Machine Learning for Single-User Ultra Wideband Wireless Communication Systems

Thesis submitted in accordance with the requirements of
the University of Liverpool for the degree of Doctor in Philosophy

by

Mohamed Salem Musbah

March 2010

Declaration

The work in this thesis is based on research carried out at the University of Liverpool. No part of this thesis has been submitted elsewhere for any other degree or qualification and it is all my own work unless referenced to the contrary in the text.

Abstract

Ultra wideband (UWB) technology is intended to provide a high-data-rate short-range wireless communication solution for the forthcoming wireless personal area networks (WPAN). Efficient channel equalisation and estimation solutions are required to be robust against the severe frequency selective fading nature of UWB channels that causes the inter-symbol interference (ISI).

Machine learning algorithms are concerned with the design and development of techniques that allow a computer to "learn" from observations. They have gained special importance with kernel-induced learning that maps highly nonlinear problems to linear formulation. The recent advances in both computing and digital signal processing (DSP) have attracted many researchers to use machine learning for different real time applications.

This thesis investigates and examines machine learning techniques for the application of channel equalisation and estimation for wireless communication systems. Emphasis is laid on the single-user direct sequence UWB (DS-UWB) system, aiming to outperform conventional time-domain and frequency-domain channel equalisation and estimation at a low computational complexity considering low signal processing complexity. The thesis contains three main contributions described as follows.

First, a support vector machine (SVM) based equalisation structure is proposed, which utilises the promising classification performance of the SVM methodology for the purpose of estimating and equalising the frequency-selective fading channels in DS-UWB systems, and is effective to combat the ISI accordingly. The proposed system constructs block-by-block signal arrangement so that a bank of SVM based classifiers is employed at the proposed receiver. With comparable complexity settings, the proposed SVM based equaliser significantly outperforms the conventional receivers in severe ISI conditions. The training complexity of the proposed equaliser is further reduced by employing the least squares support vector classifiers (LS-SVCs) with nearly identical performance. The detection complexity of the LS-SVC based equalisation can be reduced by imposing sparsity to LS-SVC.

Second, a multi-criteria quadratic programming (MCQP) based equalisation structure is proposed, aiming to produce improved performance over SVM by modifying the cost function in the SVM optimisation processing module, so that a better performance and a less training complexity are obtained. The MCQP based equaliser is applied for nonlinear time-variant channel and the results confirm a close performance to the typical optimal Bayesian detector. With a similar structure to that of SVM based system, the MCQP based equlisation and its sparse version are used and investigated for DS-UWB system. Expectedly, results show that the MCQP based equaliser outperforms the SVM based equaliser with better convergence in terms of required pilot size. On the contrary, the sparse version of the MCQP shows worse performance than sparse LS-SVC due to the multi-criteria being applied in optimisation. Furthermore, the MCQP has a high sensitivity to the kernel parameter.

Third, probabilistic classification methods with sparse Bayesian inferred models are investigated, and a relevance vector machine (RVM) based equalisation structure is proposed. Two prediction criteria are adopted for the proposed equaliser: one with maximum *a posterior* (MAP) RVM model parameters, referred to as (MAP-RVM), and the other with integration (marginalisation) over the RVM model parameters distribution, referred to as (MRVM). The proposed equalisers are applied to the DS-UWB system and simulation results confirm the outperformance of RVM based equaliser over the SVM and MCQP based equalisers with less sensitivity to kernel parameters. In particular, MRVM provides a performance very close to the case of frequency-domain equalisation (FDE) with perfect channel state information (CSI).

Acknowledgment

The work in this PhD study was not to be undertaken and finished without great loads of support and encouragement during its process. First of all, I would deeply want to thank my supervisor Dr. Xu (Judy) Zhu for her invaluable continuous guidance and support she gave to me throughout all this project.

I also would like to acknowledge my country Libya represented by the Libyan cultural affairs bureau in London for the financial support that covered my tuition and living expenses during my stay at Liverpool. And for making my dream to come and study in UK come true.

A very special thank goes to my very special person, my wife; Ginan El Gallal for her patience and her great invaluable efforts to take care of me, supporting me, encouraging me, and...loving me all the way. Not to forget acknowledging my lovely three little princesses; Suha, Simah and the newly born Saja, for giving me the motivation to go forward.

Warm thanks are given to fellow workmates in communications lab: N. Surajudeen-Bakinde, N. Zhou and L. Dong for creating family-like atmosphere in the lab, and who share with me the good and tough times during the research work. Also, I would like to thank many friends in Liverpool who really made my stay like home; in particular, O. Khattab, N. Bubaker, Y. Sharkasi and E. Al Attar.

Finally, I am deeply indebted to my dear parents; Salem Musbah and Suha Sergiewa for their love, patience and encouragement to pursue my goals in my life. Their unselfish support was, is and will always be the most precious gift I can receive in this world.

Mohamed Musbah, 2010.

Contents

List of Figures

List of Tables

List of Acronyms

Ultra Wideband Communications

ARAKE	All RAKE
AWGN	Additive white Gaussian noise
BER	Bit error rate
BPM	Biphase modulation
BPSK	Binary phase shift keying
CIR	Channel impulse response
CM1-4	Channel model 1-4
CP	Cyclic prefix
CSI	Channel states information
DS	Direct sequence
DS-CDMA	Direct sequence code division multiple access
DSP	Digital signal processor
DS-UWB	Direct sequence ultra wideband
EM	Electromagnetic
FCC	Federal communications commission
FD	Frequency-domain
FDE	Frequency-domain equalisation
FIR	Finite impulse response
GI	Guard interval
IR-UWB	Impulse radio ultra wideband
ISI	Intersymbol interference
LMMSE	Linear minimum mean square error

LOS	Line of sight
MAI	Multiple access interference
MB-OFDM	Multiband orthogonal frequency division multiplexing
MCSK	M-ary code shift keying
MC-UWB	Multicarrier ultra wideband
MFB	Matched filter bound
ML	Maximum likelihood
MLSE	Maximum likelihood sequence estimator
MMSE	Minimum mean square error
MP	Matched pursuit
MPPM	M-ary pulse position modulation
MRC	Maximal ratio combiner
MUD	Multiuser detection
MUI	Multiuser interference
NLOS	Non line of sight
OFDM	Orthogonal frequency division multiplexing
OOK	On off keying
OPM	Orthogonal pulse modulation
PAM	Pulse amplitude modulation
PHY	Physical layer
PPM	Pulse position modulation
PRAKE	Partial RAKE
RLS	Recursive least squares
SC	Successive channel
SNR	Signal to noise ratio
SRAKE	Selective RAKE
S-V	Saleh-Valenzuela
SW	Sliding window
TH	Time hopping
UWB	Ultra wideband
WPAN	Wireless personal area network

Machine Learning

ANN	Artificial neural networks
EM	Expectation maximisation
ERM	Empirical risk minimisation
GRBF	Gaussian radial basis function
i.i.d.	Identical independent distributed
IRLS	Iterative reweighted least squares
KKT	Karush-Kuhn-Tucker
kNN	k-Nearest Nieghbour
LOO	Leave one out
LOSVOCV	Leave one support vector out cross validation
LS	Least squares
LS-SVC	Least squares support vector classifier
MAP	Maximum a posterior
MAP-RVM	Maximum a posterior-Relevance vector machine
MCQP	Multi-criteria quadratic programming
MRVM	Marginalised relevance vector machine
NBC	Naïve Bayes classifier
PAC	Probably approximately correct
PCVM	Probabilistic classification vector machine
QP	Quadratic programming
RBF	Radial basis function
RV	Relevance vectors
RVM	Relevance vector machines
SRM	Structural risk minimisation
SV	Support vector
SVC	Support vector classifier
SVM	Support vector machine
VC	Vapnik-Chervonenkis

Chapter 1

Introduction

This introductory chapter gives a general overview and organisation of this PhD research. The presented aspects are: the introductory background and the motivation beyond this work, in Section 1.1; Research contributions are summarised in Section 1.2; then the organisation of the thesis in Section 1.3; A list of publications, produced during this PhD study, is provided in Section 1.4.

1.1 Background and Motivation

The increasing demand for wirelessly-connected devices with a huge data rate transmission is inevitable, and the wireless communication community has continuously striven forward to tackle the challenges such as high throughput and low power consumption at wireless devices. Ultra wideband (UWB) communication systems [1, 2, 3] have an unprecedented opportunity to impact communication systems considering the mentioned requirements. So that the enormous bandwidths available (by sending very short impulse-like pulses), the wide scope of the data rate/range tradeoff, and the potential for very-low-cost operation leading to pervasive usage. All these present a unique opportunity for UWB to impact the way people interact with communications systems. In the past decades, UWB has been used for radar, sensing, and military communications [4]. UWB systems have been proposed as an air-interface to the physical layer of wireless personal area networks (WPANs) in the IEEE 802.15 standards using the license-free spectrum [5], and many technologies have been proposed to implement these

UWB systems [6].

On the other hand, the UWB communication systems will face highly dispersive channels relative to the increased bandwidth and therefore high symbol rate. In other words, the channel characteristics in the UWB spectrum of operation suffer from severe inter-symbol interference (ISI) [5, 7], which, accordingly, causes tremendous degradation of the overall system performance. Therefore, a proper equalisation technique is required to mitigate the ISI effects. Some, but not too many, methods and techniques have been proposed from research community to combat the ISI in UWB as will be described in Section 2.5. The system performance in these proposals, however, was unsatisfactory for low-to-medium SNR range which corresponds to the assigned power level for UWB systems. Also, most of the performance improvements were based on the assumption that a perfect knowledge of the UWB channel is present. This motivates the research for more promising alternatives to tackle the performance issue, as well as the insufficient knowledge of channels information.

Machine learning algorithms [8, 9] provide state-of-the-art technologies that solve many scientific and engineering problems. In particular, machine learning algorithms can effectively solve pattern recognition and function regression problems. The main task of a learning machine is to automatically discover the regularities in given data through the use of computer algorithm. In pattern classification, this can be interpreted as to find the class (or, label) of a given data point (or, vector) without precise knowledge of the underlying generating function of the data, by just training the machine with some known data (*i.e.* training set) from the same unknown generator. The statistical learning theory [10, 11] can be considered as a breakthrough to the development of many promising machine learning algorithms that are proved to improve the performance of solving many classification tasks in different fields.

In the world of digital wireless communications, the recent advances in digital signal processing (DSP) technologies have inspired many researchers to apply the machine learning methodology in many communication applications. Most of the existing work perceives the communication task in hand as a pattern recognition solution. For instance, support vector classifiers (SVCs) have been applied for channel equalisation, channel estimation [12, 13] and multiuser detection [14]. However, most previous work only considered simple theoretical (not practical) channels. Other examples are mentioned throughout the text that follows.

The thesis presents different performance-effective channel equalisation and estimation techniques of single-user DS-UWB systems based on machine learning algorithms, which all operate at the receiver side and are incorporated with each other to combat frequency-selective fading channels in DS-UWB systems considering realistic channel models that are globally approved. Both deterministic (*i.e.* single solution) and probabilistic learning models were investigated, and low complexity models were proposed incorporating sparsity at both training and detecting stages. The thesis reveals the application prospect of the proposed machine learning based equalisation and estimation in the impulse radio UWB communication system.

1.2 Research Contributions

The research conducted in this PhD study has produced the following main original contributions.

- SVM based equalisation techniques are proposed for DS-UWB systems. An extensive investigation is provided for SVCs based equaliser. The investigation was developed from basic wireless communication scenario to more realistic UWB channel models. The least squares support vector classifiers (LS-SVCs) based equaliser is also proposed to reduce the training complexity of the SVC based equaliser without sacrifying much of the performance. Furthermore, sparse LS-SVCs based equaliser is proposed to reduce the detection complexity of the LS-SVCs based equaliser, with little performance loss compared to SVCs based equaliser. The performance and complexity analyses are discussed.

- Multi-criteria quadratic programming (MCQP) based equalisation techniques are proposed and examined as a competitive alternative to standard SVM in both performance and complexity. The performance of MCQP equaliser has been intensively investigated in nonlinear channel equalisation scenarios and is also examined for DS-UWB systems. A sparse version of MCQP based equaliser is proposed to reduce the detection complexity. The effect of imposing sparseness, learning convergence, and the sensitivity to kernel parameter are discussed.

- Probabilistic learning models with Bayesian inference methodology are adopted to propose relevance vector machine (RVM) based equalisers. The performance of these equalisers is investigated and examined for nonlinear channel equalisation and for DS-UWB systems, with two variants of prediction strategies; the first is based on a single maximum a posterior (MAP) solution (MAP-RVM), and the other is based on marginalising the posterior of model parameters. The performance and the learning convergence of the proposed system are discussed.

1.3 Thesis Organisation

The work on this thesis has been arranged into 8 chapters, which are organised as follows. After this introductory chapter, an introduction and literature survey on UWB, including the general DS-UWB system model, are presented in Chapter 2. Chapter 3 provides the fundamentals and main principle of the machine learning algorithms and the kernel-induced functions. The SVM based receiver for channel estimation and equalisation is proposed and discussed in Chapter 4. The MCQP based receivers are presented and discussed in Chapter 5. Chapter 6 proposes and describes the RVM based receivers. A comparison discussion to all of the proposed receivers is presented in the end of Chapter 6. The thesis conclusions are drawn in Chapter 7, with hints to more future work suggestions.

1.4 Publication List

A number of publications, that contributes to the thesis, has been arisen during the work on this PhD. They are listed as follows:

Journal papers

- M. Musbah and X. Zhu, “Support Vector Machine based Equalisation for DS-UWB Systems” submitted to Wireless Communications and Mobile Computing.
- M. Musbah and Xu Zhu, “Multi-Criteria Quadratic programming for DS-UWB Channel Equalization”, to be used in the proposed work given in the later chapters.

Conference papers

- M. Musbah and X. Zhu, "Support Vector Machines for DS-UWB Channel Equalisation," in Proc. International Conference on Wireless Communications, Networking and Mobile Computing, WiCom 2007, pp.524-527, Wuhan, China, September 2007.
- M. Musbah and X. Zhu, "Low-Complexity Equalization Based on Least Squares Support Vector Classifiers for DS-UWB Systems," in Proc. IEEE International Conference on Communications ICC 2009, pp.1-5, Dresden, Germany, June 2009.
- M. Musbah and X. Zhu, ”Multi-Criteria Quadratic programming based Low Complexity Nonlinear Channel Equalisation,” in Proc. 17th European Signal Processing Conference (EUSIPCO 2009), pp 328-332, Glasgow, Uk August 2009.
- M. Musbah and Xu Zhu, “Sparse Probabilistic Classification Models for Nonlinear Channel Equalization”, submitted to IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC 2010).

Chapter 2

Ultra Wideband Wireless Communication Systems

This chapter introduces and presents the fundamental principles of wireless UWB communication systems. In particular, it considers the impulse-based format, since the work in this research considers this form in the applied algorithms. Also, a general DS-UWB system model is formulated to be used in proposed the proposed work in later chapters. The organisation of this chapter is as follows. A brief introductory background to the UWB system is presented in Section 2.1. The multiple access schemes for UWB systems are summarised in Section 2.2. Section 2.3 presents the UWB channel modeling principles and the approved model that is widely adopted in literature. The general DS-UWB system model is provided in Section 2.4. Section 2.5 provides a description and basic simulations for the conventional receivers that are used in UWB systems.

2.1 Introduction

The ever-increasing demand for the amount of data transmission requires the increase in the available bandwidth for many applications (such as multimedia and number of communicating devices and so on). UWB technology is one of the promising solutions in terms of high-speed short-range wireless communication systems, or, in other words, a WPAN which links devices, both, and as diverse as, portable and fixed appliances, personal computers, and entertainment equipment [5]. Unlike conventional radio systems which operate within a relatively narrow bandwidth, the impulse-based UWB system operates across a very wide radio spectrum (up to a few GHz), by transmitting a series of very short pulses which results in very high-resolution timing information [15]. Furthermore, UWB can also transmit a large amount of data at a data rate of several hundreds Mbps with an extreme low power over a short distance of a few meters [16]. In a multipath dominant environment, larger transmission bandwidth results in the ability to increase the resolution of multipath arrivals to a very fine level, which leads to reduced fading per resolved path, since the impulsive nature of the transmitted waveform prevents significant overlap and, hence, reduces possibility of destructive combining [17].

As a regulatory example, the federal communications commission (FCC), in USA, has regulated the power levels for UWB systems to be very low (below -41.3 dBm), which allows the UWB technology to co-exist with some services that also operate in subbands of 3.6-10.1 GHz, such as the global positioning system (GPS) and the IEEE 802.11 WLANs. Figure 2.1 depicts the spectral mask mandated by FCC 15.517 (b,c) for UWB indoor systems [18]. Although UWB signals can propagate for greater distances at higher power levels, current FCC regulations enable high-rate (above 110 Mbps) data transmissions over a short range (10-15 m) at very low power spectral density. Major efforts are currently under way (some have been done and published) by the IEEE 802.15 working group for standardising UWB wireless radios for indoor (home and office) multimedia transmissions. Similar to the frequency reuse principle exploited by wireless cellular architectures, low-power, short-range UWB communications are also potentially capable of providing high spatial capacity, in terms of bits per second per square meter. In addition, the high resolution of UWB pulse enables the technology to be used in many ranging and location-determining applications such as radar and position estimation technologies [6].

To fulfill the above expectations and confinements, research and development on UWB have to cope with formidable challenges that limit their bit error rate (BER) performance, capacity, throughput, and network flexibility. Those include high sensitivity to optimal exploitation of fading propagation effects with pronounced frequency-

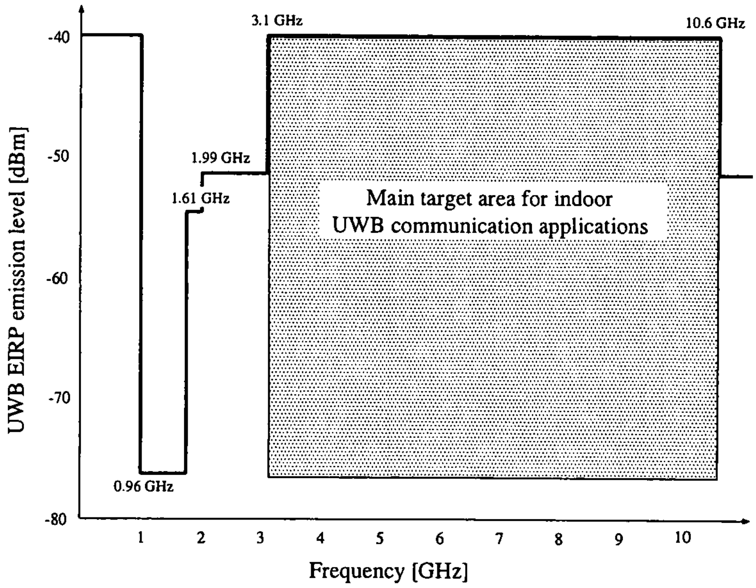


Figure 2.1: The spectral mask mandated by FCC [18].

selectivity, low-complexity constraints in decoding high-performance multiple access protocols. Also, the strict power limitations imposed by the desire to minimise interference among UWB communicators, and the coexisting with other systems, particularly GPS, aircraft radar, and WLANs [5], are challenging the design of UWB technologies. These challenges call for advanced DSP expertise to accomplish tasks such as channel estimation and equalisation [19].

A brief history of the UWB technology

The modern era in the UWB started in the early 1960s from work in time-domain electromagnetics and was led by Harmuth at Catholic University of America, Ross and Robins at Sperry Rand Corporation, and Van Etten at the United State Air Force (USAF) Rome Air Development Centre. Harmuth's work culminated in a series of books and articles between 1969 and 1990 [20]. Harmuth, Ross, and Robins all referred to their system as baseband radio [4]. During the same period, engineers at Lawrence Livermore, Los Alamos National Laboratories (LLNL and LANL), and elsewhere performed some of the original research on pulse transmitters, receivers, and antennas [2].

