi_2=-1$, two partial path metrics are computed by $|(1.05+1.95i)+(0.15+0.2i)|=3.2$ and $|(-0.35-0.65i)+(0.45+0.6i)|=0.1$ respectively.

Step 3: Select the path with the larger path metric as a survivor. For the node $\Psi_2=1$, the path with path metrics of 2.6 is the survivor. For the node $\Psi_2=-1$, the path with path metrics of 3.2 is the survivor. The surviving paths to each node are shown. The other two paths with relatively smaller path metric are discarded.

Step 4: Each node at $k=3$ have two incoming branches from previous nodes, computes the partial path metrics of each path to that node. For the node $\Psi_3=1$, two partial path metrics are 0.8 and 2.8 respectively. For the node $\Psi_3=-1$, two partial path metrics are 2.9 and 6.2 respectively.

Step 5: Select the path with the smaller path metric as a survivor. For the node $\Psi_3 = 1$, the path with path metrics of 2.8 is the survivor. For the node $\Psi_3 = -1$, the path with path metrics of 6.2 is the survivor. The surviving paths to each node are shown. The other two paths with relatively smaller path metric are discarded.

Step 6: All the paths will enter a known ending node ($\Psi_4 = 1$), compute all path metrics from the previous surviving paths. To the $\Psi_4 = 1$, the path metrics are 3.3 and 6.3 respectively.

Step 7: The path with path metric of 6.3 is used to make the final decision and the path with path metric of 3.3 will be discarded.

Results of the example: The decided DPSK signal is $\{1,-1,-1\}$. After the differential decoding, the binary transmitted data is $\{0,1,0\}$.

In the Viterbi algorithm, total M^{L+1} possibilities are needed to compute at each stage except the initial and ending stages. When the process of dispreading the spread spectrum signals is incorporated in the MLDSE detection, the resource needed by the Viterbi algorithm operation can be greatly reduced. Substituting the equation (4-1) into the equation (4-5), the decision variable can be rewritten as:

$$Z_k = Z_{k-1} + \tilde{\upsilon}_k \sum_{j=0}^{L} f_j c_a \tilde{V}_b^* \tag{4-7}$$

Where $a \triangleq (k-j)\%M$, $b \triangleq \left\lceil \frac{k-j}{M} \right\rceil$. The sign $\lceil x \rceil$ represents the minimum integer larger than x, and the sign $x\%y$ denotes the remainder of the integer division $(x \div y)$. For example, the value of $7\%5$ is equal to 2. Defining the channel memory L be the integer folds of spreading code length N_c, In each stage of the Viterbi algorithm, only M^{L/N_c} possible sequences from the previous stage are needed to calculate the M probabilities of the current stage, therefore total M^{L/N_c+1} probabilities are computed each stage.

Therefore, the MLDSE scheme for the DS-UWB communications is capable to perform three functions at one time: I) maximum likelihood sequence estimation equalization, II) multiple symbols differential detection, and III) the decoding of the direct sequence encoding.

4.3 The Adaptive Channel Estimator

The proposed MLDSE receiver will improve the reception performance for the IEEE 802.15.3a DS-UWB scheme at the expense of increasing complexity to the multipath channel estimation process. There are more path parameters of the UWB channel, including the ICI and ISI multipath components, needed to be estimated. To simplify the UWB channel estimation process, a joint data and channel estimation strategy is proposed to perform the evaluation of the discrete-time channel dispersion coefficients $\{f_j\}$ in the chip interval for the MLDSE reception scheme through some adaptive signal estimation algorithms.

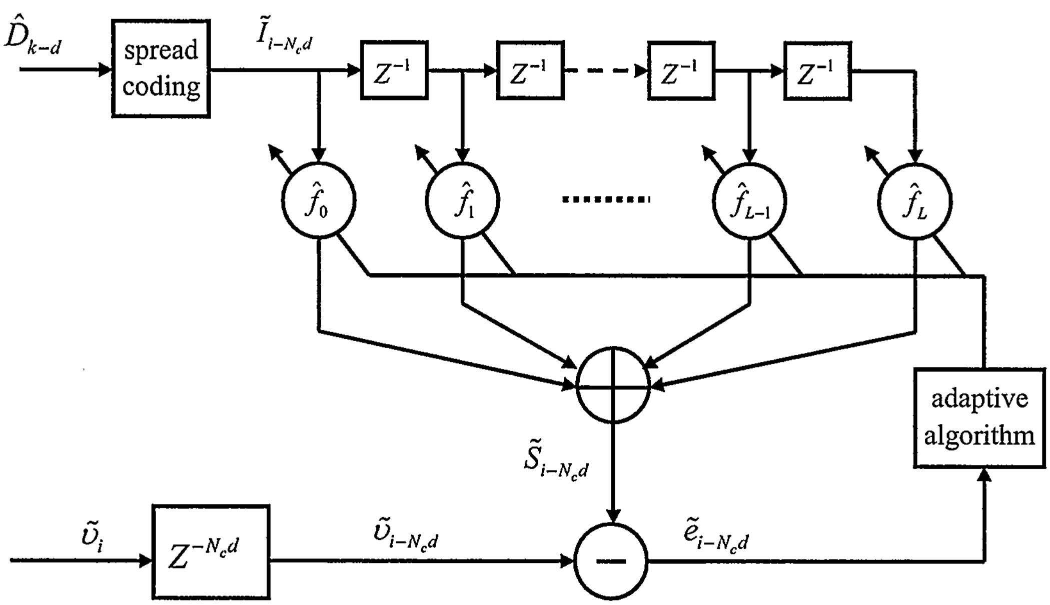

Figure 4-7: The Adaptive channel estimator

Figure 4-7 illustrates the structure of the channel dispersion coefficients estimation using an adaptive algorithm. The implementation of the adaptive algorithm requires the channel is static or slowly time-varying for the transmitted duration. The channels of the WPAN applications conform to the requirement. The DS-UWB scheme is used mainly in the indoor environment for the high speed data transmission. In the indoor environment, the transmitter and the receiver generally are fixed or move at a pedestrian speed, the barriers and scatters are also low speed objects. To the high speed data transmission, the indoor channel can be modeled as quasi-static in a short time. Several indoor channel measurements confirm it [13, 48, 52]. For example, in the DS-UWB scheme, a packet of 2^{15} bits transmitting at data rate 110Mbps will last only 0.298ms, the channel within the packet time can be seemed as a static channel. Some Adaptive algorithms need a training sequence which is sent to facilitate the rapid convergence. In the DS-UWB scheme, each packet has a preamble ($5\mu s \sim 30\mu s$) which is used for clock/carrier acquisition [53]. The length of the preamble is depended on the data rate and the extent of the channel dispersion. The preamble can also be used as a training sequence to the channel estimator. A lot of adaptive algorithms can be used in the channel estimator, such as *recursive least-squares* (RLS) algorithm and *least mean-squared* (LMS) algorithm, etc.

4.4 Performance of the MLDSE Scheme

To simplify the performance analysis of the proposed MLDSE scheme, the spread code $\boldsymbol{C}=[1]$ is considered in the analysis and UWB channel coefficients are normalized to unity, $\sum_{n=0}^{L} f_n^2 = 1$. Without loss of the generality, the UWB channel phase delay is assumed to be $\theta = 0$ in the analysis. Then the complex baseband signal in (4-2) can be rewritten as:

$$\tilde{\upsilon}_i = \sqrt{2E_s}\left(\sum_{n=0}^{L} f_n \exp(j\theta_{i-n})\right) + \tilde{\eta}_i \tag{4-8}$$

Where E_s denotes the energy per data symbol, and $\tilde{\eta}_i$ is the complex Gaussian noise with one-sided power spectrum density $2N_0$.

To obtain the upper bound on the average bit error probability $P_b(E)$ of the MLDSE scheme on the differential MPSK signals, we use an approach analogous to that used in the derivation of the MSDD performance of MPSK signals. The upper bound on $P_b(E)$ of N_s symbols maximum likelihood differential sequence estimation is the sum of all pairwise error probabilities associated with it. Denotes $\Delta\boldsymbol{\theta} = \left(\Delta\theta_1, \Delta\theta_2, \cdots, \Delta\theta_{N_s-1}, \Delta\theta_{N_s}\right)$ is the sequence of N_s transmitted information phases, and $\Delta\hat{\boldsymbol{\theta}} = \left(\Delta\hat{\theta}_1, \Delta\hat{\theta}_2, \cdots, \Delta\hat{\theta}_{N_s-1}, \Delta\hat{\theta}_{N_s}\right)$ is the corresponding sequence of detected phases. Without loss of the generality, the initial reference phase for the differential encoder is assumed to be $\theta_i = 0, \quad i \le 0$, then we have $\theta_k = \sum_{i=1}^{k} \Delta\theta_i$.

Let Z be the decision variable of $\Delta\boldsymbol{\theta}$,

$$
\begin{aligned}
Z &= \sum_{i=1}^{N_s} \tilde{\upsilon}_i \left(\sum_{n=0}^{L} f_n \exp\left(-j\theta_{i-n}\right) \right) \\
&= \sqrt{2E_s} \sum_{i=1}^{N_s} \sum_{n=0}^{L} \left\{ f_n^2 \exp(j\theta_{i-n}) \exp(-j\theta_{i-n}) \right\} + \sum_{i=1}^{N_s} \left\{ \tilde{\eta}_i \sum_{n=0}^{L} f_n \exp(-j\theta_{i-n}) \right\} \\
&= N_s \sqrt{2E_s} + \sum_{i=1}^{N_s} \left\{ \tilde{\eta}_i \sum_{n=0}^{L} f_n \exp\left[-j \left(\sum_{k=1}^{i-n} \Delta\theta_k \right) \right] \right\}
\end{aligned}
\tag{4-9}
$$

The first and second items in (4-9) represent the signal and noise components in the decision variable Z, respectively.

Let $\hat{Z}$ be the decision variable of $\Delta\hat{\boldsymbol{\theta}}$,

$$
\begin{aligned}
\hat{Z} &= \sum_{i=1}^{N_s} \tilde{\upsilon}_i \left(\sum_{n=0}^{L} f_n \exp\left(-j\hat{\theta}_{i-n}\right) \right) \\
&= \sqrt{2E_s} \sum_{i=1}^{N_s} \sum_{n=0}^{L} \left\{ f_n^2 \exp(j\theta_{i-n}) \exp(-j\hat{\theta}_{i-n}) \right\} + \sum_{i=1}^{N_s} \left\{ \tilde{\eta}_i \sum_{n=0}^{L} f_n \exp(-j\hat{\theta}_{i-n}) \right\} \\
&= \delta \sqrt{2E_s} + \sum_{i=1}^{N_s} \left\{ \tilde{\eta}_i \sum_{n=0}^{L} f_n \exp\left[-j \left(\sum_{k=1}^{i-n} \Delta\hat{\theta}_k \right) \right] \right\}
\end{aligned}
\tag{4-10}
$$

where $\delta \triangleq \sum_{i=1}^{N_s} \exp\left\{ j \sum_{k=1}^{i-n} \left(\Delta\theta_k - \Delta\hat{\theta}_k \right) \right\}$. The first and second items in (4-10) represent the signal and noise components in the decision variable $\hat{Z}$, respectively.

Let $\boldsymbol{u}$ be the sequence of N_s information symbols that produces $\Delta\boldsymbol{\theta}$ at the transmitter and $\hat{\boldsymbol{u}}$ be the sequence resulted from the detection of $\Delta\hat{\boldsymbol{\theta}}$, then the bit error probability has the upper bound:

$$
P_b(E) \leq \frac{1}{N_s \log_2 M} \sum_{\Delta\boldsymbol{\theta} \neq \Delta\hat{\boldsymbol{\theta}}} w(\boldsymbol{u}, \hat{\boldsymbol{u}}) \cdot \Pr\left\{ \hat{Z} > Z \middle| \Delta\boldsymbol{\theta} \right\}
\tag{4-11}
$$

where $w(\boldsymbol{u},\hat{\boldsymbol{u}})$ denotes the Hamming distance between $\boldsymbol{u}$ and $\hat{\boldsymbol{u}}$ and $\Pr\left\{\hat{Z} > Z \middle| \Delta\boldsymbol{\theta}\right\}$ denotes the pairwise probability that $\Delta\hat{\boldsymbol{\theta}}$ is incorrectly chosen when indeed $\Delta\boldsymbol{\theta}$ was sent. The evaluation of the pairwise probability $\Pr\left\{\hat{Z} > Z \middle| \Delta\boldsymbol{\theta}\right\}$ is detailed in the Appendix A. The result of the evaluation is given as:

$$\Pr\left\{\hat{Z} > Z \middle| \Delta\boldsymbol{\theta}\right\} = \frac{1}{2}\left[1 - Q\left(\sqrt{b},\sqrt{a}\right) + Q\left(\sqrt{a},\sqrt{b}\right)\right] \tag{4-12}$$

where $Q(\alpha,\beta)$ is referred as the Marcum's Q-function and is given by

$$Q(\alpha,\beta) \triangleq \int_{\beta}^{\infty} x \exp\left\{-\frac{x^2+a^2}{2}\right\} I_0(ax)\,dx \tag{4-13}$$

The arguments in (4-12) are given as

$$\begin{Bmatrix} b \\ a \end{Bmatrix} = \frac{E_s}{2N_0}\left[N_s \pm \sqrt{N_s^2 - |\delta|^2}\right] \tag{4-14}$$

From the evaluation, we find the performance of the maximum likelihood differential sequence estimation scheme in the ISI channel is same as the performance of the multiple symbols differential detection scheme without the ISI. Thus we conclude that the inter-symbol interference (ISI) will not cause any loss in SNR in the proposed maximum likelihood differential sequence estimation (MLDSE) scheme for the differential M-PSK signals. While N_s gets large, we can expect the performance of the maximum likelihood differential sequence estimation approaches that of ideal coherent detection for the differential *M*-PSK signals.