A major breakthrough in UWB communications occurred as a result of the development of sampling oscilloscope by both Tektronix and Hewlett-Packard in the 1960s

[2], These sampling circuits not only provided a method to display and integrate UWB signals, but also provided simple circuits necessary for sub-nanosecond, baseband pulse generation. In the late 1960s, Cook and Bernfeld published a book that summarised Sperry Rand Corporation's developments in pulse compression, matched filtering, and correlation techniques [20]. The invention of a sensitive baseband pulse receiver by Robbins in 1972, as a replacement of the sampling oscilloscope, led to the first patented design of a UWB communication system by Ross at Sperry Rand Corporation [4].

By the early 1970s, the basic designs for UWB radar and communication systems evolved with advances in electronic component technology. The first ground-penetrating radar based on UWB was commercialised in 1974 by Morey at the Geophysical Survey Systems Corporation [20]. In 1994, McEwan at LLNL developed the Micropower Impulse Radar (MIR), which provided a compact, inexpensive and low power UWB system for the first time.

Through the late 1980s, the UWB technology was alternately referred to as baseband, carrier-free or impulse the term "ultra wideband" not being applied until approximately 1989 by the U.S. Department of Defense. By that time, UWB theory, techniques and many hardware approaches had experienced nearly 30 years of extensive development. By 1989, for example, Sperry had been awarded over 50 patents in the field covering UWB pulse generation and reception methods, and applications such as communications, radar, automobile collision avoidance, positioning systems, liquid level sensing and altimetry [21].

In 1993, Robert Scholtz at the University of Southern California wrote a landmark paper that presented a multiple access technique for UWB communication systems [22]. With a viable multiple access scheme, UWB became capable of supporting not only radar and point-to-point communications but wireless networks as well.

Recently, there has been a rapid expansion of the number of companies and government agencies involved with UWB, growing from a handful in mid 1990s that included Multispectral Solutions, Time Domain, Aether Wire, Fantasma Networks, LLNL and a few others, to the plethora of today's players. These companies and many governmental bodies have spent many years investigating the effect of UWB emissions on existing narrowband systems. The results of those studies were used to inform the FCC of how UWB could be allowed to operate [2]. In 2003, the first FCC certified commercial system was installed, and in April 2003 the first FCC-compliant commercial UWB chipset wes announced by Time Domain Corporation [23].

UWB Signalling

In this subsection, the fundamental properties of the UWB signal have been investigated. In the context of UWB, there are two common forms of UWB signals: one based on sending very short duration pulses to convey information, referred to as impulse radio UWB (IR-UWB), and another approach using multiple simultaneous carriers, referred to as called multicarrier UWB (MC-UWB). Each approach has its relative technical merits and demerits. This research will primarily focus on IR-UWB since the most common form of MC-UWB modulation, orthogonal frequency division multiplexing (OFDM), has been extensively investigated in the context of wideband OFDM. Hence, and for convenience, the term UWB will be used through the research instead of IR-UWB.

So far two UWB technologies have been proposed to the IEEE 802.15.3a task group TG3a: 1) direct-sequence UWB (DS-UWB) [2], supported by the UWB Forum [24]; 2) multi-band orthogonal frequency division multiplexing (MB-OFDM) UWB [4], supported by the WiMedia Alliance [25]. After numerous attempts by each proposer and several discussion sessions to choose one of the two UWB technologies for wireless personal area networks (WPANs), no conclusion was reached. As a result, the plan for a unique standardisation process by the IEEE 802.15.3a task group (TG3a), has been withdrawn. Hence, it is expected that both the DS-UWB and MB-OFDM UWB technologies will coexist in the future [26].

Advantages of IR-UWB

In addition to its high data rate capabilities, the impulse radio UWB technology has some attractive advantages. The following list summarises the most significant benefits that are expected from UWB systems:

- Firstly, it is a baseband modulation and demodulation technology, therefore, the systems become less complex, allowing for significantly lower cost and smaller size, since they do not use any RF/IF conversion stages [15].
- Secondly, because of the combination of large spectrum, lower power, and pulse shaped data, the impulse radio system has very high multipath resolution, which leads to the reduced fading and thus improved communication quality, as well as granting higher level of security and privacy than narrowband radio systems [17].
- Thirdly, the repetition period of the pulse is very large compared with the pulse duration, which has two advantages: one is that using time hoping multiple access technology, the impulse radio system can accommodate many users; the other is that the power spectral density is very low, so the impulse radio system has very little impact on other narrow band systems operating in the same frequency range [27].
- Eventually, the possibility of using both precise ranging (object location) and high speed data communication in the same wireless device, which introduces new devices and applications. For example, collision avoidance radar and communication can give accident-free smooth traffic flow [18].

UWB signal definition

The UWB signal is defined as a signal with bandwidth greater than 20% of the centre frequency (this percentage represents the fractional bandwidth) [19] or greater than 500 MHz of absolute bandwidth [17]. Unlike conventional communications, UWB does not use a sinusoidal carrier to convey information. Instead, the transmit signal is a series of extremely short baseband pulses (in nanoseconds), to obtain a bandwidth of several GHz. In general, the transmit signal can be mathematically represented as [1]

$$s(t) = \sum_{i=-\infty}^{\infty} Ap(t - iT_f) \tag{2.1}$$

where A is the amplitude of the pulse which, for binary coding, equals to $\pm\sqrt{E_p}$, with E_p denoting the energy per pulse, $p(t)$ is the transmitted pulse shape with normalised energy, and T_f is the frame repetition time which is defined as the time interval in which one pulse is transmitted. An important note one should realise is that the pulse width (duty cycle) is much smaller than frame repetition time T_f. Most practical systems will use some form of pulse-shaping to control the spectral content to conform to spectrum restrictions. UWB symbols usually contain a group of repeated, or formatted, pulses so that the symbol duration T_s equals to NT_f. The general form in (2.1), however, is to be mapped according to the information via some sort of modulation as will be shown soon.

UWB pulse shapes

The most common form for the pulse shape in the literature on UWB, is *Gaussian* and its derivatives [2] as they are easy to describe and work with. The Gaussian pulse shape can be formulated in the following form:

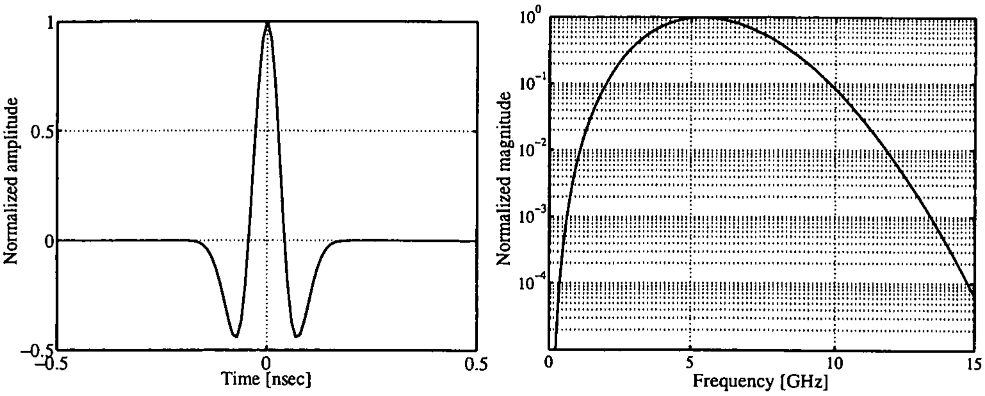
$$p(t) = K_0 e^{-2\pi(t/\tau_m)^2} \tag{2.2}$$

where K_0 is a power normalising factor. The second derivative of a Gaussian function with zero-mean, also known as a *Gaussian Doublet*, is illustrated in Figure 2.2(a). The idealised Gaussian Doublet pulse can be expressed as following

$$p(t) = \left[1 - 4\pi\left(\frac{t}{\tau_m}\right)^2\right] e^{-2\pi(t/\tau_m)^2} \tag{2.3}$$

where τ_m is a parameter determining the time and frequency characteristics of the Gaussian Doublet pulse. This pulse is often used in UWB systems because of the simplicity of its generation. It is simply a square pulse which has been shaped by the limited rise and fall times of the pulse and filtering effects of the transmit and receive antennas. A square pulse can be generated by switching a transistor on and off quickly.

The spectrum of the Gaussian Doublet is shown in Figure 2.2(b). The centre frequency can be seen to be approximately 5 GHz, with the 3 dB bandwidth extending over several GHz.



(a) (b)

Figure 2.2: (a) Idealised UWB pulse shape and, (b) idealised spectrum of a single UWB pulse [18].

UWB signal waveform simulator

This simulation is aimed to generate a UWB pulse shape according to second derivative Gaussian waveform. Figure 2.3 depicts the output of the generator for 2 ns of pulse width and 10 ns of repetitive duration. The sampling frequency is 20 GHz.

The power spectrum of this signal is provided in the following diagram (Figure 2.4). It can be noticed that the 10 dB bandwidth of this signal is approximately 600 MHz.

UWB signal modulation

The ultra wide bandwidth and exceptionally narrow pulses (the carrierless nature) of UWB signals make it difficult to employ conventional narrowband modulation techniques in UWB systems [3]. However, the transmitted pulses should be characterised in some manner to convey information, *i.e.*, modifying Equation (2.1) to include information bearing contents.

The modulation schemes that are used in UWB can be categorised into two broad types: 1) the time-based techniques. The most common method in this category is the pulse position modulation (PPM) [28] where each pulse is delayed or sent in advance of a regular time scale. 2) the shape-based techniques. Many considerable methods are laid in this category. The common method is the bi-phase modulation (BPM) where pulses are created with opposite phase [18]. Also, shape-based modulation can be attained via generating special forms of orthogonal pulses as in orthogonal pulse modulation

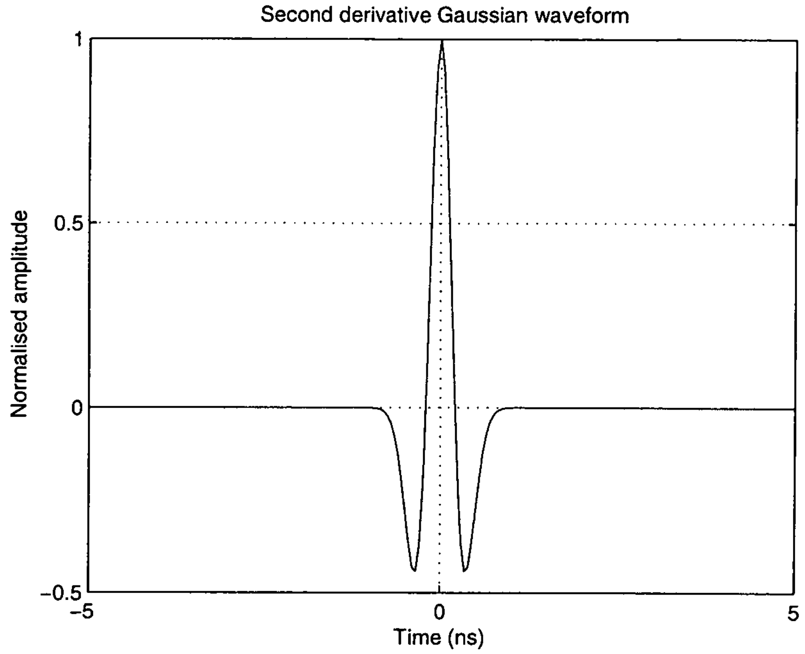


Figure 2.3: UWB Gaussian pulse shape.

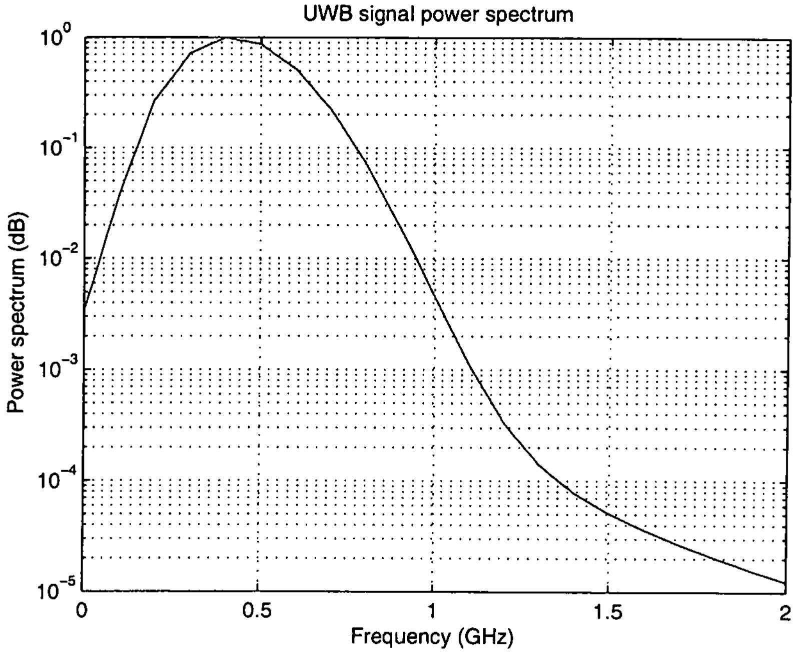


Figure 2.4: Power spectrum of UWB pulse

(OPM). Some of the conventional forms of modulation are also used in shape-based UWB systems. This includes the pulse amplitude modulation (PAM) in which digital information is contained by varying the amplitude of transmitted pulse. Binary phase shift key (BPSK) can be considered as a special case of both PAM and BPM. On-off keying (OOK) is another well-known method that can be employed in modulating UWB pulses [19].

More recently, an M-ary code shift keying (MCSK) impulse modulation has been proposed in [29], where the effect of multipath-delayed pulses on M decision variables was explicitly provided in terms of channel impulse response coefficients. By randomising locations of the transmit pulse, the MCSK demonstrates a performance gain over M-ary pulse position modulation (MPPM) as it reduces the effects of multipath delays on the decision variables.

2.2 Multiple Access Schemes for UWB Systems

One of the most important issues that have to be considered in designing a communication system is the multiuser capabilities since single user detection is typically suboptimal [17]. A number of multiple access schemes have been proposed to enable a channel-sharing purpose for multi-user networking. The major access schemes for the pulse based UWB are: the time hopping (TH) UWB, and the direct sequence (DS) spread spectrum UWB. The following will provide a brief overview of these schemes [30].

TH-UWB

In TH-UWB, each data symbol, or frame duration, is divided into N_c chips of duration T_c. Each user is assigned a unique pseudo-random time shift pattern, $h_{u,n}$, where u is the user index, called a TH sequence, which provides an additional time shift to each pulse in the pulse train. The n^{th} pulse undergoes an additional shift of $h_{u,n}T_c$, where T_c is also the duration of an addressable time delay bin [22]. The addressable TH duration must be strictly less than the frame time [30].

In order to mathematically represent the typical TH-PAM UWB signal, consider the following formulation

$$s_{TH-PAM}^{(u)}(t) = \sum_{n=-\infty}^{\infty} \sqrt{E_s} b_{u,n} p\left(t - nT_f - h_{u,n}T_c\right), \tag{2.4}$$

where E_s is the symbol energy, $b_{u,n}$ is the n^{th} symbol for the u^{th} user, and $p(t)$ is the applied pulse shape.

DS-UWB

The symbols in DS-UWB are represented by a sequence of pulses that are pulse-amplitude-modulated to the corresponding symbol. The modulation schemes that are used accompanying DS can be summarised as PPM and PAM. For binary DS-PPM-UWB systems, information bit 1 is represented by a frame of zero delay shift, and information bit 0 is represented by the same frame of pulses but with a delay of τ relative to the time reference.

In DS-PAM systems, the information bits are represented by bipolar pattern of the pulse sequence [30]. As an example of a binary DS-PAM UWB system, the transmit signal for the u^{th} user can be mathematically expressed as

$$s_{DS-PAM}^{(u)}(t) = \sum_{n=-\infty}^{\infty} \sqrt{E_s} b_{u,n} \sum_{i=1}^{N_c} p\left(t - nT_f - c_{u,i}T_c\right), \tag{2.5}$$

where $c_{u,i}$ is the i^{th} component of the u^{th} user spreadingcode.

Low cross-correlation ternary codes

The ternary codes have shown superior performance in terms of signal correlation [31]. Hence, it is being widely used for impulse-based DS-UWB systems and, subsequently, is adopted for IEEE 802.15-03/334r5 proposals [32]. Ternary direct sequences includes bursts of zero signal amplitude as a natural extension of the typical binary antipodal format. An example of designing a low cross-correlation ternary codes can be found in [31] where zero correlation zone sequences have been proposed.

TH-UWB vs. DS-UWB

The multiple accessing capability of these approaches depends on a variety of factors that contribute to multiple access interference (MAI) at the receiver detector input, namely, the properties of the respective sequence design and the type of receiver used. In [33], it is concluded that, with a matched filter receiver, DS-UWB multiple access is more suitable for higher rates, as it can accommodate more users compared to TH-PPM for a given BER. At lower data rates, the multiple accessing capacity of the two systems are approximately the same. In such cases, TH-PPM may be preferable over DS-UWB, since it is potentially less susceptible to the near–far effect. For a multiuser detector, the system capacity of the two approaches are approximately the same.

2.3 Channel Modeling

In order to evaluate the performance of any communication system, as well as to design a receiver, it is necessary to have a good knowledge of how this system behaves on the signal transmitted through it. Such behaviour includes attenuation, delay, and all possible factors that may distort the signal. This requires to develop a sufficient mathematical model representing the overall relationship between transmitted and received signals. Generally, for wireless channels, there are two prevalent types of modelling of electromagnetic (EM) wave propagation. The first can be termed *Deterministic Modelling,* which attempts to model the exact interaction of the EM wave in the specific environment of interest. This type is often used to predict coverage patterns in wireless systems when detailed information concerning the environment is available. The second type of modelling attempts to model the relevant statistics of the received signal and is called *Statistical Modelling.* Statistical modelling is particularly useful in communication system development where the system must work in a wide variety of environments [2]. The statistical models will be discovered and studied in this research.

2.3.1 Overview of Wireless Channel Propagation

Mainly, propagation models are classified into two categories: large scale and small scale. Large-scale models predict the mean signal strength for the transmission distance.

These models are useful in estimating the radio coverage area of the transmitter. The small-scale propagation models, on other hand, characterise the rapid fluctuations of the received signal strength over very short travel distances or short time durations [34].

2.3.2 UWB Channel Modelling

For UWB systems, the most potential operational environment is the indoor channel because of the low transmission power that restricts UWB communications. Therefore, UWB channel modelling will concern the effects of small-scale propagation much more than large-scale models. In addition, the model should be relatively simple to use in order to allow physical layer (PHY) proposers to use it and evaluate the performance of their proposals in typical operational environments. Currently, the IEEE802.15.3a [35] is an approved model for UWB channels which concludes many indoor-channel models that are considered to match the PHY UWB operational environment.

IEEE 802.15.3a standard

In November 2002, the channel modelling subcommittee of the IEEE 802.15.3a Task Group [35] recommended a channel model which includes the previous proposals and refinements that capture the important characteristics of UWB channel. This model is basically a modified version of the Saleh-Valenzuela (S-V) model [36], where the average multipath power is considered to be distributed following a *Log-Normal* pattern instead of *Rayleigh* distribution, and multipath components have a phase shift of either 0 or π instead of uniformly distributed phases. The total number of paths is defined as the number of multipath arrivals with expected power within 10 dB from that of the strongest path [7].

In the proposed model, according to the S-V model, multipath components are assumed to arrive in exponential power decaying groups (described as *clusters*), and the time arrival for each cluster follows a Poisson's process with a rate of Λ. Within each cluster, the paths also have exponential power decaying and arrive following Poisson distribution with rate $\lambda > \Lambda$ [19]. The time arrivals of the clusters and inside components, however, are defined by the exponential distributions which are represented respectively by

$$\begin{aligned} P(T_m \mid T_{m-1}) &= \Lambda e^{-\Lambda(T_m - T_{m-1})},\ m > 0 \\ P(\tau_{m,n} \mid \tau_{m,n-1}) &= \lambda e^{-\lambda(\tau_{m,n} - \tau_{m,n-1})},\ n > 0 \end{aligned} \tag{2.6}$$

The channel impulse response (CIR) can be expressed as

$$h(t) = X \sum_{m=0}^{\infty} \sum_{n=0}^{\infty} \alpha_{m,n} e^{j\theta_{m,n}} \delta(t - T_m - \tau_{m,n}) \tag{2.7}$$

where $T_m + \tau_{m,n}$ ($\tau_{m,0} = 0$) denotes the arrival time of the nth multipath component of the mth cluster, $\theta_{m,n}$ are equally distributed values of 0 and π, and $\alpha_{m,n}$ are independent Log-Normal random variables with power decaying as shown in Equation (2.8) below, where $\Gamma > \gamma$.

$$E\left\{\alpha_{m,n}^2\right\} = E\left\{\alpha_{0,0}^2\right\} e^{-\frac{T_m}{\Gamma}} e^{-\frac{\tau_{m,n}}{\gamma}} \tag{2.8}$$

The model also includes a shadowing term X to account for the total received multipath energy variation that results from blockage of the line-of-sight (LOS) path. The shadowing factor is also Log-Normal distributed.

Since it is difficult to match all possible channel characteristics, the main characteristics of the channel that are used to derive the above model parameters are chosen to be the following:

- Mean excess delay
- Root mean squares (RMS) delay spread
- Number of multipath components (defined as the number of multipath arrivals that are within 10 dB of the peak multipath arrival)
- Power decay profile

The main channel characteristics that are used by the Task Group subcommittee to determine the model parameters are the first three above, since the model parameters are difficult to match to the average power decay profile. Table 2.1 lists some initial

Table 2.1: Multipath channel target characteristics and model parameters [35].

Target Channel Characteristics[5]	CM1[1]	CM2[2]	CM3[3]	CM4[4]
Mean excess delay (ns)	5.05	10.38	14.18	
RMS delay (ns)	5.28	8.03	14.28	25
NP$_{10dB}$			35	
NP$_{(85\%)}$	24	36.1	61.54	
Model Parameters				
Λ ($1/ns$)	0.0233	0.4	0.0667	0.0667
λ ($1/ns$)	2.5	0.5	2.1	2.1
Γ	7.1	5.5	14	24
γ	4.3	6.7	7.9	12
Model Characteristics				
Mean excess delay (ns)	5.0	9.9	15.9	30.1
RMS delay (ns)	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

[1] This model is based on LOS ($0-4m$) channel measurements.

[2] This model is based on NLOS ($0-4m$) channel measurements.

[3] This model is based on NLOS ($4-10m$) channel measurements.

[4] This model was generated to represent an extreme NLOS multipath.

[5] These characteristics are based upon a $167ns$ sampling time.

model parameters for a couple of different channel scenarios (NLOS refers to non LOS) that were found through measurement data.

IEEE802.15.4

The IEEE established the 802.15.4 Study Group to define a new physical layer concept for low-data-rate applications. The IEEE802.15TG4 is chartered to investigate low-data-rate solutions for very low power and very low complexity systems. It is intended to operate in unlicensed, international frequency bands. Potential applications are sensors, interactive toys, smart badges, remote controls, and home automation, etc. [37].

Discrete UWB indoor channel model

The CIR model in Equation (2.7) can be finitely re-expressed as

$$h(t) = \sum_{i=1}^{C_L} \sum_{k=1}^{K} a_{i,k} \delta(t - T_i - \tau_{i,k}) \tag{2.9}$$

where T_i is the delay of the i^{th} $(i = 1, ..., C_L)$ cluster, and $\tau_{i,k}$ is the delay of the k^{th} $(k = 1, ..., K)$ path within the i^{th} cluster relative to T_i. T_i and $\tau_{i,k}$ are described with a double-Poisson process and are rounded to integer multiples of the delay resolution T_c. $a_{i,k} = p_{i,k}\xi_{i,k}$ is the path gain of the k^{th} path within the i^{th} cluster, where $p_{i,k} \in \{+1, -1\}$ denotes the equally-likely random polarity (the possible phases for real coefficients), and the fading amplitude $\xi_{i,k}$ is real valued and follows the lognormal distribution [7].

With τ_{exc} denoting the multipath delay spread, $L = \tau_{exc}/T_c$ is the total number of paths, and h_l is the sum of all $a_{i,k}$ at time index l where $l = \lfloor (T_i + \tau_{i,k})/T_c \rfloor$. Due to clustering of multipath components [38], the channel does not necessarily have multipath arrivals within each delay bin. This is accounted for by setting $h_l = 0$ for any lT_c that has no path arrival. Therefore, the CIR of Equation (2.9) can be simplified to

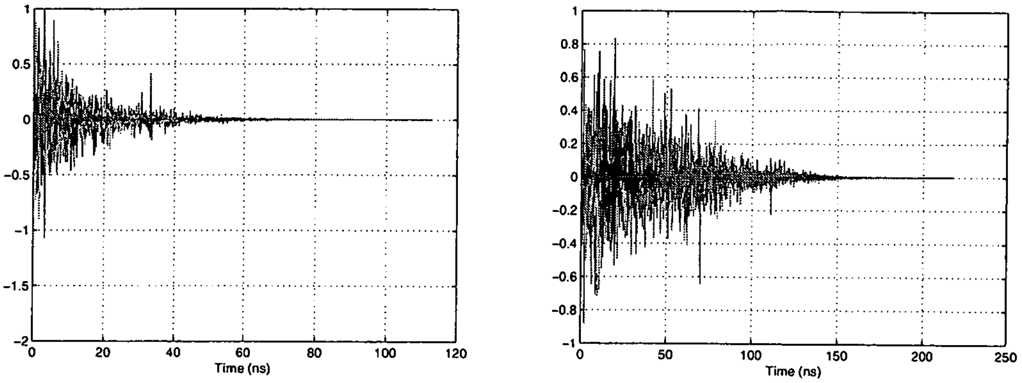
$$h(t) = \sum_{l=1}^{L} h_l \delta \left(t - (l-1)T_c\right) \tag{2.10}$$

This discretised model that represents the CIR can be further expressed by a $N_c(L_{GI} + M) \times N_c(L_{GI} + M)$ matrix $\mathbf{H}$ [39] as follows:

$$\mathbf{H} = \begin{bmatrix} h_1 & 0 & 0 & \cdots & 0 \\ \vdots & h_1 & 0 & \cdots & 0 \\ h_L & & \ddots & & \vdots \\ \vdots & \ddots & & h_1 & 0 \\ 0 & \cdots & h_L & \cdots & h_1 \end{bmatrix} \tag{2.11}$$

2.3.3 UWB Channel Realisations

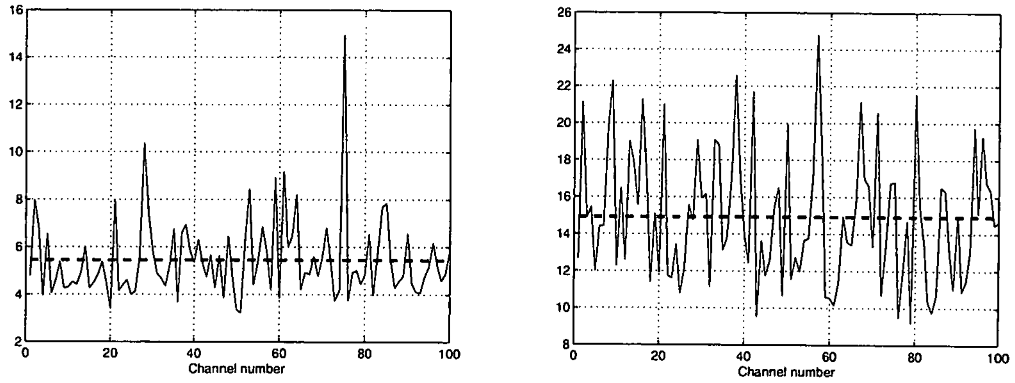
As shown in Subsection 2.3.2, the proposal of the IEEE802.15.3a working group is the widely known standardisation for indoor UWB channel modelling. In this experiment, 100 channel realisations of this standard have been implemented and discretised to a time resolution of $T_c = 0.167ns$ according to the specifications discussed in Subsection 2.3.2. Two environments have been considered in this simulation: CM1 for LOS scenario, and CM3 for medium-range NLOS scenario. The results of these realisations have been included in the following figures and descriptions. These scenarios will be



(a) CM1

(b) CM3

Figure 2.5: Channel impulse response



(a) CM1

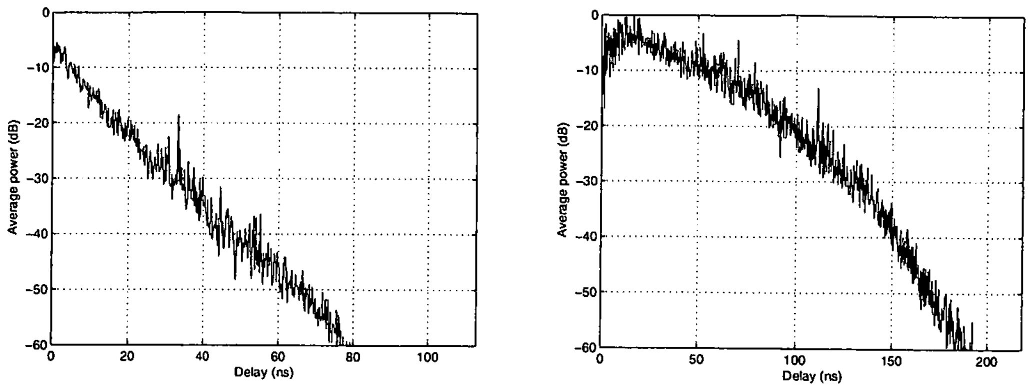
(b) CM3

Figure 2.6: RMS delay spread

adopted for the rest of the work in this study.

The CIRs of both CM1 and CM3 are shown in Figure 2.5, and it is clearly noticed that the NLOS model in CM3 has a longer excess time delay (much more time dispersive) than the LOS scenario in CM1. The RMS time delay for both channel models in 100 realisations is illustrated in Figure 2.6, where the thick-dashed line of both graphs represents the average of τ_{RMS} , and results show that $\tau_{RMS} = 5\,ns$ for CM1, and $\tau_{RMS} = 15\,ns$ for CM3.

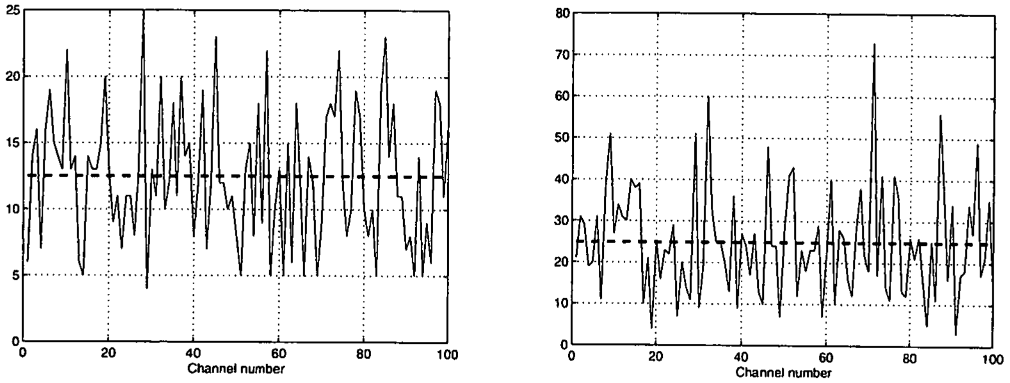
Figure 2.7 depicts the power-delay profile (in dB) of each of the tested channels. The graphs in Figure 2.7(a) and Figure 2.7(b) show that the significant power components are laid very close to zero for CM1 (*i.e.* most power is in direct LOS), whereas the significant power components for CM3 are distributed over a wider range of time delay.



(a) CM1

(b) CM3

Figure 2.7: Power delay profile

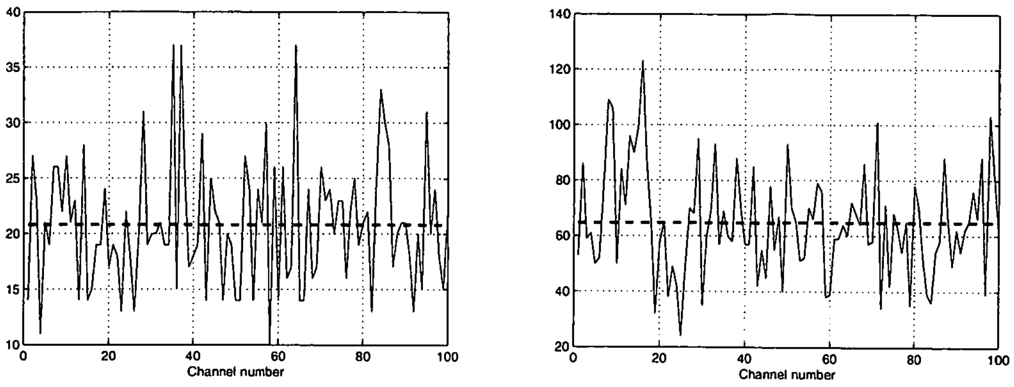


(a) CM1

(b) CM3

Figure 2.8: Number of significant paths within 10 dB from maximum path

The number of significant power components are shown in Figure 2.8 and Figure 2.9. These numbers are calculated for two criteria: one is the number of power components within 10 dB of maximum path power, *i.e.*, Figure 2.8; the other is to evaluate the number of paths that contain 85% of the total channel energy, as shown in Figure 2.9. The average numbers of significant paths are indicated by thick-dashed lines. Their values are: $\overline{NP_{10dB}} = 12.5$ and $\overline{NP_{85\%}} = 20.8$ for CM1, and $\overline{NP_{10dB}} = 24.9$, $\overline{NP_{85\%}} = 64.7$ for CM3.



(a) CM1

(b) CM3

Figure 2.9: Number of significant paths that contain 85% of energy

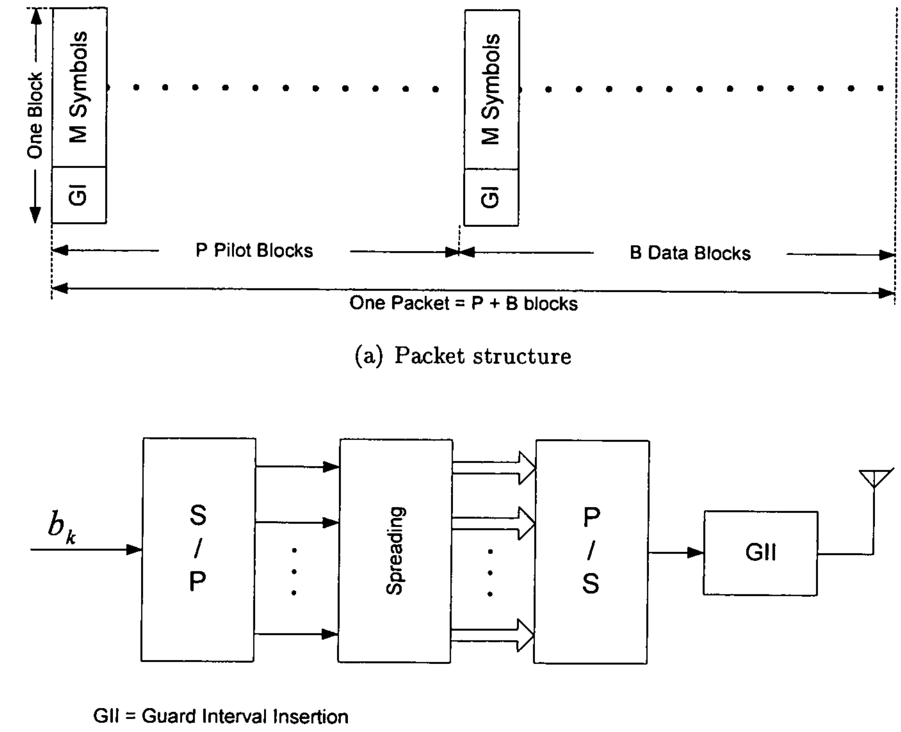
2.4 DS-UWB System Model

The work in this study focuses mainly on the design of machine learning based single-user receivers for high data rate DS-UWB systems. This section describes the general structure of the DS-UWB system model that contains design of the transmitted and the received signals. The system model is used for the proposed machine learning based receivers in later chapters.

Under anticipated regulations, as mentioned in Section 2.1, UWB transmit power will be likely to be limited by the power spectral density (PSD) of the transmitted signal, affecting the choice of modulation in two ways. First, the modulation technique needs to be power efficient [40]. Second, the choice of a modulation scheme has effects on the structure of the PSD and thus has the potential to impose additional constraints on the total transmit power. DS-UWB suggests a reasonable choice of UWB system models. In DS-UWB modulation, a number of pulses, representing chips, are sent per bit duration. The chip pulse sequence corresponds to a short pseudo-random code sequence for the u^{th} user, analogous to code division multiple access (CDMA) [30].

2.4.1 Transmitted Signal

Figure 2.10 illustrates the overall transmitting system model of a binary DS-UWB system. The transmit signal is designed so that the information symbols $b_m \in \{-1, +1\}$ $(m = 1, ..., M)$, which are of unit energy, are arranged in blocks of length M by a serial-to-



(a) Packet structure

(b) Tranismitter structure

Figure 2.10: System model of the DS-UWB transmitter

parallel (S/P) converter for spreading by a ternary code. Then, a guard interval (GI) of L_{GI} zero symbols is prefixed to each block, in order to mitigate the effect of inter-block interference (IBI). A whole transmission session is represented by a packet, which consists of P pilot blocks for training and B data blocks for communicating. The packet structure is illustrated in Figure 2.10(a). The j^{th} extended (GI-inserted) block is expressed as $\mathbf{b}_{\mathrm{GI}}(j) = [\mathbf{0}^T_{L_{\mathrm{GI}}} b_1 ... b_M]^T$ where $\mathbf{0}_l$ denotes an all-zero column vector of length l. The discrete-time signal representation will be used throughout the work.

Considering a single user transmission, a ternary spreading code $\mathbf{S} = [s_1\, s_2\, ...\, s_{N_c}]^T$ is used, where $s_k \in \{-1, 0, 1\}$ $(k = 1, ..., N_c)$, and N_c is the spreading code length (in chips). The j^{th} block after spreading can be evaluated by

$$\mathbf{b}_{DS}(j) = \begin{bmatrix} \mathbf{S} & \mathbf{0}_{N_c} & \cdots & \mathbf{0}_{N_c} \\ \mathbf{0}_{N_c} & \mathbf{S} & & \vdots \\ \vdots & & \ddots & \\ \mathbf{0}_{N_c} & \cdots & & \mathbf{S} \end{bmatrix} \mathbf{b}_{GI}(j) \tag{2.12}$$

where the spreading matrix is of size $N_c(M + L_{GI}) \times (M + L_{GI})$, or in vector compact formulation, $\mathbf{b}_{DS}$ can be expressed as

$$\mathbf{b}_{\mathrm{DS}}(j) = \begin{bmatrix} \mathbf{0}^T_{N_c L_{\mathrm{GI}}} & b_1\mathbf{S}^T & \ldots & b_M\mathbf{S}^T \end{bmatrix}^T \tag{2.13}$$

2.4.2 Received Signal

For block-by-block transmission, the discrete-time form of the j^{th} received signal block can be expressed as

$$\mathbf{r}_{\mathrm{GI}}(j) = \mathbf{H}\mathbf{b}_{\mathrm{DS}}(j) + \mathbf{v}(j) \tag{2.14}$$

where $\mathbf{H}$ is the UWB channel matrix defined in Equation (2.11), $\mathbf{v}(j)$ is an additive white Gaussian noise (AWGN) vector whose elements are independent Gaussian random variables with zero mean and variance σ^2. The GI samples (the first $N_c L_{\mathrm{GI}}$ chips of each block) are then removed from $\mathbf{r}_{\mathrm{GI}}$ so that the j^{th} received signal can be defined as

$$\mathbf{r} = [r_{\mathrm{GI}}\,[N_c L_{\mathrm{GI}} + 1] \ldots r_{\mathrm{GI}}\,[N_c(L_{\mathrm{GI}} + M)]]^T \tag{2.15}$$

2.5 Overview of Receivers for DS-UWB Systems

This section presents and describes the conventional receivers for UWB systems in the literature. In particular, it discusses the problem of channel estimation and equalisation. The presentation comprises of two general subsections for the UWB channel estimation and equalisation, followed by detailed description and simulations of two conventional receivers for DS-UWB systems. The discussion of UWB channel equalisation techniques will mainly focus on DS-UWB, because of its similarity to other signaling format and its utilisation for the applications of machine learning algorithms in the rest of the research.