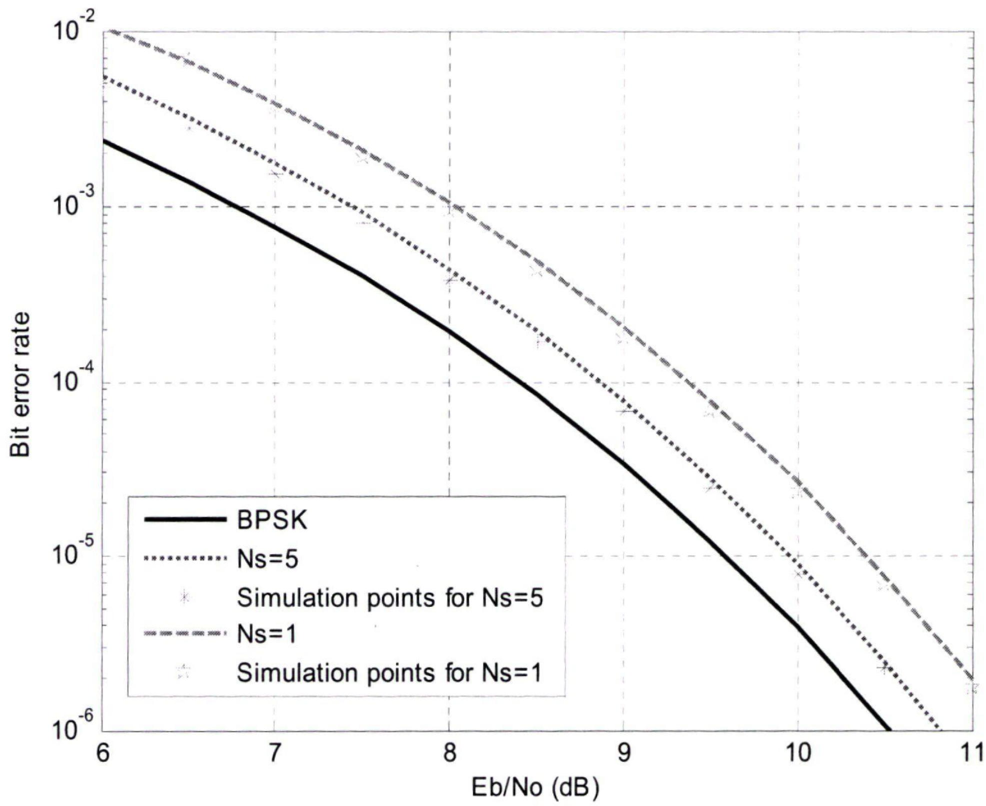

Figure 4-8: The upper BER bounds of MLDSE algorithm for DPSK signal

Figure 4-8 illustrates the upper bit error rate bounds of the MLDSE detection technique for the DPSK signals. The channel used in the figure is a two-path channel with $f_0 = 1/\sqrt{2}$, $f_1 = -1/\sqrt{2}$. The sequence length are given as $N_s = 1$ and $N_s = 5$. The case of $N_s = 1$ is actually the classical differential detection. The case of $N_s = 5$ approaches the performance limit of the coherent detection of BPSK signals with differential encoding. As the benchmark, the bit error rate of the coherent detection of BPSK signals without differential encoding in the AWGN channel is also shown in the figure. The MLDSE simulation points are obtained by the modified Viterbi algorithm via the Monte Carlo simulation. The simulation points prove that the upper bit error rate bounds of the MLDSE scheme are quite tight to reflect the real performance.

4.5 The Drawbacks of Rake Receiver in DS-UWB Scheme

Rake receiver is widely applied in existing systems such as CDMA and WCDMA to exploit the multipath diversity. The implementation issues of Rake receiver in the IEEE 802.15.3a DS-UWB scheme include: I) The Rake receiver only perform the diversity reception to a part of multipath energy in the DS-UWB scheme which means that the inter-chip interference (ICI) components in the multipath channel can be collected by the Rake receiver and the inter-symbol interference (ISI) components in the multipath channel are treated as an addition noise by the Rake receiver, and II) The non-ideal autocorrelation property of the spreading codes will cause some degradation in the *signal-to-noise ratio* (SNR) during the the decoding process of the spreading codes .

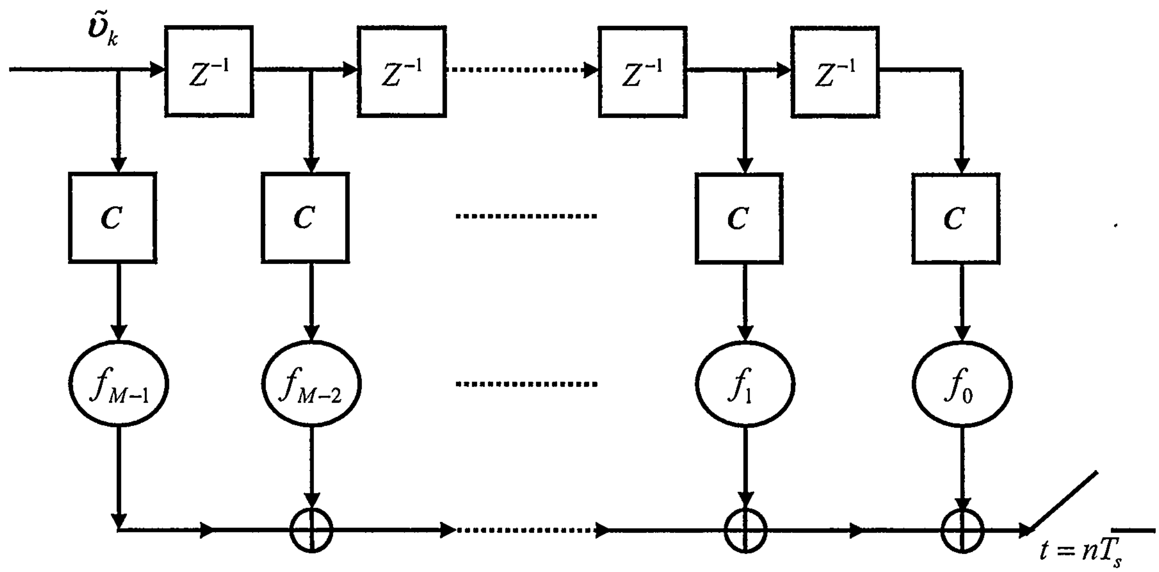

Figure 4-9: The discrete-time Rake receiver for the DS-UWB scheme

Figure 4-9 illustrates the structure of an equivalent discrete-time Rake receiver used to collect the multipath energy in the received chip samples for the DS-UWB system. The Rake receiver is assumed to an ideal Rake receiver where all inter-chip interference (ICI) components are ideally estimated and all inter-symbol interference (ISI) components are treated as an additive noise by the Rake receiver. The *maximal ratio combining* (MRC) technology is used in the ideal Rake receiver scheme to combine the multipath energy. In Figure 4-9, $\boldsymbol{C}$ represents a FIR filter with coefficients as the spreading code $\left[c_{N_c}, c_{N_c-1}, \cdots, c_2, c_1\right]$. The equivalent discrete UWB channel dispersion coefficients which samples in chip rate can be expressed as $\{f_0, f_1, f_2, \cdots, f_L\}$. In the classical Rake receiver the ICI components are resolved and collected. The ISI components are treated as an additive noise. Therefore when a short spreading code is used in the DS-UWB scheme ($N_c < L$), the Rake receiver suffers the diversity gain decrease since only a part of multipath energy can be collected. The energy loss due to a limited collection by the Rake receiver is defined as:

$$L \triangleq 10\log\left(\frac{\sum_{i=0}^{N_c-1}|f_i|^2}{\sum_{i=0}^{L}|f_i|^2}\right) \tag{4-15}$$

Table 4-2 gives the energy loss (dB) in the UWB channel models provided by the IEEE802.15.3a when the chip frequency is 1320MHz and the code length N_c is 6, 12, and 24 respectively. The values are a statistical mean of over 100 channel generations for each channel model.

Loss(dB)	CM1	CM2	CM3	CM4
M=6	-5.4	-11.4	-12.8	-16.5
M=12	-3.5	-6.4	-10.0	-14.8
M=24	-2.3	-2.7	-7.9	-13.4

Table 4-2: Diversity gain loss due to the partial energy collection

On the other hand, we know Rake receiver exploits the multipath diversity by using the auto-correlation property of the spread code. When the auto-correlation property is not ideal, it may cause performance degradation to the received signals. For example, IEEE802.15.3a DS-UWB scheme provides some ternary sequences as the spreading codes. The ternary sequences have relatively good auto-correlation property at a short length [78]. One of length $M = 12$ given ternary code in the IEEE802.15.3a DS-UWB scheme is $[0\ \text{-1}\ \text{-1}\ \text{-1}\ 1\ 1\ 1\ \text{-1}\ 1\ 1\ \text{-1}\ 1]$.

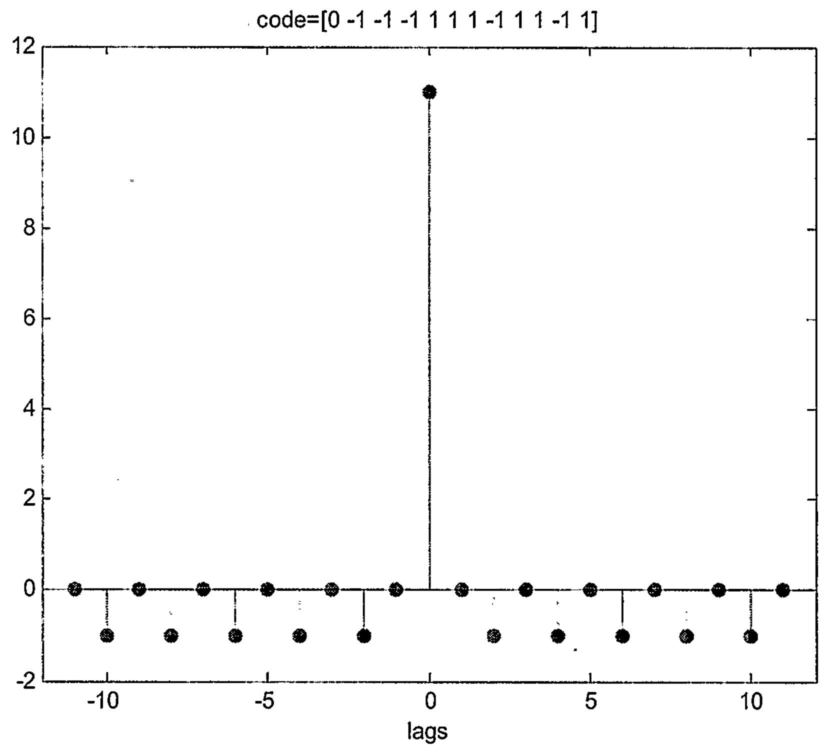

Figure 4-10: The auto-correlation property of a M=12 ternary code

Figure 4-10 gives the auto-correlation property of the spreading code $[0\ \text{-1}\ \text{-1}\ \text{-1}\ 1\ 1\ 1\ \text{-1}\ 1\ 1\ \text{-1}\ 1]$. When lag is zero, the auto-correlation has the largest value 11. When the lags are the odd numbers (1, 3, 5, 7, 9, 11), the auto-correlation coefficients are zero. When the lags are the even numbers (2, 4, 6, 8, 10), the auto-correlation coefficients are -1. The auto-correlation coefficients on the even number lags will cause a self interference in the despreading process. Even without the additive noise, the despreading process will generate a $10\log(1/11) = -10.4dB$ noise by itself due to the non-ideal autocorrelation property of the spread code. When the spread code length is large, the negative impact of non-ideal autocorrelation property will be a limiting factor to the detection performance and can be neglected in low SNR cases. But the degradation becomes more severe with the decrease of the spread code length. Actually the Rake receiver is not suitable to IEEE 802.15.3a DS-UWB scheme when the spread code length M is 2, 3 or 6, since the autocorrelation property of these spread codes are far away from the ideal case.

Therefore, we can expect the Rake receiver will have the poor performance in high speed transmission (several hundred Megabits per second) for the DS-UWB scheme. The proposed maximum likelihood differential sequence estimation (MLDSE) reception scheme can be an excellent alternative to the Rake receiver for the IEEE 802.15.3a DS-UWB scheme.

4.6 Summary

The chapter introduces the proposed maximum likelihood differential sequence estimation (MLDSE) receiver for the IEEE 802.15.3a DS-UWB scheme as an alternative to the classical Rake receiver. The classical Rake receiver has poor performance in the DS-UWB scheme due to the partial energy collection (suffering ISI) and the non-ideal autocorrelation property of the spreading codes.

The proposed MLDSE receiver implements the maximum likelihood sequence estimation (MLSE) equalization and multiple symbols differential detection (MSDD) reception at the same time with a modified Viterbi algorithm. A new decision rule for the modified Viterbi algorithm is proposed to enable the Viterbi algorithm to perform the two functions at the same time. The branch metrics and path metrics computations of new Viterbi algorithm are changed according to the new decision rule respectively. The MLDSE reception scheme possesses the both advantages of MSDD reception and MLSE equalization. I) The MLDSE algorithm don't need the carrier phase recovery circuit, which can simplify the receiver RF end design and more computations are done in the baseband to exploit the rapid development of the DSP. II) Similar to MLSE equalization, the MLDSE algorithm can also untangle the ISI components in the received signals and combine all these ISI components to improve the SNR. The performance upper-bound analysis demonstrates that there is no loss in performance caused by inter-symbol interference when the channel is ideally estimated in the MLDSE reception scheme. Therefore the performance of the MLDSE algorithm approaches the limit of the coherent detection of M-PSK signal with differential encoding. The limit is also the performance limit of the multiple symbols differential detection (MSDD) reception.

Comparing the Rake receiver in the DS-UWB scheme, the higher performance of the proposed MLDSE algorithm comes as the expense of more complexity to the receiver. The increased complexities include:

I) More multipath channel response parameters need be estimated in the MLDSE scheme. In the proposed MLDSE scheme, the digital signal processing is implemented in the chip rate of DS-UWB. The received RF signals are down-converted to the baseband signals and sampled in the chip rate before sending these samples to the MLDSE processing unit. The proposed MLDSE scheme needs to estimate the whole discrete-time chip-interval channel impulse response which defines as $L+1$ points (L is the channel memory). For the classical Rake receiver, only initial N_c points (N_c is the length of spreading code) in the channel impulse response need to be estimated. For the severe ISI channel, we have $(L+1) \gg N_c$. For instance, in the CM4 channel of the IEEE802.15.3a DS-UWB scheme the channel memory L is larger than 120 to collect 99% multipath energy. That means there are over 120 channel points need be estimated in the MLDSE algorithm. Compared to the MLDSE algorithm, the classical Rake receiver for the IEEE802.15.3a DS-UWB scheme only need estimate 12 channel points when the spreading code length is 12. Therefore, the MLDSE scheme achieves the higher performance at a price of the more computation capacity needed to estimate more channel impulse response points. To reduce the complexity of the DS-UWB channel estimation, a joint data and channel estimation strategy using the adaptive algorithms is also proposed in the chapter for the MLDSE scheme.