2.5.1 UWB Channel Estimation

In the equalisation process, *i.e.,* in order to compensate for the channel effects, a good knowledge of the channel impulse response is necessary at the receiver. Hence, an appropriate channel estimation method is very important, especially for UWB channels where when the CIR is long and large number of parameters have to be taken into account [19].

The impulse response estimators for UWB channels have been developed in [41] and [42] based on the maximum-likelihood (ML) criterion, The input-output channel identification algorithm in [41] uses a single transmitted pulse in the absence of MUI; whereas the approaches in [42] form impulse response estimates using either training symbols, referred to as data-aided (DA) or unknown information-conveying symbols, referred to as non data-aided (NDA). Both DA and NDA channel estimators are tested in [42] over a fixed channel with three multipath components. The formidably high sampling rates required by the UWB increases the computational complexity of the optimal ML estimators to a prohibitive complexity, as the number of multipath components increases. For instance, the number of parameters to be estimated, *i.e.*, the number of delays and amplitudes, can be as large as 400 for a typical UWB indoor channel.

For more practical indoor UWB channel estimators, however, suboptimal ML-based estimators are adopted. The sliding window (SW) or sliding correlator, with the help of known pilot symbols, is widely used in the literature (*e.g.,* [43] and [44]), where the algorithm cross-correlates the received pilot signal with the transmitted known pilot in order to calculate channel gains and delays.

Another suboptimal estimator is the successive channel estimator (SC) which has originally been proposed for DS-CDMA systems in [45]. In the SC algorithm, the SW is used in an iterative manner to search for the strongest path, then a delayed version of signal is subtracted from the received signal accordingly, and so on until the number of assigned taps of the estimator is evaluated.

Furthermore, the characteristics of the frequency response of the UWB channel can also be estimated using the frequency-domain (FD) channel estimation methods. FD channel estimators show fast convergence and lower complexity. To this end, a recursive least squares (RLS) algorithm has been adopted in [46] to independently operates over channel frequency bins. More recently, an FD based channel estimator has been proposed in [47] by deriving a lower MSE bound for the linear minimum mean squared error (LMMSE).

2.5.2 Channel Equalisation for DS-UWB Systems

In general, most digital communication channels can be represented by bandlimited filters with an impulse response of $h_{tr}(t)$ and a frequency response of $H_{tr}(f)$. Hence, the transmission is degraded by the channel. The intersymbol interference (ISI) can be considered as one of the main distortions of broadband wireless channels [48]. In UWB systems, ISI is mostly occurred due to the overlapping among the transmitting frames of pulses.

Channel equalisation for wireless communication systems

In the context of wireless communications, equalisation compensates for the ISI caused by time dispersive channels [34]. In a broad sense, equalisers can be categorised into two main types: linear and nonlinear equalisers. A linear equaliser is typically implemented as a finite impulse response (FIR) filter, called the feedforward filter, in which the current and past values of the received signal are linearly weighted by the filter coefficients and summed to produce the equaliser output. A nonlinear equaliser known as the decision feedback equaliser (DFE) consists of a feedforward filter and a feedback filter. Both the linear equaliser and DFE can be implemented either in the transversal or lattice structure [34].

Linear equalisers are widely used in modern communication systems. However, they are less effective in the high-ISI environments where the channel distortion is too severe. Two designs are commonly used in linear equalisation, and they are the zero-forcing solution [49] and the minimum mean square error (MMSE) solution [48]. The zero-forcing design is effective in combating ISI but suffere from a serious noise enhancement problem. The MMSE design is often preferred in practice.

On the other hand, the nonlinear equalisers can perform well on channels with much severe time dispersion where their spectrals have deep nulls in the passband [48]. Therefore, nonlinear equalisers are mostly used in practical wireless communication systems. Three main methods have been developed for nonlinear equalisers, including decision feedback equalisation (DFE), maximum likelihood symbol detection [34] and maximum likelihood sequence estimation (MLSE) [49].

UWB channel equalisation

Due to the very dispersive effects of UWB channels, it can be arguably stated that the issues of UWB channel estimation and equalisation are the most challenging aspects in designing and implementing the physical layer for UWB systems. Many researchers, therefore, have proposed different techniques and strategies for this purpose.

The mechanism of energy capturing and ISI combating of a multipath fading channels was adopted via RAKE receivers. RAKE receivers were employed in [50] based on decorrelation effect. In [51], a combined RAKE and an MMSE equaliser structure was proposed for the UWB systems. The RAKE receiver concatenated with the MMSE equaliser was also constructed in [52] for the DS-UWB system. Different versions of matched-filter bound (MFB) were derived for DS-UWB in [53], where a BPSK modulation with the square-root raised cosine (RC) was proposed. A tap selection method using the matching pursuit (MP) algorithm with quadratic constraint was proposed in [54].

More recently, some other techniques were proposed for DS-UWB equalisation such as the combination of received response sequence at the transmitter and matched filter-equaliser-RAKE [55]. Also, spatial diversity schemes were considered in DS-UWB equalisation as proposed in [56], or jointly with pre-equalisation and pre-RAKE [57].

Because of the receiver complexity issues for the mentioned time-domain equalisers, some FDE based techniques were proposed, including the DS-UWB system with MMSE-FDE in [58], channel estimation and equalisation in the FD in [44], and the FD turbo equalisation in [59].

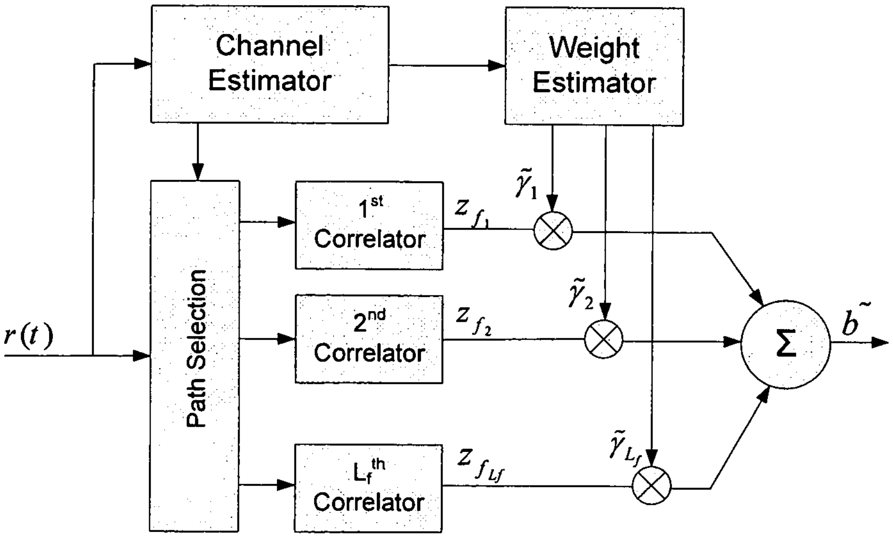


Figure 2.11: RAKE receiver structure for DS-UWB.

2.5.3 RAKE Receiver for DS-UWB Systems

As mentioned in Subsection 2.5.2, the RAKE receiver demodulator is conventionally used to detect DS-UWB signals, due to its simple implementation. Typically, for low symbol rates, it can be used for multiuser detection (MUD), and, for high symbol rates, it can be used to mitigate the frequency-selective channels' effect. This subsection briefly describes the structure and functionality of RAKE receiver. Detailed discussion on RAKE receiver can be found in [44], [60].

The general structure of RAKE receiver is illustrated in Figure 2.11. It consists of L_f correlators followed by a RAKE combiner. The reference signature, $s_{ref}(t)$, is the DS code signature of the user. Each correlator correlates the received signal with the reference signature at the delay times τ_{f_l} , and integrates over one symbol duration (T_f). The l^{th} correlator output $z^j_{f_l}$ for the j^{th} desired symbol is given by:

$$z^j_{f_l} = \int_{(j-1)T_f+\tau_{f_l}}^{jT_f+\tau_{f_l}} r(t)s_{ref}(t - jT_f - \tau_{f_l})dt \tag{2.16}$$

where τ_{f_l} is the delay time of the l^{th} path within one symbol duration.

Different combining criteria are used to select the number of RAKE fingers, which is normally less than the number of channel taps. The most common types of RAKE receivers includes all RAKE (ARAKE), partial RAKE (PRAKE), and selective RAKE

(SRAKE) [61]. However, RAKE receivers require good channel estimation technique in order to obtain a satisfactory performance.

RAKE receiver simulation

Here, the performance of the conventional RAKE receiver [44][60] is investigated and examined for DS-UWB systems. The maximal ratio combining (MRC) SRAKE receiver [61] is used. In MRC SRAKE (denoted by RAKE for convenience), the RAKE fingers corresponding to the L_f strongest estimated path gains with the delay times τ_{f_l} $(l = 1, 2, ..., L_f)$ are selected.

Assuming binary transmission scheme and perfect chip synchronisation between the transmitter and the receiver, the l^{th} correlator (finger) output $z^j_{f_l}$ for the j^{th} desired symbol, from a discrete received signal from Equation (2.15), is given by:

$$z^{f_l}_j = \sum_{i=1}^{N_c} s_i \mathbf{r}[(j-1)N_c + i + \tau_{f_l}] \tag{2.17}$$

where τ_{f_l} is the delay time of the l^{th} strongest path gain within one symbol duration. The combined output of the L_f-finger RAKE receiver for the j^{th} symbol can be expressed as follows:

$$\tilde{b}_j = \tilde{\gamma}^T \mathbf{z}_j \tag{2.18}$$

where $\mathbf{z}_j = [z^{f_1}_j, ..., z^{f_{L_f}}_j]^T$, and $\tilde{\gamma} = [\tilde{\gamma}_1, ..., \tilde{\gamma}_{L_f}]^T$ is the finger weight vector of the RAKE receiver. Based on the MRC criterion, $\tilde{\gamma}_l$ is given by

$$\tilde{\gamma}_l = \hat{h}_{f_l} \tag{2.19}$$

where $\hat{h}_l$ is the estimate of the l^{th} strongest path gain. The estimated binary symbol is then determined by the decision function as

$$\hat{b}_j = \text{sign}(\tilde{b}_j) = \begin{cases} +1 & , \tilde{b}_j \geq 0 \\ -1 & , \tilde{b}_j < 0 \end{cases} \tag{2.20}$$

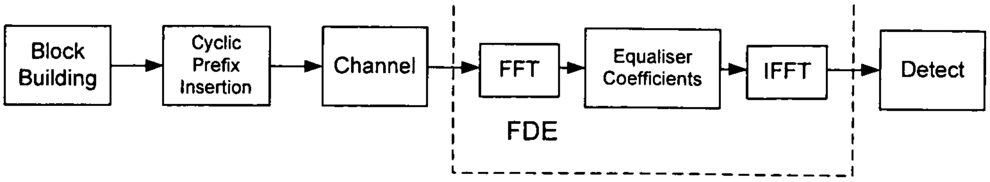


Figure 2.12: A communication system with FDE

Data-aided channel estimation for RAKE receiver

A data-aided (DA) approach [42] is used to estimate the channel impulse response in this simulation. The general sliding correlator method [51, 43] is employed for channel estimation. This is accomplished by sending P known pilot symbols b_j^t ($j = 1, 2, ..., P$) for training. The RAKE receiver, during the training, gives the output signal vector $\mathbf{z}_j^{L_{\text{est}}} = [z_j^1, z_j^2 ..., z_j^{L_{\text{est}}}]^T$ where L_{est} is the number of paths to be estimated, and it is assumed that the receiver knows the optimal value of L_{est}, *i.e.*, $L_{\text{est}} = L$ [62]. By applying the cross-correlation method, the estimated path gains in the form of vector of the channel ($\hat{\mathbf{h}}$) can be expressed as follows:

$$\hat{\mathbf{h}} = \frac{1}{P} \sum_{j=1}^{P} b_j^t \mathbf{z}_j^{L_{\text{est}}} \tag{2.21}$$

where $\hat{\mathbf{h}} = [\hat{h}_1, \hat{h}_2, ..., \hat{h}_{L_{\text{est}}}]^T$.

2.5.4 Frequency Domain Equaliser for DS-UWB System

FDE is simply the frequency representation analogy of what is done by a conventional time domain equalisers. For channels with severe delay spread, FDE is computationally simpler than the corresponding time-domain equalisation, because it is performed on a block of data at a time, and the operations on this block involve an efficient FFT operation and a simple channel inversion operation [63]. A cyclic prefix (CP) insertion technique has been proposed [64], jointly with FFT processing, to make convolutions appear circular and to avoid the channel time dispersion effects. Figure 2.12 illustrates the general structure of a communication system that employs linear FDE.

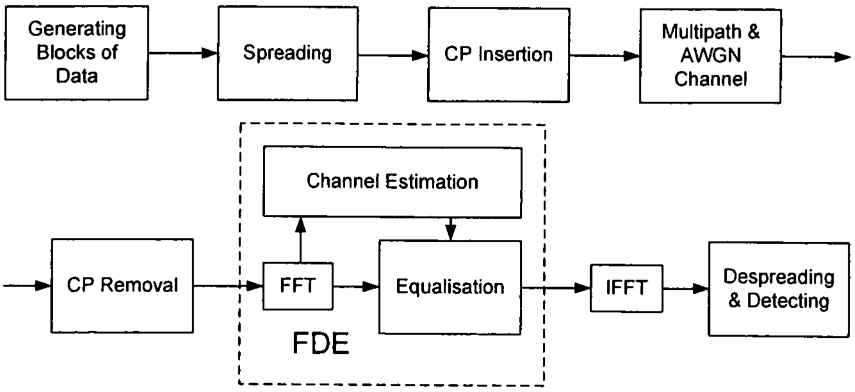


Figure 2.13: FDE for DS-UWB system design

The CP insertion is accomplished by repeating the last L_{CP} symbols in a given block of symbols and pre-appending them to the header of this block. FDE is regarded as a low-complexity method of reducing the ISI resulted from the multipath environments.

FDE receiver simulation

A simulation has been implemented to examine the FDE performance for DS-UWB systems. The over all system is illustrated in Figure 2.13, where data are generated by random uniform binary bits in blocks and spreaded by an arbitrary ternary code [32]. Then, a CP is added to the head of each block to generate the transmitted signal. The received signal is passed through an FFT block after removing the CP symbols, then the frequency components are multiplied by the equaliser coefficients from the channel information, to produce an FD signal. The FD equalised signals are transferred back into the time domain by IFFT, and are then despread and passed through a decision device to recover the transmit symbols.

Assuming perfect chip synchronisation, the received samples can be expressed as

$$r(n) = s(n) \otimes h(n) + n(n), \tag{2.22}$$

where $\otimes$ represents the convolution operation.

After removing the CP samples, and applying the FFT, the frequency components of the received signals are expressed as

$$R(k) = H(k)S(k) + N(k) \tag{2.23}$$

where $R(k)$, $H(k)$, $S(k)$ and $N(k)$ are the FFT of the received signal, the CIR, the transmitted signal and the AWGN, respectively.

Frequency domain LS channel estimation for FDE

A simple least squares (LS) channel estimation technique can be used to estimate $H(k)$ for binary signalling. The estimation can be achieved by transmitting a pilot block prior to data blocks. This pilot is a known block of symbols that are used to estimate the channel in the training mode as following

$$\hat{H}(k) = \frac{R^t(k)}{S^t(k)} \tag{2.24}$$

where $S^t(k)$ is the discrete Fourier transform of the transmitted pilot signal, and $R^t(k)$ is the received pilot in frequency representation. Hence, the channel estimate $\hat{H}(k)$ is then applied at the equaliser to restore the original signal by

$$\hat{S}(k) = R(k)\hat{H}^*(k). \tag{2.25}$$

where $\hat{H} * (k)$ is the complex conjugate of $\hat{H}(k)$.

2.5.5 Simulation Results

This subsection presents the simulation results of the RAKE receiver and the FDE. The simulation setup is similar to that in later chapters for consistency. A channel model 3 of IEEE802.15.3a standard (CM3) is considered, and a pilot size of 200 symbols is employed for channel estimation parts. For RAKE receiver simulations, the number of fingers L_f considered is 200. The simulations were tested on a range of SNR levels of $0 - 25dB$. The BER performance of the two receivers with both perfect CSI and the channel estimation (CE) schemes described in Subsections 2.5.3 and 2.5.4, is shown in Figure 2.14.

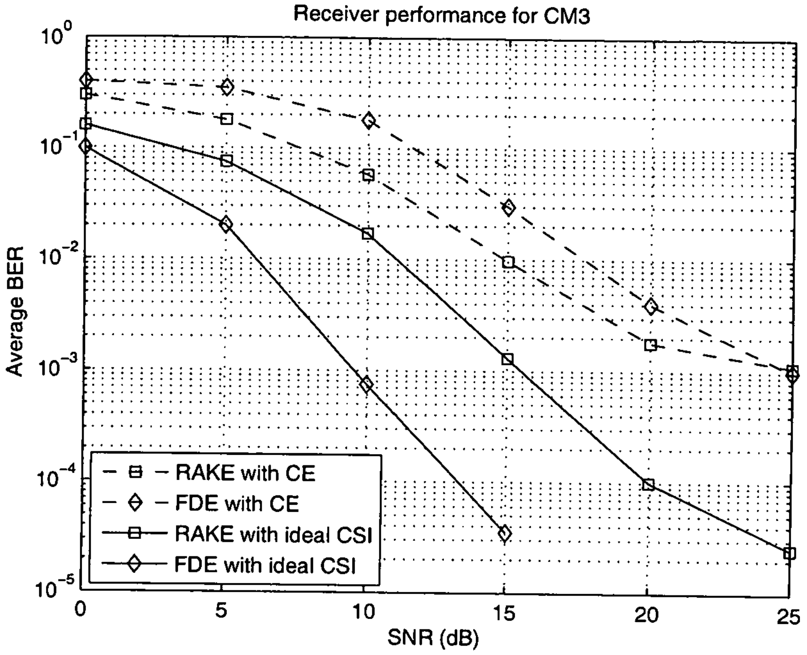


Figure 2.14: DS-UWB system performance with conventional receivers ($L_f = 200$ for RAKE receiver)

It is precisely expected that the system performance is better when ideal channel is assumed. This is true even comparing with ideal RAKE receiver, because of the limited number of RAKE fingers used in these simulations compared to the number of UWB channel paths. Also it is noticed that the FDE system performance dramatically improves at high SNRs, this is because of neglecting the noise power in Equation (2.24). Whereas the RAKE receiver with CE outperforms the FDE at low-to-medium SNR levels.

The choice of pilots plays an important role in channel estimation, especially in the FD channel estimation. This is because of that the null frequency components, if any, lead to unreliable estimate in Equation (2.24). In these simulations, an impluse-like pilot is considered representing almost a flat frequncy response.

2.6 Summary

In this chapter, a general overview of the UWB system has been presented in the context of its main signal processing aspects. The presentation starts by describing the definition of UWB signalling, its major pulse waveforms, and the common multiple access schemes of these technologies. Then, system model of the DS-UWB systems investigated in this thesis is presented, including the transmitted and the received signals formulation.

More emphasis is being laid on the current literature of the issues of representing and tackling UWB severe channel degradation; such as channel modeling, estimation, and equalisation. In particular, the RAKE receiver and the FDE based receiver are described and discussed as conventional receivers for DS-UWB systems. Some basic simulations of conventional UWB channel equalisation which will be used as a reference for the proposed equalisers in subsequent chapters.

Chapter 3

Machine Learning Algorithms

This chapter presents a general overview on the ever-growing area of machine learning techniques and algorithms. In fact, machine learning is quite a broad subject with loads of information and contributions, hence, it is beyond the scope of this chapter to cover all the aspects of machine learning. Rather, the necessary basics and fundamental principles, that relate to the work in this research, are considered in this presentation. The organisation of this chapter is as follows. An introduction is provided in Section 3.1. Section 3.2 considers linear models in classification followed by kernel-induced methods for nonlinearly separable patterns in Section 3.3. The statistical learning methods are illustrated in Section 3.4, and their promising finding of support vector machines are described in Section 3.5. Subsequently, the issue of model selection is presented in Section 3.6. Section 3.7 is concerned with the application of Bayesian inference in the context of machine learning. Eventually, a chapter summary is provided in Section 3.8.

3.1 Introduction

The field of machine learning was conceived nearly five decades ago with the bold objective to develop computational methods that would implement various forms of learning [8], in particular algorithms capable of inducing knowledge from examples or data, *i.e.*, via training. Gaining such knowledge automatically is particularly desirable in many problems in real world applications in fields such as medical diagnoses, engineering and computer design, as will be shown soon. Figure 3.1 depicts the general task of a

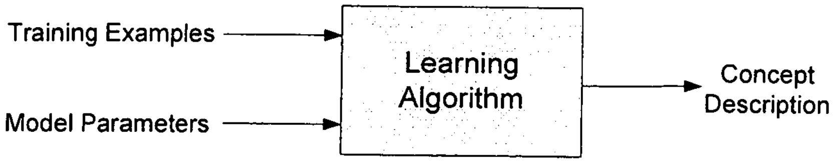


Figure 3.1: The general framework of machine learning

learning machine.

Machine learning is, by nature, a multidisciplinary field [8, 9, 65]. Its algorithms represent results drawn from many disciplines such as philosophy, mathematics, statistics and probability theory, data mining, signal processing, optimisation theory, computational complexity theory, information theory, and artificial intelligence, and many other disciplines.

Types of learning algorithms

The learning tasks can be categorised into [9]: Probability distribution estimation, pattern association (*e.g.*, clustering), pattern recognition (*e.g.*, classification), function approximation (*e.g.*, regression), beamforming, and control. The learning processes, on the other hand, are widely classified into the following three categories:

- Supervised learning, which requires the availability of a target or desired response for the realisation of specific input-output mapping by minimising a cost function of interest.
- Unsupervised learning [66], the implementation of which relies on the provision of a task-independent measure of the quality of representation that the network is required to learn in a self-organised manner.
- Reinforcement learning [9], in which input-output mapping is performed through the continued interaction of a learning system with its environment so as to minimise a scalar performance index.

The work in this PhD study applies the supervised learning in the problem of channel equalisation by mapping the problem domain into a pattern classification one [67].

Therefore, the rest of this overview will treat the subject of machine learning from supervised-pattern-recognition point of view.

Applications of machine learning

As mentioned in the introduction of this section, machine learning has a wide spectrum of applications that make it increasingly desirable. Among many applications to consider, some include handwritten recognition, search engines, medical diagnosis, biometric recognition (*e.g.*, face and fingerprint), stock market analysis, load forecasting in power systems, machine games, natural language processing, image processing, and DNA sequence classification.

3.2 Linear Models for Classifications

One of the most important learning tasks in pattern recognition, and is the core of this study's application, is the class of models for solving classification problems. The goal in classification is to take an input vector $\mathbf{x}$ and to assign it to a target y from one of K discrete classes C_k where $k = 1, ..., K$. In the most common scenario, the classes are taken to be disjoint, so that each input is assigned to one and only one class. The input space is thereby divided into decision regions whose boundaries are called decision boundaries or decision surfaces. Figure 3.2 shows a simple binary classification example, where a set of N training data points $\{\mathbf{x}_i, y_i\}$, $\mathbf{x} \in \mathbb{R}^2$ and $y \in \{-1, +1\}$, is passed through the classifier. The classifier's task is to produce the linear decision boundary defined by $\mathbf{w}^T\mathbf{x} + \beta = 0$, where $\mathbf{w}$, β are the classifier's parameters to be resulted from learning. And the discriminant function is then

$$f(\mathbf{x}) = \mathbf{w}^T\mathbf{x} + \beta \tag{3.1}$$

In the following subsections, the linear models for classification are considered, which means that the decision surfaces are linear functions of the input vector $\mathbf{x}$, as shown in the example, and hence are defined by $(d-1)$-dimensional hyperplanes within the d-dimensional input space. Data sets whose classes can be separated exactly by linear

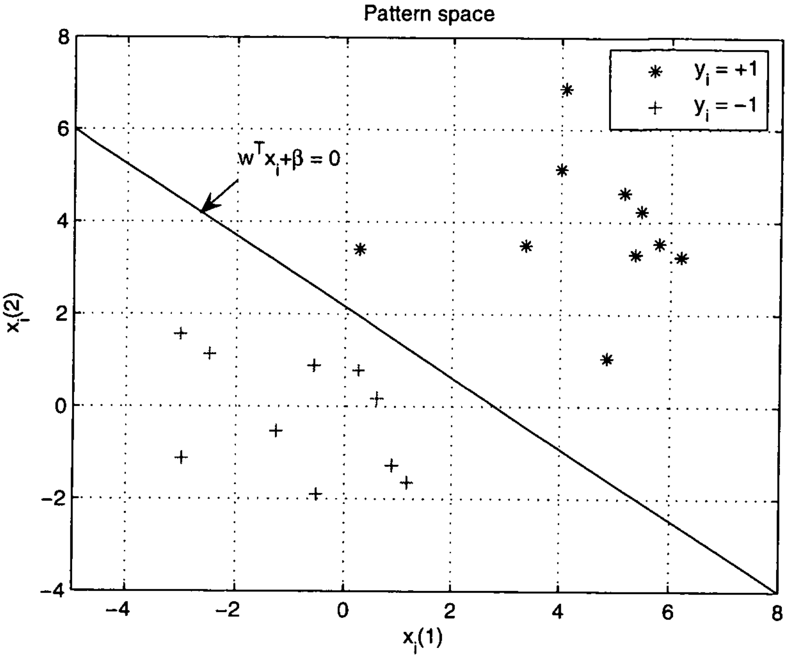


Figure 3.2: General linear model for binary classification with a decision boundary (solid line)

decision surfaces are said to be linearly separable. Others, of course, are said to be nonlinearly separable datasets. The classification models are categorised into three main types as follows.

3.2.1 Deterministic Models

In these models, the parameters are evaluated in a 'single-shot' manner, so that only one optimum coefficients vector is obtained. Common examples are presented next.

LS methods for linear classification

LS methods are matured techniques that are widely used in parameter estimation problems for many applications [66]. They consider the models that are linear functions of the model parameters, and by minimising the sum-of-squares of some error function, a simple closed-form solution for the parameter values can be obtained. The same formalism can be applied to classification problems. Consider a general classification problem with K classes, with a binary coding scheme for the target vector $\mathbf{y} = [\mathbf{y}_1, ..., \mathbf{y}_\mathrm{K}]^\mathrm{T}$.

Each class C_k is described by its own linear discriminant model so that

$$f_k(\mathbf{x}) = \mathbf{w}_k^T\mathbf{x} + \beta_k \tag{3.2}$$

where $k = 1, ..., K$. These models can be conveniently group together using vector-matrix notation so that

$$\mathbf{f}(\mathbf{x}) = \mathbf{W}^T\tilde{\mathbf{x}} \tag{3.3}$$

where $\mathbf{W}$ is the $(d+1) \times K$ weight matrix whose k^{th} column comprises the $(d+1)$-dimensional vector $\tilde{\mathbf{w}}_k = (\beta_k, \mathbf{w}_k^T)^T$, $\tilde{\mathbf{x}} = (1, \mathbf{x}^T)^T$ is the corresponding augmented input vector with a dummy input $x_0 = 1$, and $\mathbf{f} = (f_1, ..., f_k)^T$ is the discriminant model vector. A new input $\mathbf{x}$ is then assigned to the class k for which the output $f_k = \tilde{\mathbf{w}}_k^T\tilde{\mathbf{x}}$ is the largest. The parameter matrix $\mathbf{W}$ is determined by minimising a sum-of-squares error function. Consider a training data set $\mathcal{D} = \{\mathbf{x}_i, y_i\}$ where $i = 1, ..., N$, and define the $N \times K$ target matrix Y whose i^{th}row is $\mathbf{y}_i^T$, together with the $N \times (d+1)$ input matrix $\mathbf{X}$ whose i^{th} row is $\tilde{\mathbf{x}}_i^T = (1, \mathbf{x}_i^T)$. The sum-of-squares error function can then be written as

$$E_D(\mathbf{W}) = \frac{1}{2}\mathrm{Tr}\left\{(\mathbf{XW} - \mathbf{Y})^T(\mathbf{XW} - \mathbf{Y})\right\}, \tag{3.4}$$

where $\mathrm{Tr}\{\}$ denotes the matrix trace operator. Setting the derivative of $E_D(\mathbf{W})$ with respect to $\mathbf{W}$ to zero leads to the least squares solution $\mathbf{W}_{LS}$ given by the form

$$\mathbf{W}_{LS} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y} = \tilde{\mathbf{X}}\mathbf{Y} \tag{3.5}$$

where $\tilde{\mathbf{X}}$ is the pseudo-inverse of the matrix $\mathbf{X}$. The discriminant function is then obtained in the form

$$\mathbf{f}_{LS}(\mathbf{x}) = \mathbf{W}_{LS}^T\tilde{\mathbf{x}} = \mathbf{Y}^T(\tilde{\mathbf{X}})^T\tilde{\mathbf{x}}. \tag{3.6}$$

The LS approach gives an exact closed-form solution for the discriminant function parameters. However, even as a discriminant function, which is a single parameter set, it suffers from some severe problems. Most importantly, the lack of robustness to outliers to the classification application. The outliers sensitivity means the change in the location of the decision boundary according to some additional data points that are relatively far from their cloud of data class.

The sum-of-squares error function penalises predictions that are 'too correct' in that they lie a long way on the correct side of the decision boundary [68]. However, problems with LS can be more severe than simply lack of robustness. That is, having the property that linear decision boundaries can give excellent separation between the classes, the LS solution gives poor results for multiclass problems, with only a small region of the input space assigned to the corresponding class. Indeed, the technique of logistic regression, described later in this section, gives a satisfactory solution.

Fisher's linear discriminant

This discriminant is widely used for binary (and can be extended to multiclass) linear classification [69]. The main idea behind it is to reduce the dimension of the problem so that a decision threshold can be estimated by maximising the distances between the class means and minimising the class variances. In other words, Fisher linear discriminant finds a linear projection such that the classes are well separated. Separability is measured by two quantities: maximising the interclass difference and minimising the intraclass spread. This confirms the reduction of interclass overlapping. Hence, the Fisher's criterion for two-class problem is to maximise

$$J(\mathbf{w}) = \frac{\mathbf{w}^T \mathbf{S_B} \mathbf{w}}{\mathbf{w}^T \mathbf{S_W} \mathbf{w}} \tag{3.7}$$

where $\mathbf{w}$ is the hyperplane coefficient vector. $\mathbf{S_B}$ in Equation (3.7) is the between-class covariance matrix, defined by

$$\mathrm{S_B} = (\boldsymbol{\mu}_2 - \boldsymbol{\mu}_1)(\boldsymbol{\mu}_2 - \boldsymbol{\mu}_1)^T, \tag{3.8}$$

where $\boldsymbol{\mu}_1$and $\boldsymbol{\mu}_2$ are the means of the classes C_1 and C_2, respectively. $\mathrm{S_W}$ in Equation (3.7) is the total within-class covariance matrix, defined by

$$\mathrm{S_W} = \sum_{i \in C_1} (\mathrm{x}_i - \boldsymbol{\mu}_1)(\mathrm{x}_i - \boldsymbol{\mu}_1)^T + \sum_{i \in C_2} (\mathrm{x}_i - \boldsymbol{\mu}_2)(\mathrm{x}_i - \boldsymbol{\mu}_2)^T \tag{3.9}$$

By differentiating Equation (3.7) with respect to $\mathbf{w}$, the maximum of $J(\mathbf{w})$ is found when

$$\mathbf{S}_\mathrm{B}\mathbf{w} = \lambda \mathbf{S}_\mathrm{W}\mathbf{w} \tag{3.10}$$

where λ is a scaling factor. Recognising that the matrix product $\mathbf{S}_\mathrm{B}\mathbf{w}$ is always in the direction of the difference vector $\boldsymbol{\mu}_2 - \boldsymbol{\mu}_1$, the solution of Equation (3.10) is simply

$$\mathbf{w} = \mathbf{S}_\mathrm{W}^{-1}(\boldsymbol{\mu}_2 - \boldsymbol{\mu}_1). \tag{3.11}$$

Equation (3.11) is known as Fisher's linear discriminant, although strictly it is not a discriminant but rather a specific choice of direction for projection of the data down to one dimension. However, the projected data can subsequently be used to construct a discriminant, by choosing a threshold f_0 so that we classify a new point $\mathbf{x}$ as belonging to C_1 if $f(\mathbf{x}) = \mathbf{w}^T\mathbf{x} \geq f_0$ and classify it as belonging to C_2 otherwise.

3.2.2 Iteration based Learning Models

In this type of models, the evaluation process is conducted in an iterative manner. Rosenplatt's perceptron is the common pioneered example.

Rosenplatt's perceptron

The first iterative algorithm for learning linear classifications is the procedure proposed by Rosenblatt [70] for the perceptron. The algorithm created a great deal of interest when it was first introduced. It is an 'on-line' and 'mistake-driven' procedure, which starts with an initial weight vector $\mathbf{w}_0$ (usually $\mathbf{w}_0 = \mathbf{0}$, the all zero vector) and adapts it each time a training point is misclassified by the current weights. The algorithm is shown in Table 3.1.

The algorithm updates the weight vector and bias directly. This procedure is guaranteed to converge provided there exists a hyperplane that correctly classifies the training data. In this case the data are perfectly linearly separable. If no such hyperplane exists the data are said to be linearly nonseparable.

The number of iterations in Rosenplatt's procedure depends on a quantity called the margin. This quantity will play a central role in the majority of the techniques used

```
Given a linearly separable training set $S = \{\mathbf{x}_n, y_n\}_{n=1}^{N}$ and learning rate $\eta \in \mathbb{R}^+$.
$\mathbf{w}_0 \leftarrow \mathbf{0}$; $b_0 \leftarrow 0$; $k \leftarrow 0$
$R \leftarrow \max_{1 \leq n \leq N} \|\mathbf{x}_n\|$
repeat
    for $n = 1$ to $N$
        if $y_n\{\mathbf{w}_k^T \mathbf{x}_n + b_k\} \leq 0$ then
            $\mathbf{w}_{k+1} \leftarrow \mathbf{w}_k + \eta y_n \mathbf{x}_n$
            $b_{k+1} \leftarrow b_k + \eta y_n R^2$
            $k \leftarrow k + 1$
        end if
    end for
until no mistakes made within the for loop
return $(\mathbf{w}_k, b_k)$ where $k$ is the number of mistakes
```

Table 3.1: The perceptron algorithm [71]

in this research and so some formal definition is emphasised.

The (functional) margin of an example $(\mathbf{x}_i, y_i)$ with respect to a hyperplane $(\mathbf{w}, \beta)$ is defined to be as the quantity in Equation (3.1). Note that $f(\mathbf{x}) \geq 0$ implies correct classification of $(\mathbf{x}_i, y_i)$. The margin distribution of a hyperplane $(\mathbf{w}, \beta)$ with respect to a training set $\mathcal{D}$ is the distribution of the margins of the examples in $\mathcal{D}$. The minimum of the margin distribution is sometimes referred to as the (functional) margin of a hyperplane $(\mathbf{w}, \beta)$ with respect to a training set $\mathcal{D}$. In both definitions if the functional margin is replaced by geometric margin we obtain the equivalent quantity for the normalised linear function $(\frac{1}{\|\mathbf{w}\|}\mathbf{w}, \frac{1}{\|\mathbf{w}\|}\beta)$, which therefore measures the Euclidean distances of the points from the decision boundary in the input space. Finally, the margin of a training set $\mathcal{D}$ is the maximum geometric margin over all hyperplanes. A hyperplane realising this maximum is known as a maximal margin hyperplane, and will be discussed in further detail in Section 3.5. The size of its margin will be positive for a linearly separable training set.

3.2.3 Probabilistic Models

The resulting model parameters in this category are obtained by their probabilistic distributions. This provides a better measurement of the uncertainty degrees to the estimation. Common examples are as follows.

Naive Bayes classifier

A naive Bayes classifier (NBC) [72] is a simple probabilistic classifier by applying the Bayes' theorem with strong (naive) independence assumptions. Let $p(C_k)$ be the probability of occurrence of class C_k, $k = 1, 2, ..., K$. $p(C_k)$ is known as *a priori* probability. A posterior probability that an observed sample $\mathbf{x}$ came from class C_k is expressed as $p(C_k|\mathbf{x})$. According to Bayes rule,

$$p(C_k|\mathbf{x}) = \frac{p(\mathbf{x}|C_k)p(C_k)}{p(\mathbf{x})}, \tag{3.12}$$

where $p(\mathbf{x})$ is the unconditional probability density function of $\mathbf{x}$, and $p(\mathbf{x}|C_k)$ is called the likelihood function of class C_k.The NBC simply assumes features are independent given the class C_k, that is $p(\mathbf{x}|C_k) = \prod_{i=1}^{N} p(\mathbf{x}_i|C_i)$. Thus

$$p(C_k|\mathbf{x}) = \frac{1}{p(\mathbf{x})} p(C_k) \prod_{i=1}^{N} p(x_i|C_k), \tag{3.13}$$

where $p(\mathbf{x})$ is a scaling factor dependent only on $\mathbf{x}$, *i.e.*, a constant. Models of the form in Equation (3.13) are much more manageable, since the independent probability distributions $p(x_i|C_k)$ allows the model parameters to be approximated from the training samples, in addtion to the only factor of the class prior $p(C_k)$. The decision function of the Bayes classifier is given as

$$f(\mathbf{x}) = \arg\max_k p(C_k) \prod_{i=1}^{N} p(x_i|C_k) \tag{3.14}$$

The decoupling of the class conditional feature distributions means that each distribution can be independently estimated as a one dimensional distribution. This in turn helps to alleviate problems stemming from the curse of dimensionality, such as the need for data sets that scale exponentially with the number of features.