II) More computational capacities are needed to implement the modified Vertibi algorithm in the MLDSE scheme. As a generality, the proposed MLDSE scheme needs more computational capacity to untangle the ISI components and provide higher transmission performance than the Rake receiver. Similar to the Vertibi algorithm, the modified Vertibi algorithm used in the MLDSE scheme need huge computational capacities. When the MLDSE scheme is deployed with the *M*-PSK signals and channel memory L, all M^{L+1} possible branch metrics and path metrics will be computed in each stage. For example, if $M=4$ and $L=120$, each stage need calculate all $4^{121} \approx 7.067 \times 10^{72}$ possible branch metrics and path metrics in the Vertibi algorithm, which is far ahead of the achievement of current DSP technology. But in the DS-UWB communications, the direct sequence encoding is a special case of the repeating encoding. The decoding of the direct sequence encoding can be incorporated into the MLDSE algoirhm. The computational capacities needed by the MLDSE algorithm can be reduced greatly in the DS-UWB communications. When the MLDSE scheme for the DS-UWB communications is deployed with the *M*-PSK signals, channel memory L and the direct sequence length N_c, there are total M^{L/N_c+1} possible branch metrics and path metrics needed to compute in each stage. For example, if $M=4$, $N_c=12$ and $L=120$, each stage need calculate all $M^{L/N_c+1} = 4^{11} \approx 4.194 \times 10^6$ possible branch metrics and path metrics in the MLDSE scheme, which is still achievable by the current DSP technology. Therefore, the computational capacity needed by the MLDSE implementation is greatly decreased in the IEEE802.15.3a DS-UWB scheme.

Chapter 5. Simulations and Comparisons

In this chapter through the computer simulations, we verify the proposed maximum likelihood differential sequence estimation (MLDSE) receiver scheme in the UWB channel models provided by the IEEE802.15.3a channel committee. The results obtained from the computer simulations are depicted and discussed. The simulations compare the error rate performance of the MLDSE detection technique with the ideal Rake receiver in the CM1, CM2, CM3 and CM4 channels, respectively. The MLDSE receiver is implemented by the modified *Viterbi algorithm* (VA). The MLDSE receiver is first invoked with perfect channel knowledge, the MLDSE receiver with ideal channel knowledge is referred as ideal MLDSE receiver in the simulations. Then with an adaptive joint data and channel estimation approach, the MLDSE receiver with an adaptive joint data and channel estimation is referred as adaptive MLDSE receiver in the simulations. The recursive least squares (RLS) algorithm for the first block of data is used to ensure rapid tap convergence. Thereafter tThe least mean square (LMS) algorithm is used to ensure rapid execution speed. In the simulations, the ideal Rake receiver collects all multipath energy of the ICI components through the MRC combiner with perfect channel knowledge. The ISI components will be treated as an additive noise by the ideal Rake receiver. The spreading code used in the simulation is the ternary sequence of $[0\ -1\ -1\ -1\ 1\ 1\ 1\ -1\ 1\ 1\ -1\ 1]$. The error rate performance of the ideal MLDSE receiver, the adaptive MLDSE receiver and the Rake receiver are obtained via Monte Carlo simulation, and fit a curve to the simulated BER points.

The main parameters used in the simulation are listed in Table 5-1.

Parameter	**Characteristics**
Chip data rate	1320Mc/s
Channel Model	CM1,CM2,CM3,CM4
Spreading Code Length	M=12
Data Modulation scheme	DPSK
Trace Back Length	32
Adaptive Algorithm	RLS,LMS
RLS Forgetting Factor	0.9999
LMS step size	0.0001

Table 5-1: General parameters used in simulations

5.1 The UWB channel responses used in the simulation

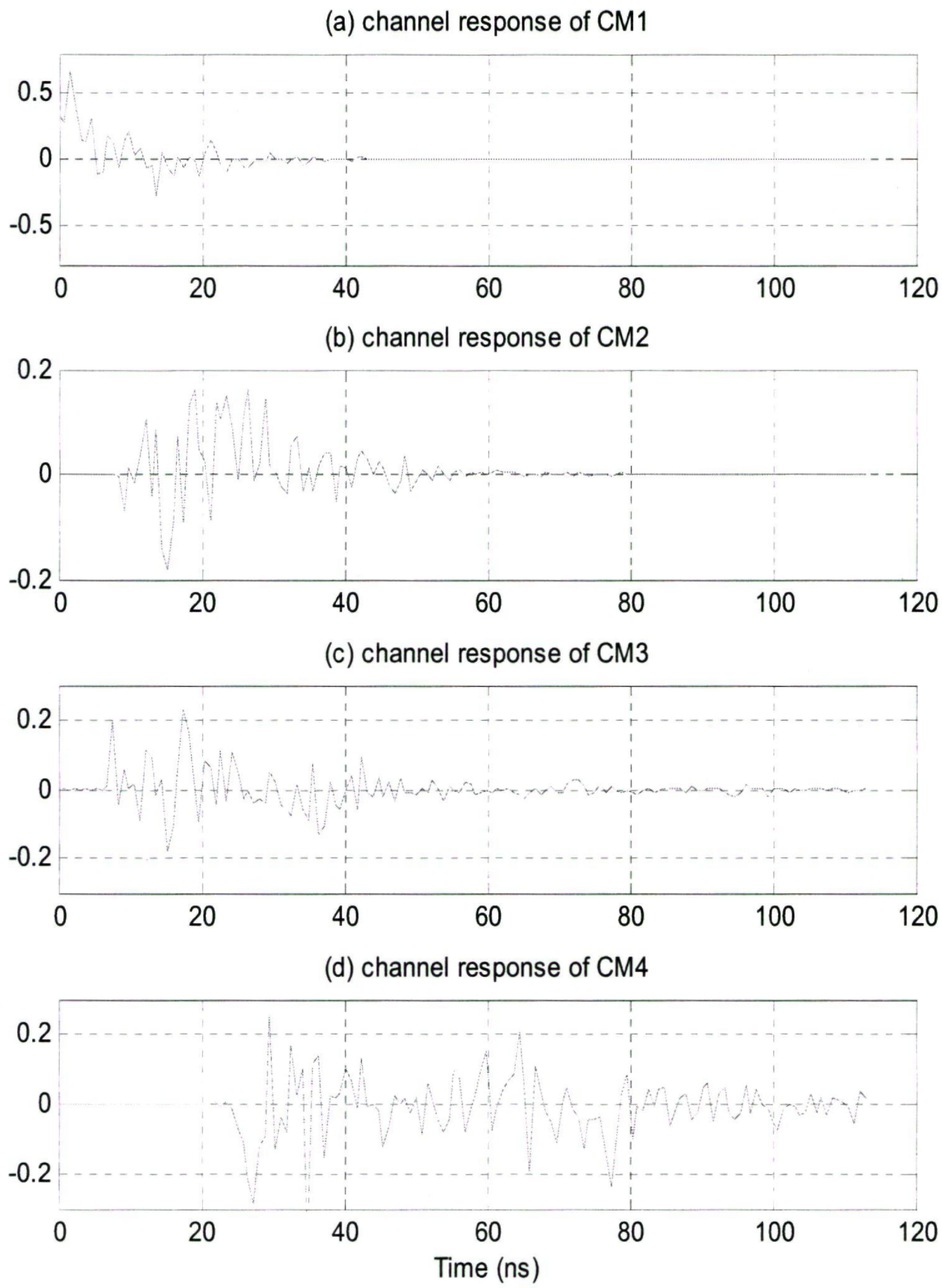

Figure 5-1: The channel responses of the four UWB channel models

Figure 5-1 illustrates the UWB channel responses of IEEE 802.15.3a channel models. CM1 is defined as the channel model where the line of sight (LOS) exists and transmission distance is within 0 to 4 meters. CM2 is denoted as the channel model where the line of sight (LOS) doesn't exist and transmission distance is within 0 to 4 meters. CM3 is defined as the channel model where the line of sight (LOS) doesn't exist and transmission distance is within 4 to 10 meters. CM4 is denoted as the channel model which is extreme non-line of sight channel and transmission distance is within 4 to 10 meters. We can find from the channel responses that the UWB channels are typical multipath channels. The path delay spreads increase and the multipath energies become more dispersive with the increase of transmission distance and without the line of sight.

5.2 The performance of the MLDSE scheme simulation

5.2.1 *The error rate of the MLDSE scheme in the CM1 channel*

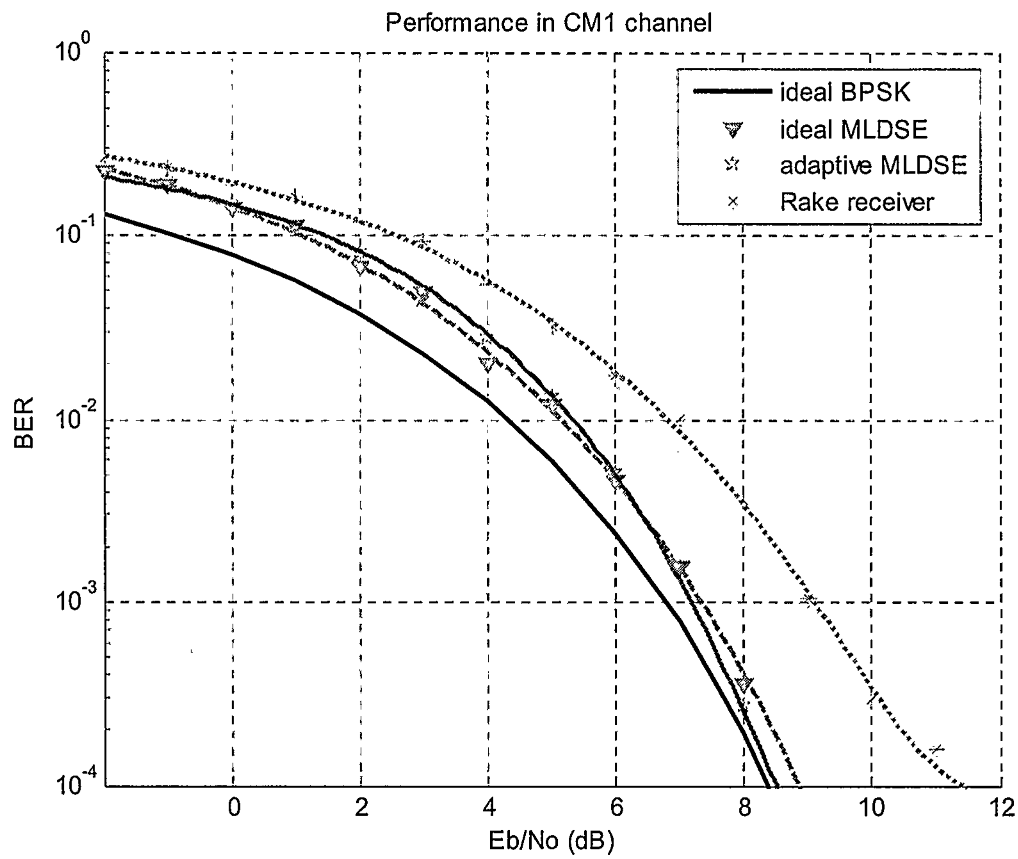

Figure 5-2: The performance of the MLDSE scheme in CM1 channel

Figure 5-2 illustrates a comparison of the error rate performance of the MLDSE detection technique with the ideal Rake receiver in the CM1 channel. CM1 channel is the transmission channel of $0 \sim 4m$ range with line of sight. The results of adaptive MLDSE receiver align fairly close with the ideal MLDSE receiver results. The error rate of the MLDSE receiver is fairly close to the performance limit of the BPSK with differential encoding. The MLDSE receiver outperforms the ideal Rake receiver about 2dB in the bit error rate of 10^{-4}. As a benchmark, the error rate for the BPSK signal for the AWGN channel is also shown in the figure.

5.2.2 *The error rate of the MLDSE scheme in the CM2 channel*

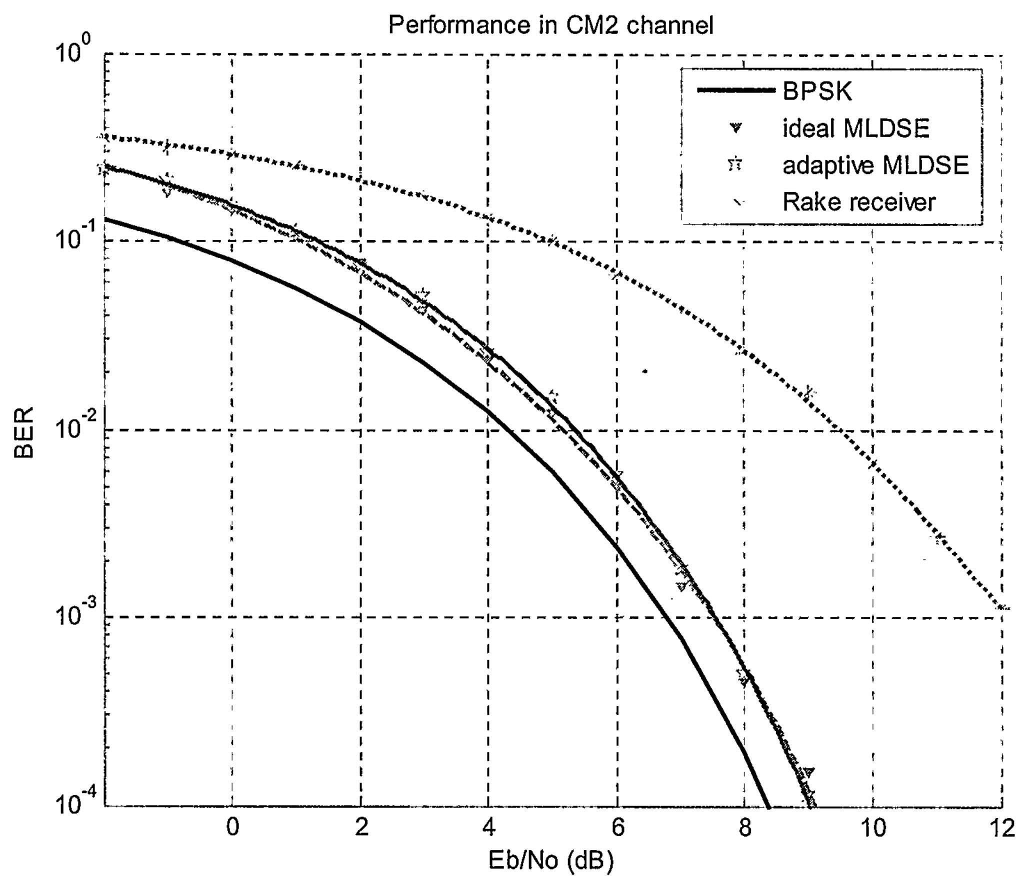

Figure 5-3: The performance of the MLDSE scheme in CM2 channel

Figure 5-3 illustrates a comparison of the error rate performance of the MLDSE detection technique with the ideal Rake receiver in the CM2 channel. CM2 channel is the transmission channel of $0 \sim 4m$ range without line of sight. The bit error rate of adaptive MLDSE receiver aligns fairly close with that of the MLDSE receiver with ideal channel knowledge. The performance of the MLDSE receiver in CM2 channel is same as the performance in CM1 channel. The bit error rate of the Rake receiver in CM2 channel degrades greatly comparing that of CM1 channel. The MLDSE receiver outperforms the Rake receiver about 4dB in the bit error rate of 10^{-2}. As a benchmark, the error rate for the BPSK signal for the AWGN channel is also shown in the figure.

5.2.3 *The error rate of the MLDSE scheme in the CM3 channel*

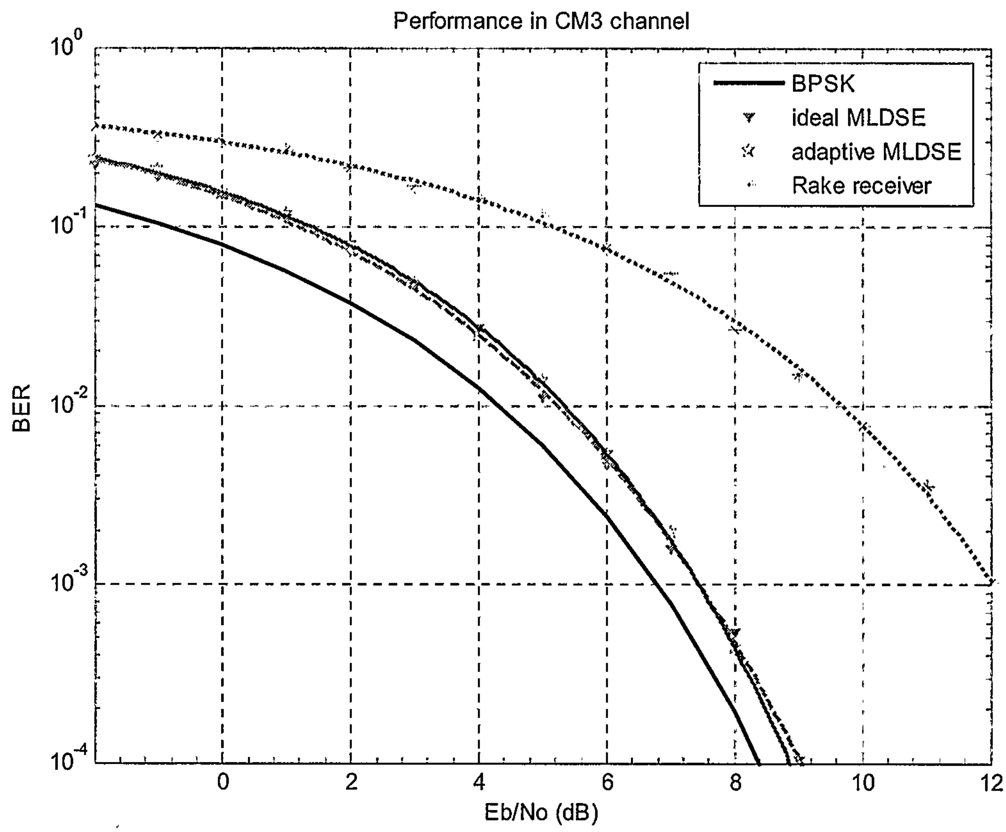

Figure 5-4: The performance of the MLDSE scheme in CM3 channel

Figure 5-4 illustrates a comparison of the error rate performance of the MLDSE detection technique with the ideal Rake receiver in the CM3 channel. CM3 channel is the transmission channel of $4 \sim 10m$ range without line of sight. The bit error rate of adaptive MLDSE receiver aligns fairly close with that of the MLDSE receiver with ideal channel knowledge. The performance of the MLDSE receiver in CM3 channel is close to the performance of the coherent detection of BPSK signal with differential encoding. The bit error rate of the Rake receiver in CM3 channel greatly degrades comparing to that in CM1 channel. As a benchmark, the error rate for the BPSK signal for the AWGN channel is also shown in the figure.

5.2.4 *The error rate of the MLDSE scheme in the CM4 channel*

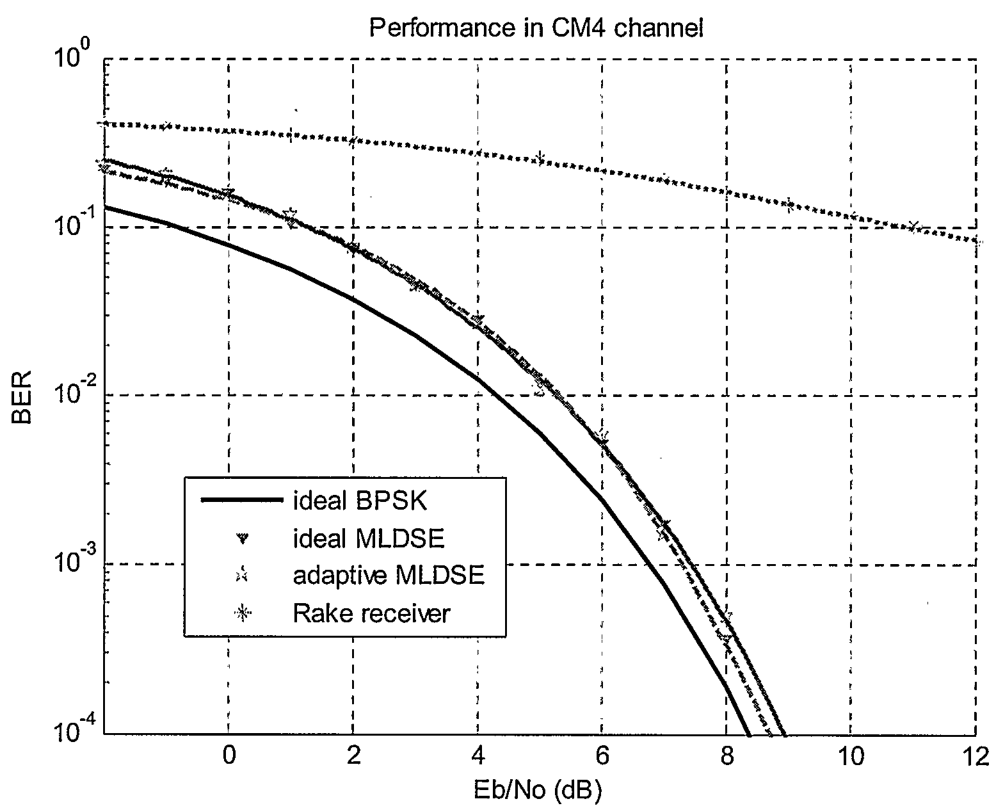

Figure 5-5: The performance of the MLDSE scheme in CM4 channel

Figure 5-5 illustrates a comparison of the error rate performance of the MLDSE detection technique with the ideal Rake receiver in the CM4 channel. CM4 channel is the transmission channel of $4 \sim 10m$ range with extreme non-line of sight. The bit error rate of adaptive MLDSE receiver aligns fairly close with that of the MLDSE receiver with ideal channel knowledge. The rigorous transmission channel has little impact on the performance of the MLDSE receiver, while the error rate of the Rake receiver is greatly degraded. The degradation of the Rake receiver is due to the factors of the partial energy collection and ISI. As a benchmark, the error rate for the BPSK signal for the AWGN channel is also shown in the figure.

5.3 Summary

In this chapter, the performances of the propose MLDSE reception scheme for DS-UWB systems are verified by the computer simulations. The IEEE 802.15.3a PHY parameters and channel models (CM1 to CM4) are applied to these simulations. From above simulations, we can conclude that the performance of the MLDSE receiver can approach the performance limit, which is also the limit of the multiple symbols differential detection (MSDD), of the coherent detection of the BPSK signals with differential encoding no matter in which channel model when the multipath channel is estimated ideally. It proves all ISI and ICI components in the DS-UWB system can be untangled and collected by the MLDSE algorithm at the expense of more complexity to the UWB channel estimation. The MLDSE reception performance with an adaptive joint data and channel estimation scheme is demonstrated to be very close to the performance limit of the MLDSE reception with ideal channel knowledge in these simulations. The proposed joint data and channel estimation approach is proved to be feasible by the computer simulations. As a comparison, the performances of the ideal Rake receiver are simulated in the same conditions for the DS-UWB system. With the increase of the transmission distance and without line of sight, the multipath energies of the UWB channel are more dispersive which means more energies are distributed in the ISI components. The ideal Rake receiver optimally collects all of ICI components and treats ISI components as an additive noise. Comparing to the MLDSE receiver, the simulations demonstrate the ideal Rake receiver suffers a severe performance loss in the dispersive channel models for the IEEE 802.15.3a DS-UWB system. As a result, the MLDSE receiver is proved to be superior to the Rake receiver in performance for the IEEE 802.15.3a DS-UWB scheme.

Chapter 6. Conclusions and Future works

6.1 Conclusions

Ultra-wideband (UWB) radio is a fast emerging technology with uniquely attractive features in the communication industry. UWB radio possesses the characteristics and capabilities that make it suitable for short-range high-speed wireless communication. The Direct Sequence Ultra-wideband (DS-UWB) system uses the direct sequence spread spectrum (DSSS) technology that is already deployed in other wireless systems to transmit the information bits over several GHz bandwidths, which offers great flexibility and promising ability for future high data rate wireless access. The DS-UWB technology is a high potential candidate standard for the IEEE 802.15.3a Task Group (TG3a). The purpose of this standard is to provide a specification for a low complexity, low-cost, low-power consumption, and high-data-rate wireless connectivity among devices within the personal operating space.

The indoor UWB channel has severe inter-symbol interference (ISI) and inter-chip interference (ICI) for the DS-UWB signals. The classical Rake receiver used in the DSSS systems is able to achieve the diversity reception of the ICI components and the ISI components are treated as an additive noise. Thereby the Rake receiver suffers severe performance degradations due to abundant ISI energies exist in the DS-UWB communications. The main purpose of the thesis is to propose and investigate an improved and feasible receiver scheme as an alternative to the Rake receiver for the DS-UWB system to achieve high performance and simple structure.

The objectives of the thesis are to:

- Introduce the basic knowledge of several classical signal detection technologies (the matched filer, differential detection and Rake receiver) and two advanced sequence detection technologies (MLSE and MSDD) in the wireless communications. Two kinds of sequence detection technology which are the foundation of the proposed MLDSE scheme, *maximum likelihood sequence estimation* (MLSE) for the inter-symbol interference (ISI) equalization and *multiple symbols differential detection* (MSDD) for compensating the performance loss of the classical differential detection, are introduced in the thesis. The proposed *maximal likelihood differential sequence estimation* (MLDSE) algorithm is to perform both functions of the MLSE equalization and MSDD reception at the same time via a modified *Viterbi algorithm* (VA). As expected, the proposed scheme owns the simplicity of the differential detection technique, which means that in the MLDSE receiver the carrier phase tracking circuit is not required and the receiver structure can be greatly simplified. On the other hand, the proposed scheme can achieve the optimal performance in the ISI and additive Gaussian noise channel. Similar to the MLSE equalization, the MLDSE scheme collects all the multipath energy via untangling the ISI components in the received signal. There is no performance loss caused by the ISI components in the proposed

MLDSE reception scheme. Therefore the proposed algorithm can approach the performance limit of the MSDD reception.

- Introduce the proposed MLDSE algorithm aimed to the IEEE802.15.3a DS-UWB scheme. The proposed MLDSE scheme performs the MLSE equalization and MSDD reception via a modified *Viterbi algorithm* (VA) without increasing additional hardware to the receiver. Viterbi algorithm is a useful and simple method to iteratively find the most possible path in the maximum likelihood sequence estimation. To implement two functions of the MLSE equalization and MSDD reception at the same time, the decision rule of the Viterbi algorithm is modified in the MLDSE receiver. As a consequence, the formulations of the branch metrics and path metrics are modified respectively. The performance bound of the MLDSE scheme is derived theoretically for the *M*-DPSK signals in the thesis using the analogous approach in the derivation of the MSDD performance bound. The derivation proves the ISI components won't result in a performance loss in the MLDSE scheme if the channel is estimated ideally. The derivation also demonstrates the performance limit of the MLDSE scheme approaches the coherent detection of the *M*-DPSK signals with differential encoding when the observation is enough long. To reduce the complexity of the DS-UWB channel estimation, a joint data and channel estimation strategy using the adaptive algorithms is also proposed in the thesis for the MLDSE scheme.
- Demonstrate the superiority of the proposed MLDSE scheme to the Classical Rake receiver in a DS-UWB system. The performances of the propose

MLDSE reception scheme for DS-UWB systems are verified by the computer simulations using the IEEE 802.15.3a channel parameters and channel models (CM1 to CM4). The feasibility of the adaptive joint data and channel estimation strategy is also verified by the computer simulations. The simulations demonstrate that the performance of the MLDSE receiver can approach the performance limit of the multiple symbols differential detection (MSDD) reception no matter in which channel model when the multipath channel is estimated ideally. It proves all ISI and ICI components in the DS-UWB system can be untangled and collected by the MLDSE algorithm at the expense of more complexity to the UWB channel estimation. The MLDSE reception performance with the joint data and channel estimation scheme using adaptive algorithms is demonstrated to be very close to that of the MLDSE reception with ideal channel knowledge in these simulations. As a comparison, the performances of the classical Rake receiver are simulated with the full knowledge of channel responses for the DS-UWB system. The simulations demonstrate the classical Rake receiver suffers a severe performance loss in the dispersive channel models for the IEEE 802.15.3a DS-UWB system. As a result, the MLDSE receiver is proved to be superior to the classical Rake receiver in performance for the IEEE 802.15.3a DS-UWB scheme.

Comparing to the classical Rake receiver in the DS-UWB scheme, the higher performance of the proposed MLDSE algorithm comes as the expense of more complexity to the receiver. . The increased complexities include:

- More multipath channel response parameters need be estimated in the MLDSE scheme. In the proposed MLDSE scheme, the digital signal processing is implemented in the chip rate of DS-UWB. The MLDSE scheme needs to estimate the whole discrete-time chip-interval channel impulse response which defines as $L+1$ points (L is the channel memory). For the classical Rake receiver, only initial N_c points (N_c is the length of spreading code) in the channel impulse response need to be estimated. For the severe ISI channel, we have $(L+1) \gg N_c$. For instance, in the CM4 channel of the IEEE802.15.3a DS-UWB scheme the channel memory L is larger than 120 to collect 99% multipath energy. That means there are over 120 channel points need be estimated in the MLDSE algorithm. Compared to the MLDSE algorithm, the classical Rake receiver for the IEEE802.15.3a DS-UWB scheme only need estimate 12 channel points when the spreading code length is 12. Therefore, the MLDSE scheme achieves the higher performance at a price of the more computation capacity needed to estimate more channel impulse response points.
- More computational capacities are needed to implement the modified Vertibi algorithm in the MLDSE scheme. As a generality, the proposed MLDSE scheme needs more computational capacity to untangle the ISI components and provide higher transmission performance than the Rake receiver. Similar

to the Vertibi algorithm, the modified Vertibi algorithm used in the MLDSE scheme need huge computational capacities. When the MLDSE scheme is deployed with the *M*-PSK signals and channel memory L, all M^{L+1} possible branch metrics and path metrics will be computed in each stage. For example, if $M=4$ and $L=120$, each stage need calculate all $4^{121} \approx 7.067 \times 10^{72}$ possible branch metrics and path metrics in the Vertibi algorithm, which is far ahead of the achievement of current DSP technology. But in the DS-UWB communications, the direct sequence encoding is a special case of the repeating encoding. The decoding of the direct sequence encoding can be incorporated into the MLDSE algoirhm. The computational capacities needed by the MLDSE algorithm can be reduced greatly in the DS-UWB communications. When the MLDSE scheme for the DS-UWB communications is deployed with the *M*-PSK signals, channel memory L and the direct sequence length N_c, there are total M^{L/N_c+1} possible branch metrics and path metrics needed to compute in each stage. For example, if $M=4$, $N_c=12$ and $L=120$, each stage need calculate all $M^{L/N_c+1}=4^{11} \approx 4.194 \times 10^{6}$ possible branch metrics and path metrics in the MLDSE scheme, which is still achievable by the current DSP technology. Therefore, the computational capacity needed by the MLDSE implementation is greatly decreased in the IEEE802.15.3a DS-UWB scheme.