Logistic regression

As mentioned in Subsection 3.2.1, Logistic regression [73] is an effective alternative for data patterns with separated outliers. Logistic regression estimates the probability of occurrence of an event. In order to explain logistic linear model, consider the problem of binary classification. The posterior probability of class C_1 can be written as a logistic sigmoid acting on a linear function of the data vector $\mathbf{x}$ so that

$$p(C_1|\mathbf{x}) = f(\mathbf{x}) = \sigma(\mathbf{w}^T\mathbf{x}) \tag{3.15}$$

with $p(C_2|\mathbf{x}) = 1 - p(C_1|\mathbf{x})$. Here $\sigma(.)$ is the logistic sigmoid function defined by

$$\sigma(\lambda) = \frac{1}{1+\exp(-\lambda)}. \tag{3.16}$$

This model is known, in statistics, as logistic regression, although it should be emphasised that this is a model for classification not regression. For a d-dimensional input space of data, this model has d adjustable parameters only, unlike that when using maximum likelihood for the Gaussian class conditional densities, where a $2d$ adjustable parameters are used for the means and $d(d+1)/2$ parameters for the common covariance matrix. Together with the class prior $p(C_1)$, this gives a total of $d(d+5)/2+1$ parameters, which grows quadratically with d. Therefore, there is a clear advantage in working with the logistic regression model directly, especially for large values of d.

To determine the parameters of the logistic regression model, the maximum likelihood [70] is used combined with the derivative of the logistic sigmoid function that is defined in Equation (3.16), which can be conveniently expressed in terms of the sigmoid function itself as follows

$$\frac{\mathrm{d}\sigma}{\mathrm{d}\lambda} = \sigma(1-\sigma). \tag{3.17}$$

For a data set $\{\mathbf{x}_i, y_i\}$, where $y_i \in \{0, 1\}$ [1], with $i = 1, ..., N$, the likelihood function can be written

$$p(\mathbf{y}|\mathbf{w}) = \prod_{i=1}^{N} f_i^{y_i}(1-f_i)^{1-y_i} \tag{3.18}$$

where $\mathbf{y} = (y_1, ..., y_N)^T$ and $f_i = p(C_1|\mathbf{x}_i)$. An error function can be defined by taking

[1] This encoding scheme is usually used for probabilistic representation.

the negative logarithm of the likelihood, which gives the form

$$E(\mathbf{w}) = -\ln p(\mathbf{y}|\mathbf{w}) = -\sum_{i=1}^{N}\{y_i \ln f_i + (1-y_i)\ln(1-f_i)\} \qquad (3.19)$$

where $f_i = \sigma(\lambda_i)$ and $\lambda_i = \mathbf{w}^T\mathbf{x}_i$. The quantity in Equation (3.19) is sometimes referred to as cross-entropy error function [69].

Taking the gradient of the error function with respect to $\mathbf{w}$, and using Equation (3.17) yield

$$\nabla E(\mathbf{w}) = \sum_{i=1}^{N}(f_i - y_i)\mathbf{x}_i. \qquad (3.20)$$

The contribution to the gradient from the i^{th} data point is given by the 'error' $(f_i - y_i)$ between the target value and the prediction of the model, times the training data vector $\mathbf{x}_i$.

It is worth noting that maximum likelihood can exhibit severe over-fitting for data sets that are linearly separable. This arises because the maximum likelihood solution occurs when the hyperplane corresponding to $\sigma = 0.5$, equivalent to $\mathbf{w}^T\mathbf{x} = 0$, separates the two classes and the magnitude of $\mathbf{w}$ goes to infinity. Furthermore, the maximum likelihood depends on the choice of optimisation algorithm and on the parameter initialisation. Note that the problem will arise even if the number of data points is large compared with the number of parameters in the model, so long as the training data set is linearly separable. The singularity can be avoided by inclusion of a prior and finding a MAP solution for $\mathbf{w}$, or equivalently by adding a regularisation term to the error function.

3.3 Kernel-Induced Methods

Despite the fact that linear machines suggest solid and tractable theoretical grounds for the problem of classification, they are blamed for their computational power limitation [71]. Also, they lack the applicability for nonlinearly separable clouds of patterns, which is the case for most real-world applications. Kernel representations provide an alternative solution by projecting the patterns into a high dimensional feature space to

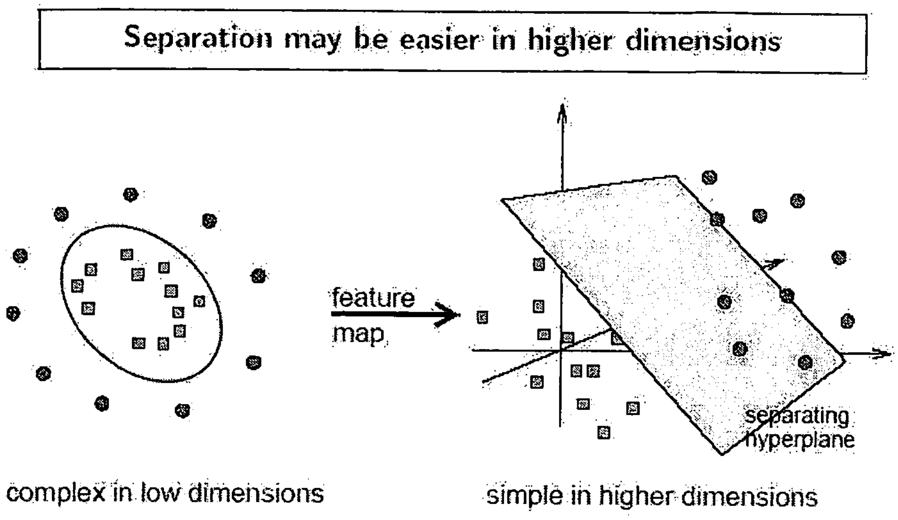

Separation may be easier in higher dimensions
feature map
separating hyperplane
complex in low dimensions
simple in higher dimensions

Figure 3.3: Mapping to high dimensional feature space [74]

increase the computational power of the linear learning machines of Section 3.2. In this section, the fundamental principles of kernel mapping are described as it is an important concept for nonlinear scenarios in which this work is highly dependent on.

3.3.1 Mapping to Higher Feature Space

Mapping the data onto another space called feature space is not new and time in machine learning [71]. The original data representations are sometimes called attributes or input space. Let the original input space be denoted by X. A mapping $\phi : \mathbf{x} \rightarrow \phi(\mathbf{x})$ is (implicitly) defined on X that maps it onto a higher-dimensional feature space $\mathcal{F} = \{\phi(\mathbf{x}) : \mathbf{x} \in X\}$. The advantage of doing so is that the project data onto the higher-dimensional feature space are more likely to be linearly separable and hence the classification task is greatly simplified.

Figure 3.3 shows an example of a feature mapping from a two dimensional input space to a three dimensional feature space. The data in this example cannot be separated by a linear function in the input space, but it is possible for the projected data in the feature space.

The aim of the subsequent subsections is to show how such mappings can be made into very high dimensional spaces where linear separation becomes more likely.

Cover's theorem

The underlying justification in mapping to a higher dimensional feature space is found in Cover's theorem [75] on the separability of patterns, which, in qualitative terms, is re-stated in [9] as follows:

> "A complex pattern-classification problem, cast in a high-dimensional space nonlinearly, is more likely to be linearly separable than in low-dimensional space, provided that the space is not densely populated."

Dual form of linear machines

In the previous discussion of linear models of classifications, the models and their optimisation formulation were in what is called 'primal form' [76]. The primal form is simply the form that comprises the hyperplane's parameters (*i.e.*, $\mathbf{w}$ and β) explicitly in the problem model. The dual representation [71], on the other hand, substitutes these parameters with a linear combination of the original attributes so that only the inner products of the original attributes appear in the form, not the explicit model parameters. An important property of the dual representation is that the data only appear through entries in the Gram matrix $\{\mathbf{G} : G_{i,j} = \mathbf{x}_i^T \mathbf{x}_j\}$ [77] and not through their individual attributes. Similarly in the dual representation of the decision function, it is only the inner products of the data with the new test point that are needed.

Most of the algorithms discussed in this study solve the optimisation problems for which a mathematical framework exists that naturally encompasses duality. An advantage of using the machines in the dual representation derives from the fact that in this representation the number of tunable parameters does not depend on the number of attributes being used. This representation will be used in subsequent subsections, and will be shown to be a general property of a wide class of algorithms. Duality is one of the crucial concepts in developing SVMs.

The kernel trick

A kernel function is an appropriately chosen function that computes the inner product of the feature vectors in the higher-dimensional feature space corresponding to the two inputs, so that one can implicitly perform a nonlinear mapping from the input space to a high-dimensional feature space without the need of explicitly defining the nonlinear mapping ϕ.

The idea of using kernel functions in machine learning was introduced by Aizerman in [78]. Kernel based methods in pattern analysis embed the data in a suitable feature space, and then use algorithms based on optimisation, linear algebra, geometry, and statistics to discover patterns in the embedded data. Two main procedures are conducted when applying any kernel based method: a procedure that performs the mapping into an empirical feature space $\mathcal{F}$, and a learning algorithm procedure designed to discover linear patterns in that space. A kernel function is the high-dimensional empirical feature space representation to the original data that ensures simple analysis. Four key aspects of kernel based machines are highlighted [77]:

- Input patterns are projected into the feature vector space.
- The patterns projections, in the feature space, are treated by Linear relations.
- Only the inner products of patterns projections are considered in implementing kernel based algorithms in such a way that a scalar value represents a pairwise of high-dimensional patterns projections.
- The pairwise inner products can be computed efficiently directly from the original input patterns using a kernel function.

Mercer theorem

In order for a kernel function to be an inner product kernel in some space, it has to satisfy Mercer's conditions that arise in the Mercer's theorem [79] of the functional analysis field. In order to briefly describe Mercer's theorem, let X be a compact subset of $\mathbb{R}^d$, and suppose that a continuous symmetric function that is defined on $X \times X$. The expansion in the series

$$K(\mathbf{x}, \mathbf{z}) = \sum_{j=1}^{\infty} \phi_j(\mathbf{x})\phi_j(\mathbf{z}), \tag{3.21}$$

in terms of functions ϕ_j, is said to be valid and uniformly convergent, if the necessary and sufficient condition that

$$\int_{X\times X} K(\mathbf{x},\mathbf{z})f(\mathbf{x})f(\mathbf{z})d\mathbf{z}d\mathbf{x} \geq 0, \tag{3.22}$$

holds for all $f(.)$, for which

$$\int_X f^2(\mathbf{x})d\mathbf{x} < \infty. \tag{3.23}$$

3.3.2 Constructing Kernels

Defining a kernel function for an input space is frequently more natural than creating a complicated feature space [71]. This subsection summarises the most important properties and principles to construct kernels with examples of commonly used forms of kernels.

Kernel properties

According to [77], a number of necessary properties of a function $K(\mathbf{x},\mathbf{z})$ is required to ensure that it is a kernel function for some feature space.

Firstly and obviously, the function must be symmetric, so that

$$K(\mathbf{x},\mathbf{z}) = \phi^T(\mathbf{x})\phi(\mathbf{z}) = \phi^T(\mathbf{z})\phi(\mathbf{x}) = K(\mathbf{z},\mathbf{x}). \tag{3.24}$$

Secondly, it satisfies the inequalities that follow from the Cauchy-Schwarz inequality, that is

$$\begin{aligned} K(\mathbf{x},\mathbf{z})^2 &= \left(\phi^T(\mathbf{x})\phi(\mathbf{z})\right)^2 \leq \|\phi(\mathbf{x})\|^2 \|\phi(\mathbf{z})\|^2 \\ &= \left(\phi^T(\mathbf{x})\phi(\mathbf{x})\right)\left(\phi^T(\mathbf{z})\phi(\mathbf{z})\right) = K(\mathbf{x},\mathbf{x})K(\mathbf{z},\mathbf{z}). \end{aligned} \tag{3.25}$$

These conditions are, however, not sufficient to guarantee the existence of a feature space.

Constructing kernels from kernels

Kernels can also be constructed from other simple kernels. This is due to the fact that a new symmetric function is a kernel if the condition in Equation (3.22) is satisfied and the matrix defined by restricting the function to any finite set of points is positive semi-definite [77].

The following proposition can be viewed as showing that kernels satisfy a number of closure properties, allowing the creation of more complicated kernels from simpler (prototype) forms.

Proposition Let $K_1$and K_2 be two kernels defined on $X \times X$, where $X \subseteq \mathbb{R}^d$. Suppose that $a \in \mathbb{R}^+$, $f(.)$ is a real-valued function on X, $f : X \rightarrow \mathbb{R}^m$, K_3 is a kernel on $\mathbb{R}^m \times \mathbb{R}^m$, and $\mathbf{B}$ a symmetric positive semi-definite $d \times d$ matrix. Then the following functions are kernels:

1. $K(\mathbf{x}, \mathbf{z}) = K_1(\mathbf{x}, \mathbf{z}) + K_2(\mathbf{x}, \mathbf{z})$,
2. $K(\mathbf{x}, \mathbf{z}) = aK_1(\mathbf{x}, \mathbf{z})$,
3. $K(\mathbf{x}, \mathbf{z}) = K_1(\mathbf{x}, \mathbf{z})K_2(\mathbf{x}, \mathbf{z})$,
4. $K(\mathbf{x}, \mathbf{z}) = f(\mathbf{x})f(\mathbf{z})$,
5. $K(\mathbf{x}, \mathbf{z}) = K_3(\phi(\mathbf{x}), \phi(\mathbf{z}))$
6. $K(\mathbf{x}, \mathbf{z}) = \mathbf{x}^T\mathbf{B}\mathbf{z}$.

Examples of kernel functions

Some of the most common forms of kernel functions that satisfy the above mentioned conditions and properties, specially for the field of support vector machine, are:

1. Polynomial generator, which is formulated in two forms as

$$K(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i^T\mathbf{x}_j)^p \tag{3.26}$$

or,

$$K(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i^T\mathbf{x}_j + 1)^p \tag{3.27}$$

for a pre-defined degree of p.

2. Gaussian radial basis function (GRBF), which can be expressed as

$$K(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(\frac{-\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\sigma^2}\right) \tag{3.28}$$

   where σ is the width parameter.

3. Sigmoidal neural network, which is represented by

$$K(\mathbf{x}_i, \mathbf{x}_j) = \tanh(k\mathbf{x}_i^T\mathbf{x}_j + \delta) \tag{3.29}$$

   where k, δ are sigmoid function parameters. It is worth mentioning that the sigmoidal kernel satisfy Mercer's theorem [9] only for some specific values of $k$andδ.

3.4 Statistical Learning Methods

The theory of Vapnik and Chervonenkis (VC) has motivated the development of the SVMs, which is the core of most work in this study. This theory was the underpinning of the field of statistical learning. In this section, the basic principles and main results of the VC theory are presented. VC theory provides reliable bounds on the generalisation of linear classifiers in a way that indicates how to control the complexity of linear functions in kernel spaces.

The initial introduction of the statistical learning theory is dated to the late 1960's [10]. It was meant to be a theoretical framework for the problem of function estimation given some sort of observed data. More attention has been given to the statistical learning theory in the 1990's when new types of learning algorithms (SVMs) based on the developed theory were proposed. This made statistical learning theory a promising tool for creating practical algorithms for estimating functions in high dimensional spaces, not just a theoretical framework. The general learning theory has the following main four theories, according to [80], to address the questions of learning process:

1. The theory of consistency of learning processes; in order to answer the question of the conditions for consistency of the empirical risk minimisation (ERM) principle.

2. The rate of convergence of learning processes; to answer "How fast does the sequence of smallest empirical risk values converge to the smallest actual risk?" In other words, what is the rate of generalisation of a learning machine that implements the empirical risk minimization principle?

3. The theory of controlling the generalisation of learning processes; How can one control the rate of convergence (the rate of generalisation) of the learning machine?

4. The theory of constructing learning algorithms. How can one construct algorithms that can control the rate of generalisation?

Empirical risk minimisation

The core task of a learning machine to be 'trained' is to find a functional form $f(\mathbf{x}, \boldsymbol{\alpha})$ that best describe the data set $\{\mathbf{x}, y\}$ as seen in previous sections; In other words, to find the optimum values of the parameter vector $\boldsymbol{\alpha}$ that minimises the risk function [76] of the form

$$R(\boldsymbol{\alpha}) = \frac{1}{2} \int |y - f(\mathbf{x}, \boldsymbol{\alpha})| \, \mathrm{d}P(\mathbf{x}, y), \tag{3.30}$$

where $P(\mathbf{x},\ y)$ is the probability distribution from which these data are drawn from. The data are assumed to be *i.i.d.* (independently and identically distributed). This distribution, however, is mostly unknown and only the observed data in the training $(\{\mathbf{x}_i, y_i\}_{i=1}^{N})$ are available.

In order to minimise the risk functional in Equation (3.30), the following induction principle, called empirical risk minimisation (ERM), is usually used [11]. The principle is to approximate the risk functional $R(\boldsymbol{\alpha})$ by the function which minimises the empirical risk

$$R_{emp}(\boldsymbol{\alpha}) = \frac{1}{N} \sum_{i=1}^{N} |y_i - f(\mathbf{x}_i, \boldsymbol{\alpha})| \tag{3.31}$$

that is defined to be the mean error rate of the training data points from a finite number of observations, which are.constructed on the basis of the training set. It is worth mentioning that the MSE can also be used in Equation (3.31).

The ERM principle is quite general. The classical methods, that solve estimation problems, such as the least squares method or the maximum likelihood method, are realisations of the ERM principle for the specific loss functions in the learning processes. Since the ERM principle is a general formulation of these classical estimation problems, any theory concerning the ERM principle applies to the classical methods as well [80].

The VC dimension

The VC dimension is a property of a set of functions $\{f(\mathbf{x}, \boldsymbol{\alpha})\}$, and can be defined for various classes of function $f(\mathbf{x}, \boldsymbol{\alpha})$. The VC dimension of a set of estimator functions $\{f(\mathbf{x}, \boldsymbol{\alpha})\}$ is the maximum number h of vectors $\{\mathbf{x}_1, ..., \mathbf{x}_h\}$ which can be separated in all possible 2^h ways using functions of this set, which is usually said that that set of vectors is shattered by that set of functions. If for any N there exists a set of N vectors which can be shattered by the set then the VC dimension is equal to infinity. Note that, if the VC dimension is h, then there exists at least one set of h points that can be shattered, but in general it will not be true that every set of h points can be shattered.

Structural risk minimisation

The principle of structural risk minimisation (SRM), introduced by Vapnik [11], is intended to minimise the risk functional with respect to both empirical risk and VC dimension of the set of functions. Let S the set of functions $f(\mathbf{x}, \boldsymbol{\alpha})$, $\alpha \in \Lambda$, be provided with a structure: so that S is composed of the nested subsets of functions $S_k = \{f(\mathbf{x}, \boldsymbol{\alpha}),\ \alpha \in \Lambda_k\}$ such that

$$S_1 \subset S_2 \subset \cdots \subset S_n. \tag{3.32}$$

Considering the quantity $\frac{1}{2}\left|y_i - f(\mathbf{x}_i, \boldsymbol{\alpha})\right|$ in Equation (3.31) that is called "loss" [76], and for binary values to this quantity $\{0, 1\}$, one can choose some η such that $0 \leq \eta \leq 1$. Then for losses with probability $(1 - \eta)$, the VC theory [11] suggests the following bound

$$R(\alpha) \leq R_{emp}(\boldsymbol{\alpha}) + \sqrt{\left(\frac{h(\log(2N/h) + 1) - \log(\eta/4)}{N}\right)}, \tag{3.33}$$

where h is the VC dimension and N is the number of training data samples. The entire right hand side of Equation (3.33) is called the risk bound or guaranteed risk, and the second term on it is called VC confidence. Note that this VC confidence term depends on the chosen class of functions, whereas the empirical risk and actual risk depend on the one particular function chosen by the training procedure. So, the aim is to find that subset of the chosen set of functions, such that the risk bound for that subset is minimised. Clearly, it is not possible to arrange things so that the VC dimension h varies smoothly, since it is an integer. Instead, a 'structure' is introduced by dividing the entire class of functions into nested subsets as in Equation (3.32). SRM then consists of finding that subset of functions which minimises the bound on the actual risk. This can be done by simply training a series of machines, one for each subset, where for a given subset, the goal of training is simply to minimise the empirical risk. One then takes that trained machine in the series whose sum of empirical risk and VC confidence is minimal. The SRM principle actually suggests a tradeoff between the quality of the approximation and the complexity of the approximating function.

3.5 Support Vector Machines

An significant result from the statistical learning theory is the emerge of SVMs as mentioned in Section 3.4. SVMs are elegant and highly principled learning methods for the design of feedforward networks. Its derivation follows the method of SRM principle that is rooted in VC dimension theory, which makes it more profound. SRM was discussed in Subsection 3.4. As the name implies, the design of the machines hinges on the extraction of a subset of the training data that serves as support vectors and therefore represents a stable characteristics of the data. The SVMs includes different models of intrinsic statistical regularities contained in the training data, yet they all stem from common root in a SVM setting.

SVMs construct a hyperplane as the decision surface in such a way that the margin of separation between the positive and negative samples is maximised in an appropriate feature space, known as maximal margin rule [76]. Figure 3.4 illustrates the maximal margin rule for a 2D data patterns. The circled points around the decision line are called support vectors (SVs) and they represent the data patterns in the testing stage.

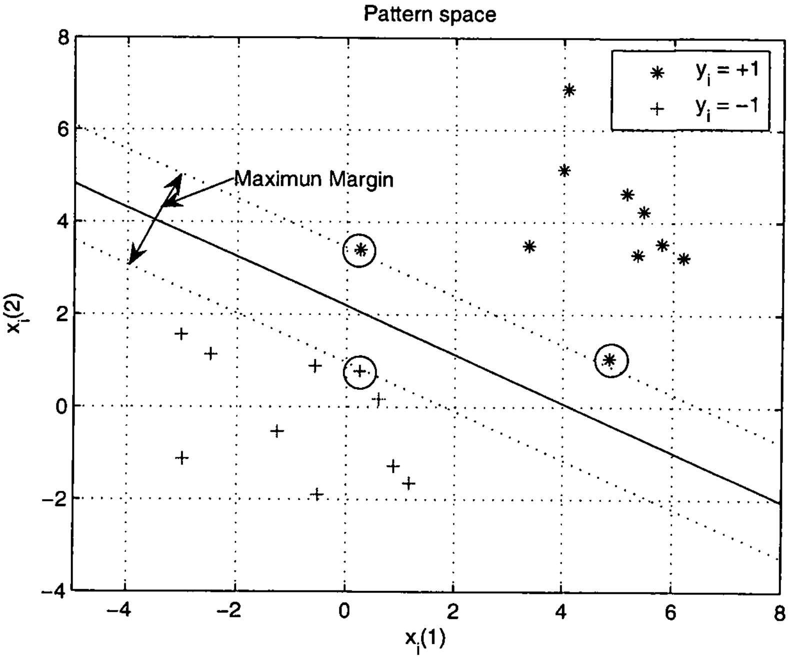


Figure 3.4: Maximum margin rule

Boser et al. [81] combined the kernel function with large margin hyperplanes, leading to kernel based SVMs that are highly successful in solving various nonlinear and linearly nonseparable problems in machine learning.

In this section, three variants of support vector classifiers (SVCs) are described, namely, the hard margin SVC, the soft margin SVC, and the ν–SVM. Before starting the discussion of these SVM based classifiers, it is necessary to emphasise the general form of the required classifier. This can be expressed as to find the class $\hat{y}$ of a given test input pattern $\mathbf{x}$ with regard to part or all of the training data $\{\mathcal{T} : \mathcal{T} \subseteq \mathcal{D}\}$ as follows.

$$\hat{y}(\mathbf{x}) = f(\mathbf{x}) = \sum_{\mathbf{x}_i \in \mathcal{T}} \alpha_i y_i K(\mathbf{x}_i, \mathbf{x}) + \beta, \tag{3.34}$$

where $\alpha's$ and β are the classifiers' parameters to be determined via various criteria as shown in the rest of this PhD study.

3.5.1 Hard Margin SVC

For linearly separable patterns, the hard margin SVC is used. The fundamental principle of hard margin SVC is to find a linear hyperplane ($\mathbf{w}$) in higher dimensional space that maximise the distance (margin) between two different patterns. Hence, the optimisation problem in its primal form is defined to minimise

$$W_p^{hard-margin}(\mathbf{w}) = \frac{1}{2}\mathbf{w}^T\mathbf{w} \tag{3.35}$$

subject to

$$y_j[\mathbf{w}^T\varphi(\mathbf{x}_j) + \beta] \geq 1, \; j = 1, 2, ..., N \tag{3.36}$$

where $\varphi(.)$ is the mapping function to feature space and β is the bias or the shift from origin.

By introducing Lagrange multipliers ($\boldsymbol{\alpha} = [\alpha_1, ..., \alpha_N]$) and applying Karush-Kuhn-Tucker (KKT) conditions [82] for constrained optimisation [71], the primal objective function (3.35) with its constrains in (4.12) is converted to dual formulation. The optimisation process, according to [10], is then to find the values of $\alpha's$ of the classifier that maximise the resulting dual objective function

$$W_{\rm d}(\boldsymbol{\alpha}) = \sum_{i=1}^{N}\alpha_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j y_i y_j K(\mathbf{x}_i, \mathbf{x}_j) \tag{3.37}$$

subject to

$$\begin{gathered} \textstyle\sum_{i=1}^{N}\alpha_i y_i = 0 \\ \alpha_i \geq 0 \end{gathered} \tag{3.38}$$

where $K(\mathbf{a}, \mathbf{b}) = \varphi(\mathbf{a})^T\varphi(\mathbf{b})$ is an arbitrary kernel function that is chosen according to the application. A well known procedure called quadratic programming (QP) [83] may be used to minimise $-W_d(\boldsymbol{\alpha})$. Nonzero Lagrange multipliers of the optimising solution correspond to the SVs that are used to construct the classifier in (3.34).

3.5.2 Soft Margin SVC

In linearly nonseparable cases, the soft margin SVC is widely used for its classification error-tolerance capabilities. The soft margin SVC introduces the margin slack vector $\boldsymbol{\xi} = [\xi_1, ..., \xi_N]$ to allow the possibility of samples violating inquality constrains in (3.36), with the soft margin loss

$$R_{LOSS}^{soft-margin-SVC}(\mathbf{x}, y) = \begin{cases} 0, & yf(\mathbf{x}) \geq 1 \\ 1 - yf(\mathbf{x}) & \text{otherwise} \end{cases} . \tag{3.39}$$

By involving the 1-norm of the margin slack vector $\boldsymbol{\xi}$, the primal form of the soft margin SVC optimisation problem is defined as to minimise

$$W_p^{soft-margin}(\mathbf{w}, \boldsymbol{\xi}) = \frac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum_{i=1}^{N}\xi_i \tag{3.40}$$

subject to

$$y_j[\mathbf{w}^T\varphi(\mathbf{x}_j) + \beta] \geq 1 - \xi_j,\ j = 1, 2, ..., N \tag{3.41}$$

where C is a controlling, or regularisation, parameter for the optimisation stability and tolerance allowance, and is a subject of wide area of research [84].

As in the hard margin case, the dual optimisation form of (3.40), after introducing Lagrange multipliers and applying KKT conditions, can be re-expressed as

$$W_{\mathrm{d}}^{soft-margin}(\boldsymbol{\alpha}) = \sum_{i=1}^{N}\alpha_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j y_i y_j K(\mathbf{x}_i, \mathbf{x}_j) \tag{3.42}$$

subject to

$$\begin{array}{c} \sum_{i=1}^{N}\alpha_i y_i = 0 \\ 0 \leq \alpha_i \leq C \end{array} \tag{3.43}$$

One can easily notice that the dual optimisation form in (3.42) is very similar to the hard margin counter-part with the change in the upper bound of the Lagrange multipliers indicating the limits of error-tolerance in the learning machine.

It is also worth mentioning that some literature use the 2-norm of $\boldsymbol{\xi}$ in formulating the optimisation primal form [71] so that the soft margin SVC learning is viewed as a special case of the hard margin SVC with a modified kernel functions.

ν–SVM

An alternative and equivalent formulation of the support vector machine, known as the ν-SVM, has been proposed in [82]. The ν–SVM is also a soft-margin SVM which employs the same margin slack vector $\boldsymbol{\xi}$ as in SVM but with a different soft-margin loss, given by

$$R_{LOSS}^{\nu-SVM}(\mathbf{x}, y) = \begin{cases} 0, & yf(\mathbf{x}) \geq \rho \\ \rho - yf(\mathbf{x}) & \text{otherwise} \end{cases} . \tag{3.44}$$

where ρ denotes the margin width varying through positive values. Thus, the SVM can be viewed as a special case of the ν-SVM with margin width equal to 1.

The dual form representation of the ν-SVM optimisation formulation involves, after introducing Lagrange multipliers, maximising

$$\tilde{L}(\boldsymbol{\alpha}) = -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j y_i y_j k(\mathbf{x}_i, \mathbf{x}_j) \tag{3.45}$$

subject to the constraints

$$\begin{array}{l} 0 \leq \alpha_i \leq \frac{1}{N} \\ \sum_{i=1}^{N}\alpha_i y_i = 0 \\ \sum_{i=1}^{N}\alpha_i \geq \nu \end{array} \cdot \tag{3.46}$$

This approach has the advantage that the parameter ν, which replaces C in standard SVM, can be interpreted as both an upper bound on the fraction of *margin errors* (points for which $\xi_i > 0$ and hence which lie on the wrong side of the margin boundary and which may or may not be misclassified) and a lower bound on the fraction of support vectors.

Karush-Kuhn-Tucker optimality conditions

The KKT conditions play a central role in both the theory and practice of constrained optimisation. For the primal problem above, Equation (3.40), the KKT conditions may be stated [83]:

$$\frac{\partial}{\partial w_n}W_p = w_n - \sum_{i=1}^{N}\alpha_i y_i \varphi(\mathbf{x}_i) = 0, \; n = 1, ..., D \tag{3.47}$$

$$\frac{\partial}{\partial \beta} W_p = -\sum_{i=1}^{N} \alpha_i y_i = 0, \tag{3.48}$$

$$\begin{aligned} y_i(\mathbf{w}^T \varphi(\mathbf{x}_i) + \beta) - 1 \geq 0 \quad i = 1, ..., N \\ \alpha_i \geq 0 \quad \forall i \end{aligned} \tag{3.49}$$

or,

$$\alpha_i [y_i(\mathbf{w}^T \varphi(\mathbf{x}_i) + \beta) - 1] = 0 \forall i \tag{3.50}$$

The KKT conditions are satisfied at the solution of any constrained optimization problem, with any kind of constraints. This rather technical regularity assumption holds for all support vector machines, since the constraints are always linear. Furthermore, the problem for SVMs is convex, and for convex problems, the KKT conditions are necessary and sufficient for $\{\mathbf{w}, \boldsymbol{\alpha}, \boldsymbol{\beta}\}$ to be an optimal (global) solution. Thus, solving the SVM problem is equivalent to finding a solution to the KKT conditions. It comes handy that these conditions are particularly simple for the dual SVM problem, Equation (4.11), so that [85]

$$\begin{aligned} &\alpha_i = 0 && \Rightarrow y_i f(x_i) \geq 1 \text{ and } \xi_i = 0 \\ &0 < \alpha_i < C && \Rightarrow y_i f(x_i) = 1 \text{ and } \xi_i = 0 \\ &\alpha_i = C && \Rightarrow y_i f(x_i) \leq 1 \text{ and } \xi_i \geq 0 \end{aligned} \tag{3.51}$$

They reveal one of the most important property of SVMs: the solution is sparse, *i.e.*, many patterns are outside the margin area and the corresponding optimal α values are zero.

Note that, while $\mathbf{w}$ is explicitly determined by the training procedure, the threshold β is not and it is implicitly determined. However, β is easily found by using the KKT 'complementarity' condition, Equations (3.50), by choosing any i for which $\alpha_i \neq 0$ and computing β (taking the mean value of β resulting from all such equations is numerically more stable). In real world problems, finding optimisation solutions requires applying numerical methods.

3.6 Model Selection

Classification performance of a parametric classifier depends on the process of picking the best value for the classifier's hyper-parameters, such as the kernel width and regularisation parameters (C) for SVMs. This leads to a nontrivial model selection problem [84] that needs either an exhaustive search over the space of hyper-parameters or an optimisation procedure that explores only a finite subset of the possible values.

Validation tests

Ideally, one would like to select models of a classifier, based on the true risk of the classifier. Unfortunately, such a quantity is not accessible, and one has to build estimates for the true risk of a classifier. The following validation procedures are widely used in the literature [9].

Single validation

If enough data are available, it is possible to estimate the error rate on a validation set. Such an estimate is unbiased and the corresponding variance gets smaller as the size of the validation set increases [86]. Letting V denotes the set of N_v labeled validation samples $V = \{\mathbf{x}_i^v, y_i^v\}_{i=1}^{N_v} \in \mathbb{R}^d \times \{-1, +1\}$, with no intersection with the training samples in the space, the single validation error estimate is given by

$$E_{SVT} = \frac{1}{N_v} \sum_{i=1}^{N_v} \text{sign}(-y_i^v f(\mathbf{x}_i^v)). \tag{3.52}$$

Cross-validation

If no enough data are available for validation, a U-fold-cross-validation procedure within the training samples can be employed to estimate the error rate of the classifier [86]. Such a procedure is executed by randomly dividing the training samples into U groups. In one trial, one group is used for validation and the remaining $U-1$ groups for training, so that every group is used as the validation set once. The cross-validation error estimate is calculated by averaging the classification error rates of the U validation sets, given as

$$E_{CVT} = \frac{1}{U} \sum_{i=1}^{U} \frac{1}{|V_i|} \sum_{j \in V_i} \text{sign}(-y_j^v f(\mathbf{x}_j^v)), \tag{3.53}$$

where $|V_i|$ denotes the number of validation samples in group V_i, and $\mathbf{x}_j^v, y_j^v$ denote the j^{th} validation sample. *sign* operator in Equations (3.52) and (3.53) is the thresholding sigmoidal function.

Leave-one-out cross-validation procedure

The LOO procedure [84] removes one sample from the N training samples and construct the decision rule on the basis of the remaining $(N - 1)$ training samples, then tests on the removed training sample. In this fashion, one tests all of the N training samples using N different decision rules. The LOO error gives an almost unbiased estimate of the expected generalisation error. Also, the estimation variance may be large.

3.7 Bayesian Inference in Machine Learning

Bayesian inference methods have been discussed for linear classification models in Subsection 3.2.3. They were mainly applied in terms of underlying data distributions. This section gives a basic introduction to the principles of Bayesian inference in a machine learning context in terms of model parameters' distributions, with an emphasis on pattern recognition problems including linear classification and kernel based classification. The importance of marginalisation for dealing with uncertainty is also presented.

3.7.1 A Probabilistic Classification Framework

Bayesian inference can be applied in the classification modelling procedure. So, for a typical parameterised classification model, the conditional probability

$$P(y|\mathbf{x}) = f(\mathbf{x}; \mathbf{w}); \tag{3.54}$$

can be used, where $\mathbf{w}$ denotes a vector of all the 'adjustable' parameters in the model. Then, given a set $\mathcal{D}$ of N examples of training patterns, $\mathcal{D} = \{\mathbf{x}_i, y_i\}_{i=1}^N$, a conventional approach would involve the maximisation of some measure of 'accuracy' (or minimisation of some measure of 'loss') of the classification model for $\mathcal{D}$ with respect to the adjustable parameters. Of course, if the model $f(\mathbf{x};\mathbf{w})$ is made too complex, *e.g.* with too many parameters, a poor model of the true underlying distribution $P(y|\mathbf{x})$ is consequently realised.

For a new testing pattern $\mathbf{x}$, as usual, the task is to find a prediction $\hat{y}$ by evaluating $f(\mathbf{x};\mathbf{w})$ with parameters $\mathbf{w}$ set to their optimal values.

The first key element of the Bayesian inference paradigm is to treat parameters such as $\mathbf{w}$ as random variables [70], exactly the same as $\mathbf{x}$ and y. So the conditional probability now becomes $P(y|\mathbf{x},\mathbf{w})$, and the dependency of the probability of y on the parameter settings, as well as $\mathbf{x}$, is made explicit. Rather than 'learning' comprising the optimisation of some quality measure, a distribution over the parameters $\mathbf{w}$ is inferred from Bayes' rule. To obtain this posterior distribution over $\mathbf{w}$, it is necessary to specify a prior distribution $p(\mathbf{w})$ before observing the data.

However, the most attractive facet of a Bayesian approach is the manner in which Occam's Razor [9][2] can be automatically implemented by "integrating out" all irrelevant variables. That is, under the Bayesian framework there is an automatic preference for simple models that sufficiently explain the data without unnecessary complexity.

Maximum likelihood inference

The maximum likelihood estimate for $\mathbf{w}$ is the value which maximises $p(y|\mathbf{w},\sigma^2)$. In fact, this is identical to the LS solution, discussed in Subsection 3.2.1, which it can be noticed that minimising the sum of the squared errors is equivalent to minimising the negative logarithm of the likelihood which here is $E_{ML}(\mathbf{w}) = -\log p(y|\mathbf{w},\sigma^2)$

[2] Occam razor principle was originally appeared in a religious context in fourteenth century, and was commonly translated as "entities should not be multiplied unnecessarily". In machine learning context, it can be re-interpreted as "models should be no more complex than is sufficient to explain the data".

Posterior inference

To control the model complexity, *a prior* distribution is defined which expresses our 'degree of belief' over values that $\mathbf{w}$ might take, so that

$$p(\mathbf{w}|\theta) = \prod_{m=1}^{M} \left(\frac{\theta}{2\pi}\right)^{\frac{1}{2}} \exp\left\{-\frac{1}{2}\theta w_m^2\right\}. \qquad (3.55)$$

This (common) choice of a zero-mean Gaussian prior, expresses a preference for smoother models by declaring smaller weights to be a priori more probable. Though the prior is independent for each weight, there is a shared inverse variance hyperparameter θ which moderates the strength of "degree of belief".

Previously, a single point estimate $\mathbf{w}_{LS}$ is determined for the weights. Now, given the likelihood and the prior, the posterior distribution over $\mathbf{w}$ can be computed via Bayes' rule:

$$p(\mathbf{w}|y, \theta, \sigma^2) = \frac{likelihood \times prior}{normalising\, factor}. \qquad (3.56)$$

So instead of determining a single value for $\mathbf{w}$, a distribution over all possible values is inferred. In effect, this is merely updating the prior "degree of belief" in the parameter values according to the information provided by the data $\mathcal{D}$, with more posterior probability assigned to values which are both probable under the prior and which best describe the data [87].

MAP estimation

The maximum a posterior (MAP) estimate for $\mathbf{w}$ is the single most probable value under the posterior distribution $p(\mathbf{w}|y, \theta, \sigma^2)$. This is equivalent to minimising $E_{MAP}(\mathbf{w}) = -\log p(\mathbf{w}|y, \theta, \sigma^2)$, or equivalently, maximising the numerator since the denominator in Bayes' rule (3.56) is independent of $\mathbf{w}$.

Marginalisation

The distinguishing element of Bayesian methods is really marginalisation [70], where instead of seeking to 'estimate' all 'nuisance' variables in the model, it is much comprehensible to integrate these variables out. As it can be shown, this is a powerful component of the Bayesian framework.

So, the "true" Bayesian way is to integrate out, or marginalise over, the uncertain variables $\mathbf{w}$ in order to obtain the predictive distribution, so that

$$p(\hat{y}|y;\theta,\sigma^2) = \int p(\hat{y}|\mathbf{w},\sigma^2)p(\mathbf{w}|y,\theta,\sigma^2)\mathrm{d}\mathbf{w}. \tag{3.57}$$

This distribution $p(\hat{y}|y;\theta,\sigma^2)$ incorporates uncertainty over the weights having seen y, by averaging the model probability for $\hat{y}$ over all possible values of $\mathbf{w}$.