The main contributions of the thesis include:

- The thesis proposes a novel DS-UWB receiver scheme referred as the *maximum likelihood differential sequence estimation* (MLDSE) reception scheme. The MLDSE algorithm can perform the MLSE equalization and MSDD reception functions in one algorithm. The performance bound of the proposed MLDSE scheme is derived theoretically in the thesis.
- The thesis proposes a modified Viterbi algorithm to iteratively implement the MLDSE algorithm. The decision rule of the Viterbi algorithm is modified to enable the MLDSE algorithm to perform the MLSE equalization and MSDD reception functions at the same time.
- The thesis proposes a joint data and channel estimation strategy using adaptive algorithms for the MLDSE scheme. The joint estimation strategy greatly simplifies the estimation process of the UWB multipath channel impulse response. The feasibility of the joint estimation strategy is verified via the computer simulations.
- The thesis demonstrates the superiority of the proposed MLDSE receiver to the ideal Rake receiver for the IEEE 802.15.3a DS-UWB scheme. Using the IEEE802.15.3a UWB channel models and PHY parameters of the DS-UWB scheme, the performances of the MLDSE receiver and ideal Rake receiver in the DS-UWB scheme are compared via the computer simulations.

6.2 Future works

The work presented in the thesis may be extended in the following areas:

- The research on the spreading code and chip boundary synchronization scheme. The implication in the proposed MLDSE receiver is that the spreading code and chip boundary synchronization have achieved. It is necessary to further research the spreading code and chip boundary synchronization scheme for the proposed receiver.
- The further research on the joint data and channel estimation. The joint data and channel estimation strategy will work in the high SNR situations. In the low SNR cases where the error rate of the data is high, the issue of the error feedback can not be neglected. A modification to the proposed joint estimation architecture need be researched further.
- The complexity reduction analysis to the implementations of the MLDSE processing schemes. The compromise of the MLDSE performance and complexity need be researched further.

Appendix A: Derivation of Pairwise Probability

In (4-9) and (4-10), it is shown that the decision variables Z and $\hat{Z}$ are complex Gaussian random variables with identical variances and arbitrary means and covariance, the pairwise probability of error $\Pr\left\{\left|\hat{Z}\right|>\left|Z\right|\middle|\Delta\boldsymbol{\theta}\right\}$ is given by [79]:

$$\Pr\left\{\left|\hat{Z}\right|>\left|Z\right|\middle|\Delta\boldsymbol{\theta}\right\}=\frac{1}{2}\left[1-Q\left(\sqrt{b},\sqrt{a}\right)+Q\left(\sqrt{a},\sqrt{b}\right)\right] \qquad \text{(A-1)}$$

Where $Q(\alpha,\beta)$ is Marcum's Q-function and

$$\begin{Bmatrix} b \\ a \end{Bmatrix}=\frac{1}{2\Omega}\left\{\frac{S+\hat{S}-2|\rho|\sqrt{S\hat{S}}\cos\left(\theta-\hat{\theta}+\phi\right)}{1-|\rho|^2}\pm\frac{S-\hat{S}}{\sqrt{1-|\rho|^2}}\right\} \qquad \text{(A-2)}$$

With $S\triangleq\frac{1}{2}\left|\overline{Z}\right|^2$, $\hat{S}\triangleq\frac{1}{2}\left|\overline{\hat{Z}}\right|^2$, $\Omega\triangleq\frac{1}{2}\overline{\left|S-\overline{S}\right|^2}=\frac{1}{2}\overline{\left|\hat{S}-\overline{\hat{S}}\right|^2}$, $\rho\triangleq\frac{1}{2\Omega}\overline{\left(S-\overline{S}\right)^*\left(\hat{S}-\overline{\hat{S}}\right)}$,

$\phi=\arg\{\rho\}$, $\theta\triangleq\arg\left\{\overline{S}\right\}$, $\hat{\theta}=\arg\left\{\overline{\hat{S}}\right\}$.

From (4-9) and (4-10), we get

$$\overline{Z}=\sqrt{2E_s}\sum_{i=1}^{N_s}\sum_{n=0}^{L}\left\{f_n^2\exp(j\theta_{i-n})\exp(-j\theta_{i-n})\right\}=N_s\sqrt{2E_s} \tag{A-3}$$

$$\overline{\hat{Z}}=\sqrt{2E_s}\sum_{i=1}^{N_s}\sum_{n=0}^{L}\left\{f_n^2\exp(j\theta_{i-n})\exp(-j\hat{\theta}_{i-n})\right\}=\delta\sqrt{2E_s} \tag{A-4}$$

Where $\delta\triangleq\sum_{i=1}^{N_s}\exp\left\{j\sum_{k=1}^{i-n}\left(\Delta\theta_k-\Delta\hat{\theta}_k\right)\right\}$. It is understood that the summation equals zero if the upper summation index is negative. Then we can get

$$S=E_sN_s^2,\quad \hat{S}=E_s\left|\delta\right|^2 \tag{A-5}$$

Also, we can get

$$\Omega=\frac{1}{2}E\left\{\left|\sum_{i=1}^{N_s}\eta_i\left(\sum_{n=0}^{L}f_n\exp\left(-j\theta_{i-n}\right)\right)\right|\right\}=N_sN_0 \tag{A-6}$$

And

$$\begin{aligned}\rho&=\frac{1}{2\Omega}\sum_{i=1}^{N_s}\sum_{m=1}^{N_s}E\left\{\eta_i\eta_m^*\right\}\left\{\left(\sum_{n=0}^{L}f_n\exp\left(j\theta_{i-n}\right)\right)\left(\sum_{k=0}^{L}f_k\exp\left(-j\hat{\theta}_{m-n}\right)\right)\right\}\\&=\frac{2N_0}{2\Omega}\sum_{i=1}^{N_s}\sum_{n=0}^{L}\left\{f_n^2\exp\left(j\theta_{i-n}-j\hat{\theta}_{i-n}\right)\right\}\\&=\frac{\delta}{N}\end{aligned} \tag{A-7}$$

Finally, substituting (A-5)-(A-7) into (A-2) gives the desired result:

$$\Pr\left\{\left|\hat{Z}\right| > |Z| \,\middle|\, \Delta\boldsymbol{\theta}\right\} = \frac{1}{2}\left[1 - Q\left(\sqrt{b}, \sqrt{a}\right) + Q\left(\sqrt{a}, \sqrt{b}\right)\right] \tag{A-8}$$

where

$$\begin{Bmatrix} b \\ a \end{Bmatrix} = \frac{E_s}{2N_0}\left[N_s \pm \sqrt{N_s^2 - |\delta|^2}\right] \tag{A-9}$$

Appendix B: Source Code of MLDSE function

```
/*
 * C-mex function.
 * Calling format: [sigout, storeMetric, storeState, storeInput]
 *     = mldse_eq(sigin,
 *                 chcffs,
 *                 const,
 *                 tbLen,
 *                 opmode,
 *                 nsamp,
 *                 preamble,
 *                 postamble,
 *                 lastProvided,  ("1" : initial TB memory is provided)
 *                 initialStateMetrics,     (length = numStates)
 *                 initialTracebackStates,  (length = tbLen*numStates)
 *                 initialTracebackInputs)  (length = tbLen*numStates)
 *
 */

#include "mex.h"
#include "math.h"
#include "Viterbi_acs_tbdec1.h"

enum {SIGIN_ARGC  =  0,        /* input signal                        */
    CHAN_COEFF_ARGC,        /* channel estimates                      */
    CONSTPTS_ARGC,          /* constellation points                   */
    TB_ARGC,                /* traceback length                       */
    OPMODE_ARGC,            /* operation mode                         */
    SAMP_PER_SYM_ARGC,      /* samples per input symbol               */
    PREAMBLE_ARGC,          /* preamble                               */
    POSTAMBLE_ARGC,         /* postamle                               */
    LAST_PROVIDED_ARGC,     /* indicator for provided traceback memory */
    LAST_METRIC_ARGC,       /* initial state metrics                  */
    LAST_STATE_ARGC,        /* initial traceback states               */
    LAST_INPUT_ARGC,        /* initial traceback inputs               */
    NUM_ARGS};

enum {SIGOUT_ARGC  = 0,    /* equalized signal                        */
    STORE_METRIC_ARGC,     /* final state metrics                     */
    STORE_STATE_ARGC,      /* final TB states of last tbLen trellis branches */
    STORE_INPUT_ARGC};     /* final TB inputs of last tbLen trellis branches */

enum {CONT=1, RST};

#define SIGIN_ARG           (prhs[SIGIN_ARGC])
#define CHAN_COEFF_ARG      (prhs[CHAN_COEFF_ARGC])
#define CONSTPTS_ARG        (prhs[CONSTPTS_ARGC])
#define TB_ARG              (prhs[TB_ARGC])
```

```
#define OPMODE_ARG          (prhs[OPMODE_ARGC])
#define SAMP_PER_SYM_ARG      (prhs[SAMP_PER_SYM_ARGC])
#define PREAMBLE_ARG         (prhs[PREAMBLE_ARGC])
#define POSTAMBLE_ARG        (prhs[POSTAMBLE_ARGC])
#define LAST_PROVIDED_ARG     (prhs[LAST_PROVIDED_ARGC])
#define LAST_METRIC_ARG       (prhs[LAST_METRIC_ARGC])
#define LAST_STATE_ARG        (prhs[LAST_STATE_ARGC])
#define LAST_INPUT_ARG        (prhs[LAST_INPUT_ARGC])

#define SIGOUT_ARG          (plhs[SIGOUT_ARGC])
#define STORE_METRIC_ARG      (plhs[STORE_METRIC_ARGC])
#define STORE_STATE_ARG       (plhs[STORE_STATE_ARGC])
#define STORE_INPUT_ARG       (plhs[STORE_INPUT_ARGC])

/* Real and Imaginary part defined for Complex multiplication */
#define CPLX_MULT_REAL(reA,imA,reB,imB) ((reA*reB)-(imA*imB))
#define CPLX_MULT_IMAG(reA,imA,reB,imB) ((reA*imB)+(imA*reB))

static const char MEM_ALLOCATION_ERROR[]  = "Memory allocation error.";

static void checkParameters(int nlhs,      mxArray *plhs[],
                    int nrhs, const mxArray *prhs[])
{
   /* checkParameters checks for the number of input and output
    * arguments
    */

  const char  *msg   = NULL;

  /* Check number of parameters */
  if (nrhs != 12)  {
    msg = "Invalid number of input arguments. mldse_eq expects 12 "
        "input arguments.";
    mexErrMsgIdAndTxt("MLDSE:", msg);
  }
  if(nlhs > 4){
    msg = "Invalid number of output arguments. mldse_eq expects at "
        "most 4 output argument.";
    mexErrMsgIdAndTxt("MLDSE:", msg);
  }


}/* checkParameters */

/* Function: initStateMetric =========================================== */
static void initStateMetric(uint32_T numStates, creal_T value, creal_T *pStateMet)
{
  /* initStateMetric initializes all the statemetrics to the given input value */

  uint32_T indx;
```

```
    for(indx = 0 ; indx < numStates ; indx++)
      pStateMet[indx] = value;

}/* end of initStateMetric */


/* Function: rstInitCond ================================================= */
static void rstInitCond(const int_T alpha,
                                                 int_T chMem,
                const real_T *preamble,
                                                 int_T lenPreamble,
                                                 creal_T *pStateMet,
                const int_T tbLen,
                                                 uint32_T *pTbState,
                uint32_T *pTbInput,
                                                 uint32_T numStates)
{
   /* rstInitCond resets the statemetrics to the initial value,
    * initializes trecaback input and states to zero. This is
    * required to the decoding starts with the initial values.
    */

   int_T    i, limit1;
   uint32_T  j, k, limit2, initState = 0;
   uint32_T  offset    = (uint32_T) pow(alpha,chMem - 1);
   creal_T  czero;
   czero.re = 0.0;
   czero.im = 0.0;

   if(lenPreamble <= 0)
   /* Set all state metrics to 0 */
   {

     initStateMetric(numStates, czero , pStateMet);
   }
   else
   /* Map the preamble to state(s) and assign those state
    * metrics to MaxValue. When the length of the preamble is
    * shorter than the channel length, the preamble would
    * map to more than one state and all those states
    * would receive a state metric of MaxValue.
    */
   {
     initStateMetric(numStates, czero , pStateMet);

     if(chMem > lenPreamble)
     {
       limit1 = lenPreamble;
       limit2 = (uint32_T) pow(alpha, chMem - lenPreamble);
     }
     else
```

```
    {
      limit1 = chMem;
      limit2 = (uint32_T)((chMem>0)?1:0);

    }

    /* Computing the starting state(s) from the preamble */
    for(i=0; i < limit1; i++)
    {
      initState+=(uint32_T) preamble[lenPreamble -1 -i] * offset;
      offset  /= alpha;
    }

    for(k=0; k < limit2 +1; k++)
    {
      creal_T MaxValue;
      MaxValue.re = MAX_int16_T;
      MaxValue.im = MAX_int16_T;
      pStateMet[k + initState] = MaxValue;
    }


  } /* end if(pLenPreamble[0]<=0) */

  /* Set traceback memory to zero */
  for(j = 0; j < (numStates*(tbLen + 1)); j++)
  {
    pTbInput[j] = 0;
    pTbState[j] = 0;
  }
} /* end of rstInitCond */


/* Function: expOutputComp ================================== */
static void expOutputComp(const int_T    alpha,
                                int_T    chMem,
                                uint32_T numStates,
                          const int_T    numSamp,
                          const real_T   *pConstRe,
                          const real_T   *pConstIm,
                          const real_T   *pCtapsRe,
                          const real_T   *pCtapsIm,
                                creal_T  *pExpOutput)
{
  /* expOutputComp computes the expected output when a set of possible
   * signal constellation points are passed through a dispersive channel
   */

  /* Expected output is effectively complex multiplication of signal input
   * and channel coefficients.   */
  uint32_T i, indx1, outIdx;
```

```
    int_T    indx2, indx3, sigIdx, chIdx, temp;

    /* Initialize the expected output vectors.*/
    for(i = 0; i < (numStates * alpha * numSamp); i++)
    {
      pExpOutput[i].re = 0.0;
      pExpOutput[i].im = 0.0;
    }

    /* Loop over the sampled channel length */
    for(indx1 = 0; indx1 < numStates*alpha; indx1++)
    {
      temp = indx1;

      /* Loop over the symbol spaced channel memory */
      for(indx2 = 0; indx2 < chMem+1; indx2++)
      {
        sigIdx = temp%alpha;


        /* Loop over all possible(numSamp) symbol spaced channels
         */
        for(indx3 = 0; indx3 < numSamp; indx3++)
        {
          /* Account for oversampling */

          outIdx = indx1 + (numSamp-1 -indx3) * numStates * alpha;
          chIdx  = (chMem+1)*numSamp-1 -indx3 - numSamp*indx2;

         pExpOutput[outIdx].re += CPLX_MULT_REAL(pConstRe[sigIdx], \
             pConstIm[sigIdx], pCtapsRe[chIdx], pCtapsIm[chIdx]) ;

         pExpOutput[outIdx].im += CPLX_MULT_IMAG(pConstRe[sigIdx], \
             pConstIm[sigIdx], pCtapsRe[chIdx], pCtapsIm[chIdx]) ;

        } /* end of for(indx3=0; indx3 < numSamp; indx3++) */

        temp /= alpha;

      } /* end of for(indx2=0; indx2<chMem+1; indx2++) */
    } /* end of for(indx1=0; indx1<numStates*alpha; indx1++) */
  }  /* end of expOutputComp */


  /* Function: MetricReset ================================== */
  static void MetricReset(int nlhs,      mxArray *plhs[],
                   int nrhs, const mxArray *prhs[],
                   creal_T   *pStateMet,
                   uint32_T *pTbInput,
                                                     uint32_T *pTbState,
                                                     creal_T  *pExpOutput,
```

```
                                   uint32_T *pNxtStates,
                                   uint32_T *pOutputs)
{

 /* Resets all metrics */

        int_T    chMem, lenPreamble;
        uint32_T  k, numStates;
        const int_T    alpha    = (int_T)mxGetNumberOfElements(CONSTPTS_ARG);
  const int_T    tbLen    = (int_T)mxGetScalar(TB_ARG);
  const int_T    numSamp  = (int_T)mxGetScalar(SAMP_PER_SYM_ARG);
        const real_T   *pConstRe = mxGetPr(CONSTPTS_ARG);
  const real_T   *pConstIm = mxGetPi(CONSTPTS_ARG);
        const real_T   *pCtapsRe = mxGetPr(CHAN_COEFF_ARG);
  const real_T   *pCtapsIm = mxGetPi(CHAN_COEFF_ARG);
        const real_T   *preamble = mxGetPr(PREAMBLE_ARG);

        chMem = ((int_T)(mxGetNumberOfElements(CHAN_COEFF_ARG)/numSamp) -1);
        numStates = (uint32_T)pow(alpha,chMem);
        lenPreamble = (preamble[0] == -1) ? 0 : mxGetNumberOfElements(PREAMBLE_ARG);


        expOutputComp(alpha, chMem, numStates, numSamp, pConstRe,
            pConstIm, pCtapsRe, pCtapsIm, pExpOutput);


  /* Compute next states and expected outputs for the equalizer trellis */
  for(k=0; k < numStates*alpha; k++)
  {
    pNxtStates[k] = (uint32_T) floor(k/alpha) ;
    pOutputs[k]   = k ;

  }

  rstInitCond(alpha, chMem, preamble, lenPreamble,
         pStateMet, tbLen, pTbState, pTbInput, numStates);


} /* end of MetricReset */


/* Function: MetricSetup ================================== */
static void MetricSetup(int nlhs,      mxArray *plhs[],
              int nrhs, const mxArray *prhs[],
              creal_T   *pStateMet,
              uint32_T *pTbInput,
              uint32_T *pTbState,
              real_T   *lastMetricRe,
              real_T   *lastMetricIm,
              real_T   *lastInput,
              real_T   *lastState,
```

```
                creal_T  *pExpOutput,
                uint32_T *pNxtStates,
                                                          uint32_T *pOutputs)
{

  /* Resets metric with the values provided as input arguments */

        int_T chMem, lenPreamble;
        uint32_T  i, k, numStates;
        const int_T     alpha    = (int_T)mxGetNumberOfElements(CONSTPTS_ARG);
  const int_T     tbLen    = (int_T)mxGetScalar(TB_ARG);
  const int_T     numSamp  = (int_T)mxGetScalar(SAMP_PER_SYM_ARG);
        const real_T   *pConstRe = mxGetPr(CONSTPTS_ARG);
  const real_T   *pConstIm = mxGetPi(CONSTPTS_ARG);
        const real_T   *pCtapsRe = mxGetPr(CHAN_COEFF_ARG);
  const real_T   *pCtapsIm = mxGetPi(CHAN_COEFF_ARG);
        const real_T   *preamble = mxGetPr(PREAMBLE_ARG);

        chMem = ((int_T)(mxGetNumberOfElements(CHAN_COEFF_ARG)/numSamp) -1);
        numStates = (uint32_T)pow(alpha,chMem);
        lenPreamble = (preamble[0] == -1) ? 0 : mxGetNumberOfElements(PREAMBLE_ARG);


        expOutputComp(alpha, chMem, numStates, numSamp, pConstRe,
             pConstIm, pCtapsRe, pCtapsIm, pExpOutput);


  /* Compute next states and expected outputs for the equalizer trellis */
  for(k=0; k < numStates*alpha; k++)
  {
    pNxtStates[k] = (uint32_T) floor(k/alpha) ;
    pOutputs[k]   = k ;

  }


  /*
   * Copy lastMetric over to pStateMet for each state
   */
        {
    for(i = 0; i < numStates ; i++) {
      pStateMet[i].re = lastMetricRe[i];
      pStateMet[i].im = lastMetricIm[i];
    }
  }

  /* Set up traceback memory with info from the passed in memory */
  {
    for(i = numStates; i < numStates*((uint32_T) tbLen + 1); i++) {

      pTbInput[i] = (uint32_T)lastInput[i-numStates];
```

```
            pTbState[i] = (uint32_T)lastState[i-numStates];
        }
    }
} /* end of MetricSetup */


/* Function: outputPreamble =============================== */
static void outputPreamble(const real_T *preamble,
                          int_T       lenPreamble,
                          const real_T *pConstRe,
                          const real_T *pConstIm,
                          real_T      *thisBlockOutRe,
                          real_T      *thisBlockOutIm)
{
    /* Store the preamble in the output vector */
    int_T i, temp;

    /* Save preamble data into the output vector */
    for(i = 0; i < lenPreamble; i++)
    {
        temp = (int_T) preamble[i];
        thisBlockOutRe[i] = pConstRe[temp];
        thisBlockOutIm[i] = pConstIm[temp];
    }
} /* end of outputPreamble */


/* Function: branchMetricComputation ==================================== */
static void branchMetricComp(const int_T  alpha,
                            uint32_T    numStates,
                            const int_T  numSamp,
                            creal_T     *pExpOutput,
                            creal_T     *pBMet,
                            real_T      *thisBlockInRe,
                            real_T      *thisBlockInIm)
{
    /* Branch Metric Computation computes the Euclidean distance
     * between the received signal and expected output.
     */
    uint32_T indx1, outIdx;
    int_T    indx2, inIdx;

    /* Loop over all branches */
    for(indx1 = 0; indx1 < numStates*alpha; indx1++)
    {
        /* Initialize the branch metrics */
        pBMet[indx1].re = 0.0;
        pBMet[indx1].im = 0.0;

        /* Account for Oversampling */
        for(indx2 = 0; indx2 < numSamp; indx2++)
```

```
    {
      inIdx = numSamp -1 -indx2;
      outIdx = indx1 + (inIdx)*numStates*alpha;

      pBMet[indx1].re +=  CPLX_MULT_REAL(thisBlockInRe[inIdx], \
      thisBlockInIm[inIdx], pExpOutput[outIdx].re, pExpOutput[outIdx].im) ;

      pBMet[indx1].im +=  CPLX_MULT_IMAG(thisBlockInRe[inIdx], \
      thisBlockInIm[inIdx], pExpOutput[outIdx].re, pExpOutput[outIdx].im) ;

    }
  }
}  /* end of branchMetricComputation */

static uint32_T getPostambleState(int_T lenPostamble, int_T ib,
                                  int_T blockSize, int_T alpha,
                                  int_T chMem, const real_T *postamble,
                                  uint32_T minState)
{
  /* getPostambleState computes the state represented by the postamble */

  if (lenPostamble > 0 && ib == blockSize - 1)
  {
    int_T    i, limit1;
    uint32_T finState = 0;
    uint32_T offset   = (uint32_T) pow(alpha, chMem-1) ;

    if (chMem > lenPostamble)
    {
      limit1 = lenPostamble;
    }
    else
    {
      limit1 = chMem;
    }

    /* Computing the ending state from the postamble */
    for(i=0; i< limit1; i++)
    {
      finState += (uint32_T) postamble[lenPostamble -1 -i]\
                  *offset;
      offset   /= alpha;
    }

    minState = finState;

  }

  return minState;

} /* end of getPostambleState */
```

```
/* Function: mldseEqualize =============================== */
static void mldseEqualize(int nlhs,      mxArray *plhs[],
                 int nrhs, const mxArray *prhs[])
{
  /* mldseEqualize performs Maximum Likelihood Differential Sequence Estimation
   * on input signal dispersed by channel. This function initializes
   * the metrics, stores the preamble, performs branch metric computation,
   * ACS and traceback decoding
   */

  /* Get inputs */
  const int_T    alpha    = (int_T)mxGetNumberOfElements(CONSTPTS_ARG);
  const int_T    tbLen    = (int_T)mxGetScalar(TB_ARG);
        const int_T    opmode   = (int_T)mxGetScalar(OPMODE_ARG);
  const int_T    numSamp  = (int_T)mxGetScalar(SAMP_PER_SYM_ARG);

        const real_T   *pConstRe  = mxGetPr(CONSTPTS_ARG);
  const real_T   *pConstIm  = mxGetPi(CONSTPTS_ARG);
        const real_T   *pCtapsRe  = mxGetPr(CHAN_COEFF_ARG);
  const real_T   *pCtapsIm  = mxGetPi(CHAN_COEFF_ARG);
        const real_T   *preamble  = mxGetPr(PREAMBLE_ARG);

        /* Compute channel memory and number of states for the viterbi algorithm */
        int_T chMem = ((int_T)(mxGetNumberOfElements(CHAN_COEFF_ARG)/numSamp) -1);
        int_T lenPreamble = (preamble[0] == -1) ? 0 : mxGetNumberOfElements(PREAMBLE_ARG);
        uint32_T numStates = (uint32_T)pow(alpha,chMem);

        boolean_T  lastprovided  = (boolean_T)(mxGetScalar(LAST_PROVIDED_ARG));
        boolean_T  isContMode    = (boolean_T)(opmode == CONT);

  /* Compute number of symbols in input stream */
  int_T     blockSize     = (int_T)(mxGetN(SIGIN_ARG)/numSamp) - lenPreamble;

  /*
   * Memory allocation : arrays for storing branch metrics, most-update state metrics,
   * and traceback memory.
   */
  creal_T    *pBMet      = (creal_T*)(mxCalloc(numStates*alpha,sizeof(creal_T)));
  creal_T    *pStateMet  = (creal_T*)(mxCalloc(numStates,sizeof(real_T)));
        real_T     *pTempMet   = (real_T*)(mxCalloc(numStates, sizeof(real_T)));
        uint32_T   *pTbState   = (uint32_T *)(mxCalloc(numStates*(tbLen+1),sizeof(uint32_T)));
        uint32_T   *pTbInput   = (uint32_T *)(mxCalloc(numStates*(tbLen+1),sizeof(uint32_T)));
  int_T      *pTbPtr     = (int32_T *)(mxCalloc(1,sizeof(int32_T)));
        creal_T    *pExpOutput = (creal_T *)(mxCalloc(numStates*alpha*numSamp,sizeof(creal_T)));
        uint32_T   *pNxtStates = (uint32_T *)(mxCalloc(numStates*alpha,sizeof(uint32_T)));
        uint32_T   *pOutputs   = (uint32_T *)(mxCalloc(numStates*alpha,sizeof(uint32_T)));

  /* Initialize variables */
  real_T    *thisBlockOutRe, *thisBlockOutIm, *outRe, *outIm, *pStoreInput;
```

```
real_T   *pStoreMetricRe, *pStoreMetricIm, *pStoreState;
int_T    ib, indx1, tbWorkStore, tbWorkLastTb;
uint32_T  currstate, minState, minStateLastTb;

/* Memory limits for numStates */
uint32_T      limit1     = (uint32_T)(pow(2, 16)-1);
uint32_T      limit2     = (uint32_T)(pow(2, 20)-1);

const char *msg        = NULL;

/* Memory allocation of outputs */
      SIGOUT_ARG  = mxCreateDoubleMatrix(1,blockSize +lenPreamble,mxCOMPLEX);
      outRe       = mxGetPr(SIGOUT_ARG);
outIm       = mxGetPi(SIGOUT_ARG);
STORE_METRIC_ARG = mxCreateDoubleMatrix(1,numStates,mxCOMPLEX);
pStoreMetricRe = mxGetPr(STORE_METRIC_ARG);
pStoreMetricIm = mxGetPi(STORE_METRIC_ARG);
pStoreState                    =              (real_T*)(mxGetPr(STORE_STATE_ARG              =
mxCreateDoubleMatrix(numStates,tbLen,mxREAL)));
pStoreInput                    =              (real_T*)(mxGetPr(STORE_INPUT_ARG              =
mxCreateDoubleMatrix(numStates,tbLen,mxREAL)));

/* Memory Limitation error / warnings when channel is specified via port */
if(numStates > limit1)
{
   if( numStates > limit2 )
   {

      mexErrMsgIdAndTxt("MLDSE:","settings describe a trellis with more "
               "than 2^20 states leading to memory "
               "allocation failure.");
   }
   else
   {
      mexWarnMsgIdAndTxt("MLDSE:","create a trellis with more than 2^16 states.");
   }

}

/* Verify memory allocation for temporary and output arrays */
if(SIGOUT_ARG == NULL)
      {
   msg = MEM_ALLOCATION_ERROR;
   goto EXIT_POINT;
}

if(STORE_METRIC_ARG == NULL)
      {
   msg = MEM_ALLOCATION_ERROR;
   goto EXIT_POINT;
}
```

```
if(STORE_STATE_ARG == NULL)
        {
   msg = MEM_ALLOCATION_ERROR;
   goto EXIT_POINT;
}

if(STORE_INPUT_ARG == NULL)
        {
   msg = MEM_ALLOCATION_ERROR;
   goto EXIT_POINT;
}

/* Verify memory allocation for local arrays*/
if(pBMet == NULL)
        {
   msg = MEM_ALLOCATION_ERROR;
   goto EXIT_POINT;
}

if(pStateMet == NULL)
        {
   msg = MEM_ALLOCATION_ERROR;
   goto EXIT_POINT;
}

        if(pTempMet == NULL)
        {
   msg = MEM_ALLOCATION_ERROR;
   goto EXIT_POINT;
}

if(pTbState == NULL)
        {
   msg = MEM_ALLOCATION_ERROR;
   goto EXIT_POINT;
}

        if(pTbInput == NULL)
        {
   msg = MEM_ALLOCATION_ERROR;
   goto EXIT_POINT;
}

        if(pTbPtr == NULL)
        {
   msg = MEM_ALLOCATION_ERROR;
   goto EXIT_POINT;
}

        if(pExpOutput == NULL)
```

```
        {
    msg = MEM_ALLOCATION_ERROR;
    goto EXIT_POINT;
  }

        if(pNxtStates == NULL)
        {
    msg = MEM_ALLOCATION_ERROR;
    goto EXIT_POINT;
  }

        if(pOutputs == NULL)
        {
    msg = MEM_ALLOCATION_ERROR;
    goto EXIT_POINT;
  }

  /*
   * INITIAL state metrics and traceback results set-up
   */

  if(lastprovided)
        {
    /* CONT mode with initial memory provided */
    real_T    *lastMetricRe  = mxGetPr(LAST_METRIC_ARG);
    real_T    *lastMetricIm  = mxGetPi(LAST_METRIC_ARG);
    real_T    *lastState   = mxGetPr(LAST_STATE_ARG);
    real_T    *lastInput   = mxGetPr(LAST_INPUT_ARG);

    MetricSetup(nlhs,plhs,nrhs,prhs,

pStateMet,pTbInput,pTbState,lastMetricRe,lastMetricIm,lastInput,lastState,pExpOutput,pNxtStates,pOutput
s);

  }
        else
        { /* CONT mode with initial memory not provided
                 * OR RST mode
                 */
    MetricReset(nlhs,plhs,nrhs,prhs,pStateMet,pTbInput,pTbState,pExpOutput,pNxtStates,pOutputs);
  }

        /* Store the preamble for RST mode */
        thisBlockOutRe = outRe;
        thisBlockOutIm = outIm;
  outputPreamble(preamble, lenPreamble, pConstRe, pConstIm,
            thisBlockOutRe,thisBlockOutIm);

  /*
   * Loop through input signal
   */
```

```
for(ib = 0; ib < blockSize; ++ib)
      {

   int_T   input, outOffset;
   int_T   inOffset = ib*numSamp;

   const real_T *postamble    = mxGetPr(POSTAMBLE_ARG);
   int_T        lenPostamble = (postamble[0] == -1) ? 0 : mxGetNumberOfElements(POSTAMBLE_ARG);

   real_T  *sigInRe = mxGetPr(SIGIN_ARG);
   real_T  *sigInIm = mxGetPi(SIGIN_ARG);
   real_T  *thisBlockInRe = sigInRe  + inOffset + lenPreamble*numSamp;
   real_T  *thisBlockInIm = sigInIm  + inOffset + lenPreamble*numSamp;

   if(isContMode)
   {
     /* CONTINUOUS mode */
     outOffset        = ib + lenPreamble;
   }
   else
   {  /* RESET EVERY FRAME mode
       *
       * Skip output indexing by (blockSize - tbLen) blocks.
       * Compute metrics and TB tables but do no decoding for
       * the blocks until the end of output buffer
       */
     outOffset        = ((ib - tbLen + lenPreamble)%blockSize);
    }

   /* Branch Metric Computation */
   branchMetricComp(alpha, numStates, numSamp, pExpOutput,
             pBMet, thisBlockInRe, thisBlockInIm);


   /* Add, Compare and Select - State metric update */
   minState = addCompareSelect(numStates, pTempMet, alpha,
                    pBMet, pStateMet, pTbState,
                    pTbInput, pTbPtr, pNxtStates, pOutputs);

   /* Initialize postamble state */
   minState = getPostambleState(lenPostamble, ib, blockSize, alpha, chMem, postamble, minState);

               /* Traceback Decoding */
   input = tracebackDecoding(pTbPtr, minState, tbLen, pTbInput,
                   pTbState, numStates);

               /* Index into the constellation points array
    * and output constellation points
    */
   if((isContMode) || (ib >= tbLen ))
   {
```

```
      real_T *thisBlockOutRe = outRe + outOffset;
      real_T *thisBlockOutIm = outIm + outOffset;
      thisBlockOutRe[0] = pConstRe[input];
      thisBlockOutIm[0] = pConstIm[input];
    }

                /* Save state metrics to pStoreMetric */
    if(ib == blockSize-1)
                {
      for(currstate=0; currstate<numStates; currstate++)
                        {
        pStoreMetricRe[currstate] = pStateMet[currstate].re;
        pStoreMetricIm[currstate] = pStateMet[currstate].im;
      }
    }


  } /* end of blockSize loop */

        /*
   * Capture starting minState and starting tbwork of the
   * last loop
   */
  minStateLastTb = minState;
  tbWorkLastTb   = (pTbPtr[0]!=0) ? pTbPtr[0]-1 : tbLen;
        tbWorkStore    = tbWorkLastTb;

  /*
   * RESET mode :
   *
   * Fill the last tbLen output blocks using the same traceback
   * path, working our way back from the very last block.
   */
        if(!isContMode)
  {
    int_T indx1, input;
                real_T *thisBlockOutRe, *thisBlockOutIm ;

    for (indx1 = 0 ; indx1 < tbLen; indx1 ++)
    {
      input = pTbInput[minStateLastTb+\
                   (tbWorkLastTb*numStates)];

      /* Extract the outputs from the traceback and
       * minState information stored
       */
      thisBlockOutRe = outRe + lenPreamble + \
                  (blockSize -1 -indx1);
      thisBlockOutIm = outIm + lenPreamble + \
                  (blockSize -1 -indx1);
```

```
            thisBlockOutRe[0] = pConstRe[input];
            thisBlockOutIm[0] = pConstIm[input];

            /* Get the minState and traceback information for
             * previous time instant
             */
            minStateLastTb = pTbState[minStateLastTb + \
                                (tbWorkLastTb*numStates)];
            tbWorkLastTb    = (tbWorkLastTb > 0) ? tbWorkLastTb-1: tbLen;
        }
    }


    /*
     * Get pStoreInput and pStoreState
     */

    /* walking horizontally, i.e. through branches */
    for(indx1=0; indx1<tbLen; indx1++)
        {
      /* walking vertically */
      for(currstate=0; currstate<numStates; currstate++)
                {

            pStoreInput[currstate+(tbLen-1-indx1)*numStates]
             = pTbInput[currstate+(tbWorkStore*numStates)];

            pStoreState[currstate+(tbLen-1-indx1)*numStates]
             = pTbState[currstate+(tbWorkStore*numStates)];
      }

      tbWorkStore = (tbWorkStore > 0) ? tbWorkStore-1 : tbLen;
    }


EXIT_POINT:
/* Free memeory allocated to temporary arrays */
mxFree(pBMet);
mxFree(pTempMet);
mxFree(pStateMet);
mxFree(pTbInput);
mxFree(pTbState);
mxFree(pTbPtr);
mxFree(pExpOutput);
mxFree(pNxtStates);
mxFree(pOutputs);

    if(msg != NULL)
        {
      mexErrMsgIdAndTxt("MLDSE:", msg);
    }
```

```
} /* end of mldseEqualize */


/* Function: mexFunction =====================================================
 * Abstract: Check the parameters, if there is any problem, reports an error.
 *        Otherwise, equalize the input signal.
 *
 */
void mexFunction(int nlhs,      mxArray *plhs[],
            int nrhs, const mxArray *prhs[])
{
  checkParameters(nlhs, plhs, nrhs, prhs);
  mldseEqualize  (nlhs, plhs, nrhs, prhs);

} /* end of mexFunction */


#include "Viterbi_acs_tbdec1.h"


/*  Viterbi_acs_tbdec1.h
 *  Helper file required for Add, Compare, Select
 *  and Traceback Decoding operations in the
 *  Maximal Liklihood Differential Sequence Estimation.
 *  This file is required to implement modified Viterbi algorithm.
 */


int_T addCompareSelect(uint32_T numStates, creal_T *pTempMet,
                const int_T alpha, creal_T *pBMet,
                creal_T *pStateMet, uint32_T *pTbState,
                uint32_T *pTbInput, int_T *pTbPtr,
                uint32_T *pNxtStates, uint32_T *pOutputs)
{
  /* Add Compare and Select */
  uint32_T indx1, currstate;
  real_T   renorm    = (real_T) 0.0;
  uint32_T maxState  = 0;
  creal_T  czero, MaxMet;
  czero.re = 0.0;
  czero.im = 0.0;
```

```
for(indx1=0; indx1 < numStates; indx1++) {
    /*
     * Set the temporary state metrics for each of
     * ending states equal to czero
     */
    pTempMet[indx1] = czero;
  }

  for(currstate=0; currstate < numStates; currstate++)
  {
    int_T currinput;

    for(currinput=0; currinput < alpha; currinput++)
    {
      /*
       * For each state and for every possible input:
       *
       *   look up the next state,
       *   look up the associated output,
       *   look up the current branch metric for that output
       *   look up the starting state metric  (currmetric)
       */
      uint32_T offset       = currinput * numStates + currstate;
      uint32_T nextstate    = pNxtStates[offset];
      int32_T  curroutput   = pOutputs[offset];
      creal_T  branchmetric = pBMet[curroutput];
      creal_T  currmetric   = pStateMet[currstate];
      real_T branchmetPow = (real_T) pow((currmetric.re+branchmetric.re),2)\
                      + (real_T) pow((currmetric.im+branchmetric.im),2);
      real_T TempMetPow   = (real_T) pow(pTempMet[nextstate].re,2)\
                      + (real_T) pow(pTempMet[nextstate].im,2);


      /*
```

```
 * Now, perform the Add-Compare-Select procedure:
 * Add the branch metric to the starting state
 * metric. Compare the sum with the best
 * (so far) temporary metric for the ending
 * state. If the sum is more, the following steps
 * consitute the select procedure:
 *    - replace the temporary metric with the sum
 *    - set the current state as the traceback
 *      path from the ending state at this level
 *    - set the current input as the decoder output
 *      associated with this traceback path
 */

if ( branchmetPow > TempMetPow )
{
   pTempMet[nextstate].re  = currmetric.re + branchmetric.re;
   pTempMet[nextstate].im  = currmetric.im + branchmetric.im;
   pTbState[nextstate + (pTbPtr[0] * numStates)] \
                 = currstate;
   pTbInput[nextstate + (pTbPtr[0] * numStates)] \
                 = currinput;

   if ( TempMetPow > renorm )
   {
      renorm = branchmetPow;
      MaxMet = pTempMet[nextstate];
   }
} /* end of if ( branchmetPow > TempMetPow ) */

} /* end of for(currinput=0; currinput<alpha; currinput++)*/

} /* end of for(currstate=0; currstate < pNumStates[0]; currstate++) */

/*
 * Update (and renormalize) state metrics, then find
```

```
 * maximum metric state for start of traceback
 */

for(currstate=0; currstate < numStates; currstate++) {

  pStateMet[currstate] = pTempMet[currstate];

  if(( pStateMet[currstate].re == MaxMet.re) && ( pStateMet[currstate].im == MaxMet.im)) {
    maxState = currstate;
  }
}

return maxState;

} /* EOF addCompareSelect */

int_T tracebackDecoding(int_T *pTbPtr, uint32_T maxState,
            const int_T tbLen, uint32_T *pTbInput,
            uint32_T *pTbState, uint32_T numStates)
{
    int_T indx1;
    int_T tbwork = pTbPtr[0];
    int_T input  = 0;

    /*
     * Starting at the minimum metric state at the current
     * time in the traceback array:
     *    - determine the input leading to that state
     *    - follow the most likely path back to the previous
     *      state by updating the value of minState
     *    - adjust the traceback index value mod tbLen
     * Repeat this tbLen+1 (for current level) times to complete
     * the traceback
     */
```

```
        for(indx1=0; indx1 < tbLen +1; indx1++)
        {
          input    = (int_T) pTbInput[maxState+(tbwork*numStates)];
          maxState = (uint32_T) pTbState[maxState+(tbwork*numStates)];
          tbwork   = (tbwork > 0) ? tbwork-1 : tbLen ;
        }

        /* Increment the traceback index and store */
         pTbPtr[0] = (pTbPtr[0] < tbLen) ? pTbPtr[0]+1 : 0;

         return input;
    }

/* EOF viterbi_acs_tbd.h */
```

References

[1] Y. Liuqing and G. B. Giannakis, "Ultra-wideband communications: an idea whose time has come," *Signal Processing Magazine, IEEE*, vol. 21, pp. 26-54, 2004.

[2] G. C. Corazza, "Marconi's history [radiocommunication]," *Proceedings of the IEEE*, vol. 86, pp. 1307-1311, 1998.

[3] T. W. Barrett, "History of Ultra Wideband radar communications: Pioneers and innovators," presented at Progress in Electromagnetics Research Symposium (PIERS), Cambridge, MA, 2000.

[4] W. M. Lovelace and J. K. Townsend, "The effects of timing jitter and tracking on the performance of impulse radio," *Selected Areas in Communications, IEEE Journal on*, vol. 20, pp. 1646-1651, 2002.

[5] A. Bharadwaj and J. K. Townsend, "Evaluation of the covertness of time-hopping impulse radio using a multi-radiometer detection," 2001.

[6] FCC, "In the matter of Revision of Part 15 of the Commission's Rules Regarding Ultra-Wideband Transmission Systems," FCC First Report and Order, 2002.

[7] E. R. Bastidas-Puga, F. Ramirez-Mireles, and D. Munoz-Rodriguez, "On Fading Margin in Ultrawideband Communications Over Multipath Channels," *Broadcasting, IEEE Transactions on*, vol. 51, pp. 366-370, 2005.

[8] R. Cardinali, L. De Nardis, P. Lombardo, and M. G. Di Benedetto, "UWB ranging accuracy for applications within IEEE 802.15.3a," presented at

Networking with Ultra Wide Band and Workshop on Ultra Wide Band for Sensor Networks, 2005.

[9] S. A. Hanna, "UWB developments within Task Group 1/8 of the ITU-R," presented at Joint UWBST & IWUWBS 2004.

[10] F. Nekoogar, *Ultra-wideband Communications: Fundamentals and Applications*: Prentice Hall Professional Technical Reference Pearson Education, 2005.

[11] G. R. Aiello, "Challenges for ultra-wideband (UWB) CMOS integration," presented at 2003 IEEE Radio Frequency Integrated Circuits (RFIC) Symposium, 2003.

[12] Z. Qi and N. Chun Sum, "Differential TDMA impulse radio systems using delay-sum scheme in UWB channel," presented at Global Telecommunications Conference, 2005. GLOBECOM '05. IEEE 2005.

[13] Z. Irahhauten, H. Nikookar, and G. J. M. Janssen, "An overview of ultra wide band indoor channel measurements and modeling," *Microwave and Wireless Components Letters, IEEE [see also IEEE Microwave and Guided Wave Letters]*, vol. 14, pp. 386-388, 2004.

[14] J. G. Proakis, *Digital Communication*: McGraw-Hill , Inc, 1983.

[15] S. A. Ghorashi, B. Allen, M. Ghavami, and A. H. Aghvami, "An overview of MB-UWB OFDM," presented at Ultra Wideband Communications Technologies and System Design, 2004. IEE Seminar on, 2004.

[16] C. Jun and Z. Zheng, "The overview of synchronization in DS-UWB," presented at Communications and Information Technology, 2005. ISCIT 2005. IEEE International Symposium on 2005.

[17] M. Z. Win and R. A. Scholtz, "Impulse radio: how it works," *Communications Letters, IEEE*, vol. 2, pp. 36-38, 1998.

[18] O. Mi-Kyung, J. Byunghoo, R. Harjani, and P. Dong-Jo, "A new noncoherent UWB impulse radio receiver," *Communications Letters, IEEE*, vol. 9, pp. 151-153, 2005.

[19] H. Liu, "High-rate transmission scheme for pulse-based ultra-wideband systems over dense multipath indoor channels," *Communications, IEE Proceedings-*, vol. 152, pp. 235-240, 2005.

[20] K. Witrisal and M. Pausini, "Equivalent system model of ISI in a frame-differential IR-UWB receiver," presented at IEEE Global Telecommunications Conference (GLOBECOM '04) 2004.

[21] M. Pausini and G. J. M. Janssen, "Analysis and comparison of autocorrelation receivers for IR-UWB signals based on differential detection," presented at Acoustics, Speech, and Signal Processing, 2004. Proceedings. (ICASSP '04). IEEE International Conference on 2004.

[22] Y. Ishiyama and T. Ohtsuki, "Performance evaluation of UWB-IR and DS-UWB with MMSE-frequency domain equalization (FDE)," presented at IEEE Global Telecommunications Conference, 2004.

[23] K. Witrisal, G. Leus, M. Pausini, and C. Krall, "Equivalent system model and equalization of differential impulse radio UWB systems," *Selected Areas in Communications, IEEE Journal on*, vol. 23, pp. 1851-1862, 2005.

[24] T. Ezaki and T. Ohtsuki, "Diversity gain in ultra wideband impulse radio (UWB-IR)," presented at 2003 IEEE Conference on Ultra Wideband Systems and Technologies, 2003.

[25] C. Krall, K. Witrisal, H. Koeppl, G. Leus, and M. Pausini, "Nonlinear Equalization for Frame-Differential IR-UWB Receivers," presented at Ultra-Wideband, 2005 IEEE International Conference on 2005.

[26] J. R. Foerster, V. Somayazulu, and R. Sumit, "A multibanded system architecture for ultra-wideband communications," presented at Military Communications Conference, 2003. (MILCOM 2003), 2003.

[27] S. Oh-Soon, S. S. Ghassemzadeh, L. J. Greenstein, and V. Tarokh, "Performance Evaluation of MB-OFDM and DS-UWB Systems for Wireless Personal Area Networks," presented at Ultra-Wideband, 2005. ICU 2005. 2005 IEEE International Conference on 2005.

[28] E. Saberinia and A. H. Tewfik, "Pulsed and non-pulsed OFDM ultra wideband wireless personal area networks," presented at Ultra Wideband Systems and Technologies, 2003 IEEE Conference on 2003.

[29] P. Armaghan and H. Shafiee, "Receiver architectures for DS-CDMA ultrawideband communication systems," presented at Wireless and Optical Communications Networks, 2005. WOCN 2005. Second IFIP International Conference on, 2005.

[30] B. Zhiquan, Z. Weihua, X. Shaoyi, L. Wei, and K. Kyungsup, "On the performance of multiple access DS-BPAM UWB system in data and image transmission," presented at Communications and Information Technology, 2005. ISCIT 2005. IEEE International Symposium on 2005.

[31] C. Yun-Sung, K. Su-Nam, K. Dong-Wook, and K. Ki-Doo, "The performance analysis of the DS-UWB system with multiband RAKE receivers," presented at Consumer Electronics, 2005. ICCE. 2005 Digest of Technical Papers, 2005.

[32] K. Takizawa and R. Kohno, "Analysis of iterative demapping and decoding for MBOK DS-UWB systems via EXIT chart," presented at Communications, 2005. ICC 2005. 2005 IEEE International Conference on, 2005.

[33] P. Runkle, J. McCorkle, T. Miller, and M. Welborn, "DS-CDMA: the modulation technology of choice for UWB communications," presented at Ultra Wideband Systems and Technologies, 2003 IEEE Conference on 2003.

[34] F. Salem, R. Pyndiah, and A. Bouallegue, "New multiple access frame differential DS-UWB system," presented at Broadband Networks, 2005 2nd International Conference on 2005.

[35] K. Takizawa and R. Kohno, "Low-complexity rake reception and equalization for MBOK DS-UWB systems," presented at Global Telecommunications Conference, 2004. GLOBECOM '04. IEEE, 2004.

[36] Z. Z. Ye, A. S. Madhukumar, P. Xiaoming, and F. Chin, "Performance analysis of a DS-UWB system in the presence of narrowband interference," presented at Vehicular Technology Conference, 2004. VTC 2004-Spring. 2004 IEEE 59th 2004.

[37] L. Zhonghua, Z. Shihua, and W. Shaopeng, "A Novel Adaptive Transmit-Receive Architecture for Indoor DS-UWB Systems," presented at Ultra-Wideband, 2005 IEEE International Conference on, 2005.

[38] L. Fang, W. Deqiang, and Y. Dongfeng, "Code selection for channel estimation in DS-UWB system," 2005.

[39] W. Horie and Y. Sanada, "Novel CSMA scheme for DS-UWB ad-hoc network with variable spreading factor," presented at Ultra Wideband Systems, 2004. Joint with Conference on Ultrawideband Systems and Technologies. Joint UWBST & IWUWBS, 2004.

[40] M. Kamoun, L. Mazet, M. de Courville, and P. Duhamel, "A Fast Maximum Likelihood DS-UWB Equalizer," presented at Signals, Systems and Computers, 2005. Conference Record of the Thirty-Ninth Asilomar Conference on, 2005.

[41] S. S. Mo and A. D. Gelman, "On the power spectral density of UWB signals in IEEE 802.15.3a," presented at IEEE Wireless Communications and Networking Conference (WCNC. 2004) 2004.

[42] S. Imada and T. Ohtsuki, "Pre-RAKE diversity combining for UWB systems in IEEE 802.15 UWB multipath channel," presented at Joint

UWBST & IWUWBS. 2004 International Workshop on Ultra Wideband Systems, 2004.

[43] B. Mielczarek, M. O. Wessman, and A. Svensson, "Performance of coherent UWB Rake receivers with channel estimators," presented at Performance of coherent UWB Rake receivers with channel estimators, 2003.

[44] S. Tantikovit and A. U. H. Sheikh, "Joint multipath diversity combining and MLSE equalization (RAKE-MLSE receiver) for WCDMA systems," presented at Vehicular Technology Conference Proceedings, 2000. VTC 2000-Spring Tokyo. 2000 IEEE 51st 2000.

[45] D. Cassioli, M. Z. Win, and A. F. Molisch, "A statistical model for the UWB indoor channel," presented at Vehicular Technology Conference, 2001. VTC 2001 Spring. IEEE VTS 53rd, 2001.

[46] R. C. Qiu, "A study of the ultra-wideband wireless propagation channel and optimum UWB receiver design," *Selected Areas in Communications, IEEE Journal on*, vol. 20, pp. 1628-1637, 2002.

[47] R. J. M. Cramer, R. A. Scholtz, and M. Z. Win, "Evaluation of an ultra-wide-band propagation channel," *Antennas and Propagation, IEEE Transactions on*, vol. 50, pp. 561-570, 2002.

[48] S. S. Ghassemzadeh, R. Jana, C. W. Rice, W. Turin, and V. Tarokh, "Measurement and modeling of an ultra-wide bandwidth indoor channel," *Communications, IEEE Transactions on*, vol. 52, pp. 1786-1796, 2004.

[49] J. Foerster and Q. Li, "UWB Channel Modeling Contribution from Intel," vol. IEEE P802.15-02/279r0-SG3a.: IEEE P802.15 Wireless Personal Area Networks Std, 2002.

[50] F. Craciun, O. Fratu, and S. Halunga, "Propagation channel simulations for UWB communications," presented at Telecommunications in Modern Satellite, Cable and Broadcasting Services, 2005. 7th International Conference on 2005.

[51] X. Huiyang and W. Gang, "Study on UWB channel modeling and UWB-IR system performance in a typical indoor environment," presented at Wireless Communications, Networking and Mobile Computing, 2005.

[52] M. Z. Win and R. A. Scholtz, "Characterization of ultra-wide bandwidth wireless indoor channels: a communication-theoretic view," *Selected Areas in Communications, IEEE Journal on*, vol. 20, pp. 1613-1627, 2002.

[53] IEEE802.15.3a, "02490r1P802-15_SG3a Channel Modeling Subcommittee Report Final," Channel Modeling Sub-committee Report, 2003.

[54] L. Piazzo and F. Ameli, "On the Inter-Symbol-Interference in several Ultra Wideband systems," presented at Wireless Communication Systems, 2005. 2nd International Symposium on 2005.

[55] L. Chul-Seung, C. Dong-Jun, Y. Young-Hwan, and S. Hyoung-Kyu, "A solution to improvement of DS-UWB system in the wireless home entertainment network," *Consumer Electronics, IEEE Transactions on*, vol. 51, pp. 529-533, 2005.

[56] A. Parihar, L. Lampe, R. Schober, and C. Leung, "Analysis of Equalization for DS-UWB Systems," presented at Ultra-Wideband, 2005 IEEE International Conference on 2005.

[57] S. Ariyavisitakul and L. J. Greenstein, "Reduced-complexity equalization techniques for broadband wireless channels," presented at Universal Personal Communications, 1996. Record., 1996 5th IEEE International Conference on, 1996.

[58] M. Ho, V. S. Somayazulu, J. Foerster, and S. Roy, "A differential detector for an ultra-wideband communications system," presented at Vehicular Technology Conference, 2002. VTC Spring 2002. IEEE 55th, 2002.

[59] S. Qicai, R. J. O'Dea, and F. Martin, "A new chip-level differential detection system for DS-CDMA," 2002.

[60] J. Sumethnapis and K. Araki, "High performance transmission using differential multi-pulse modulation in transmit-reference ultra-wideband (TR-UWB) communications," presented at Wireless Telecommunications Symposium, 2005.

[61] K. Witrisal and Y. D. Alemseged, "Narrowband Interference Mitigation for Differential UWB Systems," presented at Signals, Systems and Computers, 2005. Conference Record of the Thirty-Ninth Asilomar Conference on 2005.

[62] J. Sumethimpis and K. Araki, "High performance transmission using differential multi-pulse modulation (DMPM) in transmit-reference ultra-wideband (TR-UWB) communications," presented at Wireless and

Microwave Technology, 2005. WAMICON 2005. The 2005 IEEE Annual Conference 2005.

[63] K. Changkon, Y. Jiyong, and C. Jong-Wha, "An architecture of decision feedback differential phase detection of M-ary DPSK signals," presented at TENCON 99. Proceedings of the IEEE Region 10 Conference 1999.

[64] F. Adachi and M. Sawahashi, "Viterbi-decoding differential detection of DPSK," *Electronics Letters*, vol. 28, pp. 2196-2198, 1992.

[65] D. Divsalar and M. K. Simon, "Maximum-likelihood differential detection of uncoded and trellis coded amplitude phase modulation over AWGN and fading channels-metrics and performance," *Communications, IEEE Transactions on*, vol. 42, pp. 76-89, 1994.

[66] M. Fujii and A. Kamashita, "Decision feedback multiple-symbol differential detection with MRC diversity," *Electronics Letters*, vol. 30, pp. 2099-2101, 1994.

[67] F. Adachi and M. Sawahashi, "Decision feedback multiple-symbol differential detection for <e1>M </e1>-ary DPSK," *Electronics Letters*, vol. 29, pp. 1385-1387, 1993.

[68] F. Adachi and M. Sawahashi, "Decision feedback differential phase detection of M-ary DPSK signals," *Vehicular Technology, IEEE Transactions on*, vol. 44, pp. 203-210, 1995.

[69] R. Schober, W. H. Gerstacker, and J. B. Huber, "Improving differential detection of MDPSK by nonlinear noise prediction and sequence

estimation," *Communications, IEEE Transactions on*, vol. 47, pp. 1161-1172, 1999.

[70] N. Hamamoto, "Differential detection with IIR filter for improving DPSK detection performance," *Communications, IEEE Transactions on*, vol. 44, pp. 959-966, 1996.

[71] D. Divsalar and M. K. Simon, "Multiple-symbol differential detection of MPSK," *Communications, IEEE Transactions on*, vol. 38, pp. 300-308, 1990.

[72] M. K. Simon, Hinedi, S. M., Lindsey, W. C., *Digital Communication Techniques - Signal Design and Detection*: PTR Prentice Hall, 1995.

[73] G. Forney, Jr., "Maximum-likelihood sequence estimation of digital sequences in the presence of intersymbol interference," *Information Theory, IEEE Transactions on*, vol. 18, pp. 363-378, 1972.

[74] K. Zhou, J. G. Proakis, and F. Ling, "Decision-feedback equalization of time-dispersive channels with coded modulation," *Communications, IEEE Transactions on*, vol. 38, pp. 18-24, 1990.

[75] R. Wood, "Enhanced decision feedback equalization," *Magnetics, IEEE Transactions on*, vol. 26, pp. 2178-2180, 1990.

[76] S. U. H. Qureshi, "Adaptive equalization," *Proceedings of the IEEE*, vol. 73, pp. 1349-1387, 1985.

[77] A. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm," *Information Theory, IEEE Transactions on*, vol. 13, pp. 260-269, 1967.

[78] J. A. Chang, "Ternary sequence with zero correlation," *Proceedings of the IEEE*, vol. 55, pp. 1211-1213, 1967.

[79] S. Stein, "Unified analysis of certain coherent and noncoherent binary communications systems," *Information Theory, IEEE Transactions on*, vol. 10, pp. 43-51, 1964.