For any general model, in order to predict $\hat{y}$ given some training data y, what is really required is $p(\hat{y}|y)$. That is, to integrate out all variables not directly related to the task at hand.

In fact, and for more general modelling, the hyperparametrs θ, σ^2 are also unknowns. And, to be fully Bayesian, one needs to change the posterior distribution in Equation (3.56) so that a prior $p(\theta)$ along with a prior over the noise level $p(\sigma^2)$ are also defined. Then the full posterior over 'nuisance' variables becomes

$$p(\mathbf{w},\theta,\sigma^2|y) = \frac{p(y|\mathbf{w},\sigma^2)p(\mathbf{w}|\theta)p(\theta)p(\sigma^2)}{p(y)}. \tag{3.58}$$

3.7.2 Sparse Bayesian Models

Sparsity is a very important requirement in most practical applications, hence it is desirable to adapt probabilistic models to include sparsity. The most common approach is via an appropriate regularisation term or prior [87]. The most common regularisation term, that is widely and practically used, corresponds to a Gaussian prior and is easy to work with, but while it is an effective way to control complexity, it does not promote sparsity. In the regularisation sense, the 'correct' term would be $E_w(\mathbf{w}) = \sum_m |w_m|^0$ [88], but this, being discontinuous in w_m, is very difficult to work with. Instead, $E_w(\mathbf{w}) = \sum_m |w_m|^1$ is a workable compromise which gives reasonable sparsity and reasonable tractability, and is exploited in a number of methods, including as a Laplacian prior $p(\mathbf{w}) \propto \exp(\sum_m |w_m|^1)$ [89].

A more elegant way of obtaining sparsity within a Bayesian framework was proposed by Tipping, in [88]. The idea was that sparsity can be obtained by retaining the traditional Gaussian prior, which is preferred for tractability. The modification to the earlier Gaussian prior (3.55) is subtle, that is

$$p(\mathbf{w}|\theta_1, ..., \theta_M) = \prod_{m=1}^{M} \left(\frac{\theta_m}{2\pi}\right)^{\frac{1}{2}} \exp\left\{-\frac{1}{2}\theta_m w_m^2\right\} \tag{3.59}$$

In contrast to the model in Subsection 3.7.1, M hyperparameters $\boldsymbol{\theta} = [\theta_1, ..., \theta_M]$ are now obtained, one θ_m independently controlling the (inverse) variance of each weight w_m. This sparse Bayesian inference method has led to the introduction of what is called relevance vector machines (RVMs) [88]. In typical RVM, probabilistic predictions are produced based on Bayesian techniques. Basically, RVMs introduce a zero-mean Gaussian prior over every weight to obtain a sparse solution. As a result of sparseness-inducing prior, posteriors of many weights are sharply distributed around zero, hence these weights are pruned and the model becomes sparse. The RVM learning model will be discussed in further detail in Chapter 6.

Probabilistic classification vector machines

RVMs adopt the zero-mean Gaussian prior over weights for both positive and negative classes in classification problems, hence some training points that belong to positive class may have negative weights and vice versa. This formulation might result in the situation that the decision of RVMs is based on some untrustful vectors, and thus is unreliable in some situations.

Recently, a probabilistic classification vector machine (PCVM) algorithm is introduced [90] to overcome this RVM disadvantage. In PCVM, different priors over weights are introduced (for training points) belonging to different classes, *i.e.*, the nonnegative, left-truncated Gaussian for the positive class and the nonpositive, right-truncated Gaussian for the negative class as shown in Figure 3.5. PCVMs also implement a parameter optimisation procedure for kernel parameters in the training algorithm, which is proved to be effective in practice. However, this truncation modification makes the inference integrals intractable, thus an expectation–maximisation (EM) is used to get a MAP estimation of parameters.

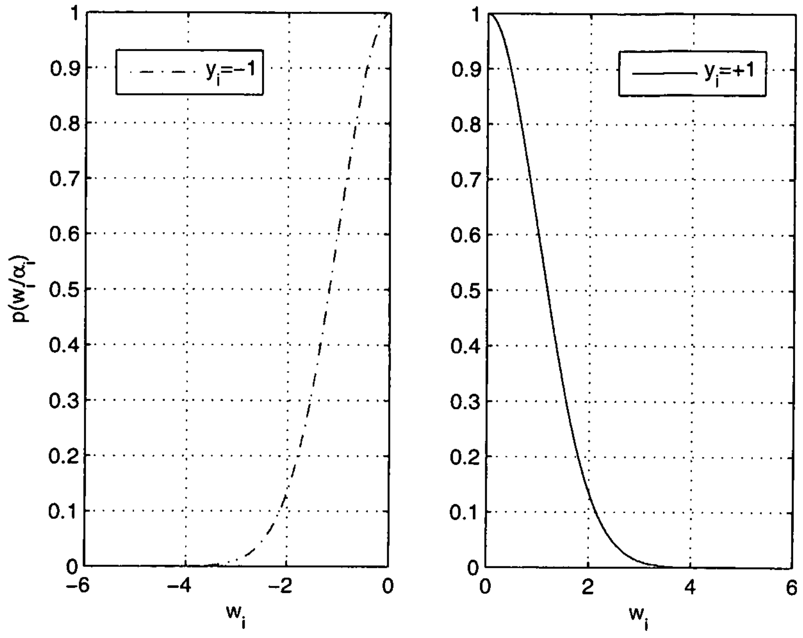


Figure 3.5: The pdf of weight prior in PCVM

3.8 Summary

This chapter has provided the necessary background to the area of machine learning in the context of pattern recognition task. Starting with a general overview, the tasks and applications of the machine learning processes are presented followed by some common linear classification algorithms to explain the concept of learning. Nonlinear classification is perceived as linear through the utilisation of kernel mapping power, hence the fundamentals of kernel induced methods were discussed.in some details. The statistical learning theory enabled the learning machines, through its resulting SVMs, to combine the linear classification and kernel mapping with a solid theoretical generalisation capabilities. Also, the selection of model parameters is discussed for better choice of machine learning parameters. The chapter concludes the presentation with a brief introduction to the probabilistic learning or Bayesian inference models of machine learning.

Chapter 4

Support Vector Machine based Equalisation

This chapter deals with the application of SVM family of algorithms in the issue of channel equalisation. SVM based equalisers are then proposed for the DS-UWB channel equalisation. Three types of equalisers are introduced, namely, SVC, least squares SVC (LS-SVC), and sparse LS-SVC. The rest of the chapter is organised as follows. After the introduction in Section 4.1, the applications of SVM methods in broadband wireless communications systems are discussed in Section 4.2. The general machine learning based system model, for the proposed receivers for DS-UWB systems, is presented in Section 4.3. The proposed SVM based equalisers are then introduced in Section 4.4 with a discussion of the simulation results. Summary is found in Section 4.5.

4.1 Introduction

As introduced in Chapter 3, the applications of statistical learning techniques have attracted many researchers. And the SVMs [10, 91] are resulting statistical learning techniques of related supervised learning methods used to solve the classification problem, by using support vector classifiers (SVCs) [71]. A special property of SVMs is that they simultaneously minimise the empirical classification error and maximise the geometric margin in the training mode, hence they provide high-level classification performance [71]. Kernel induced methods are used in SVM classifiers to perform the classification of nonlinearly separable patterns. A number of modifications and extensions to conventional SVCs have been developed to reduce the complexity of the training process, such as online SVC training [92] and LS-SVC [93].

In digital communications, SVMs have been applied for channel equalisation, channel estimation and multiuser detection (MUI) [12, 13, 14]. For those applications, in fact, the problem was mapped to symbol-by-symbol detection, and the channels were assumed to be a simple stationary multipath fading with short maximum time excess delay for theoretical and educational purposes. In [12], SVMs were considered for decision feedback equalizers (DFE) design using linear feedback filter. The DFE filter in [13] is replaced by a Volterra filter for nonlinear channels. The DS-CDMA multi-user detection (MUD) based on SVM has been discovered in [14]. SVMs have been used for UWB systems in range estimation and positioning applications [94].

4.2 SVM based Equaliser for Broadband Wireless Systems

Digital channel equalisation may be viewed as a classification problem [67]. In such a scenario, the output of a communications channel can be grouped, in a designated manner, to produce a set of vectors which are used as inputs to a classification machine. The output of the machine should match as best as possible the original signal (or, some delayed version of it) entering the channel. The raw data (*i.e.*, channel output) is transformed to a pattern space, *i.e.*, it is grouped as a state vector according to the expected amount of ISI. The pattern space is then transformed to a higher dimensional feature space (usually of much higher dimension, which is sometimes infinite dimensional) that incorporates the nonlinear nature of the model. Equalisation attempts to map the nonlinear channel output to an element of which this mapping represents the inverse of the channel as closely as possible according to some accepted functional measure. Clearly, if this can be done effectively, then detection follows in a straightforward manner. On the other hand, detection only attempts to map the channel output into a finite alphabet, say, such that this mapping is optimum in a statistical sense. The underappreciated point is that in cases where finding the system inverse only serves as an intermediate step to the goal of detecting digital symbols, the system designer usually finds that the direct detection problem is much easier to solve than the more general equalisation problem [13].

The attempt to use the SVM method can be limited by the fact that not always the overall set of incorporated choices is well-suited to the application scenarios; also in such a case, however, important advances in the classical design procedures are possible by exploiting some of the principal SVM contributions. This is clearly seen with reference to the problem of digital channel equalization [91]. Consider the discrete-time linear time-invariant noisy communication channel:

$$r(n) = h(n) \otimes x(n) + w(n) \tag{4.1}$$

where $\otimes$ denotes the discrete-time convolution, $h(n)$ is the channel impulse response with finite impulse response (FIR), $w(n)$ is a zero-mean, independent and identically distributed (i.i.d.) noise process with variance σ_n^2 and Gaussian distribution, $x(n) \in \{-1, 1\}$ is an i.i.d. sequence of information symbols.

4.2.1 System Model

In this work, the concept of SVM based equalisation in digital communication systems is examined for simple 2D scenarios for visualisation purposes. Three simple wireless communication scenarios are considered to equalise received signals to their original transmitted form, considering thereceived signal constellation as attributes space for an SVC in both training mode and detection mode.

The system block diagram is depicted in Figure 4.1, where the input data symbols $d(t)$ are the generated signals of our interest and the final target is to retrieve this signal as accurate as possible. The transmitted symbols are corrupted by the physical medium which is modelled as a FIR filter (the channel $H(z)$).

At the receiver, an AWGN is added to the received signal to produce observed signal $r(t)$, which together with the previously observed signal $r(t-1)$ make the 2D input space of the SVC. The SVC estimates the previous data symbol $\hat{d}(t-1)$ as following

$$\hat{d}(t-1) = sign(f_{SVC}(\mathrm{r}(t))) \tag{4.2}$$

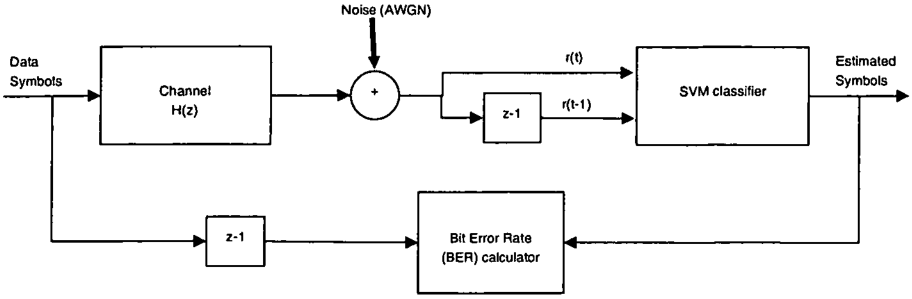


Figure 4.1: System model of SVC based equaliser for broadband wireless channel

where $\mathbf{r}(t) = [r(t), r(t-1)]$, *sign* is the signum function, and

$$f_{SVC}(\mathbf{r}(t)) = \sum_{i \in SV} \alpha_i d_i K(\mathbf{r}(t), \mathbf{r}_i). \tag{4.3}$$

where d_i and $\mathbf{r}_i$ are the support vectors from the training data set $\{\mathbf{r}(t), d(t)\}$, respectively. In this experiment, the bias term β is chosen to be zero considering the common assumption that the symmetry around the origin [14]. The α_i are calculated in the training mode via solving a QP problem to maximise

$$W(\boldsymbol{\alpha}) = \sum_{i=1}^{P} \alpha_i - \frac{1}{2} \sum_{i=1}^{P} \sum_{j=1}^{P} \alpha_i \alpha_j y_i y_j K(\mathbf{r}_i, \mathbf{r}_j) \tag{4.4}$$

where P is the training set (pilot) size and $K(\mathbf{r}_i, \mathbf{r}_j)$ is the Gaussian radial basis function (RBF) kernel mapping which is defined in Equation (5.24), and σ^2 is chosen to be proportional to the noise power in this experiment.

The system performance is then evaluated by comparing original sent symbols with those retrieved by the SVC at different noise levels (*i.e.* calculating the simulated BER).

Experiment configurations

The generated data are assumed to be independently distributed binary symbols, which take the values from the symbol set $\{-1, +1\}$ with equal probability, to represent baseband BPSK pulses (1 million symbols). Three discrete channel models were used to represent simple multipath wireless communication channels:

$$\begin{aligned}
Ch1 &: H(z) = 0.50 + 1.0z^{-1} \\
Ch2 &: H(z) = 0.30 + 0.7z^{-1} \quad +0.3z^{-2} \\
Ch3 &: H(z) = 0.35 + 0.8z^{-1} \quad +1.0z^{-2} \quad +0.8z^{-3}
\end{aligned} \tag{4.5}$$

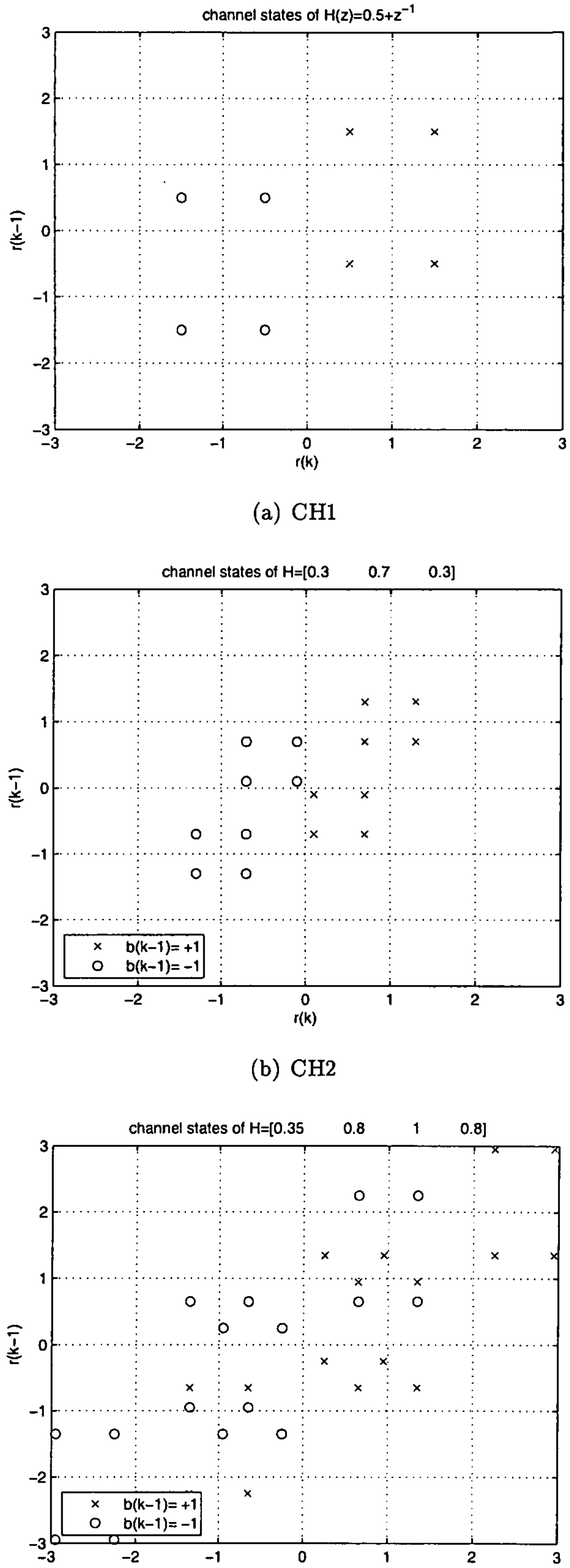
The noiseless channel states of these channels are depicted in Figure 4.2. The accompanying noise SNR (E_b/N_0) varies from 0 to 24 dB.

The SVC is designed to extract the previous transmitted symbol (delay of 1 symbol) to allow the classifier to work with two symbols a time. The upper boundary condition for the QP (C) was set to 10 after testing many empirical values.

4.2.2 Simulation Results

The simulation results of an SVC based equalisation system are presented in this subsection. The SVC simulator was tested for linear data learning and nonlinear data learning considering the RBF kernel mapping. For linear training, $K(\mathbf{r}_i, \mathbf{r}_j)$ was assumed to be merely a normal inner product $\mathbf{r}_i^T \mathbf{r}_j$. The results of both situations are illustrated in Figure 4.3, and it is clearly deduced that, for highly nonlinear separable constellations such as Ch3, considering kernel mapping would improve the performance of the classifier. Generally, the kernel mapping outperforms normal linear SVC.

The results of Figure 4.3 assume a perfect knowledge of the channel at the receiver, that is, only the output of noiseless channel states are used as a training dataset. The SVC is also trained with the noisy data set for more realistic scenarios, and the results are shown in Figure 4.4, which shows the robustness of the SVC with the noisy training data.



(a) CH1

(b) CH2

(c) CH3

Figure 4.2: Channels states constellation

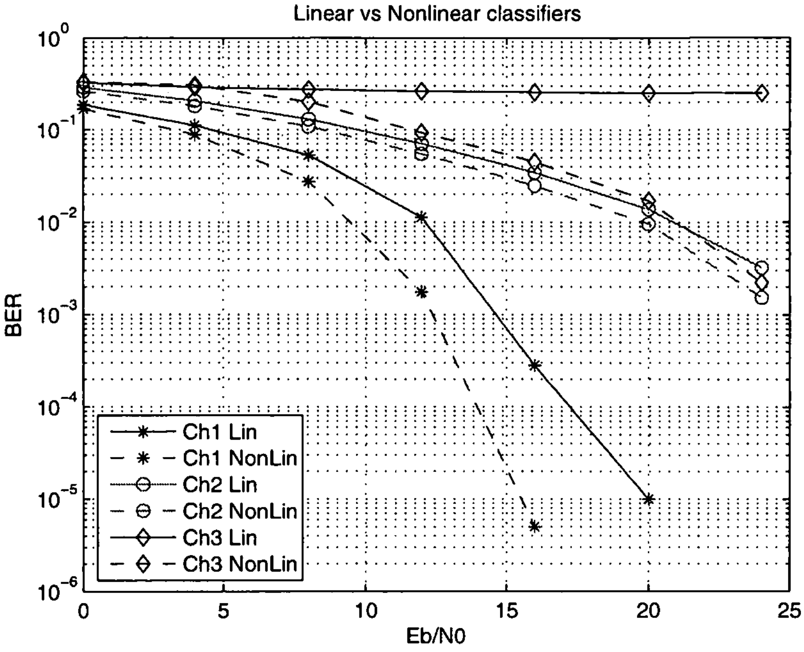


Figure 4.3: Linear vs Nonlinear SVC

To summarise the findings of this experiment, nonlinear SVC performs more effectively than linear classifier due to the introduction of kernel mapping function and the nonlinearly separable nature of the signal constellation of most multipath channels. This is at the cost of more computational complexity. High performance can be achieved if the knowledge of the channel is known to the detector. However, the performance does not change considerably if no information about the channel is available (which is the case for most real applications). This second case can be accomplished by training the detector by a sufficient training data set.

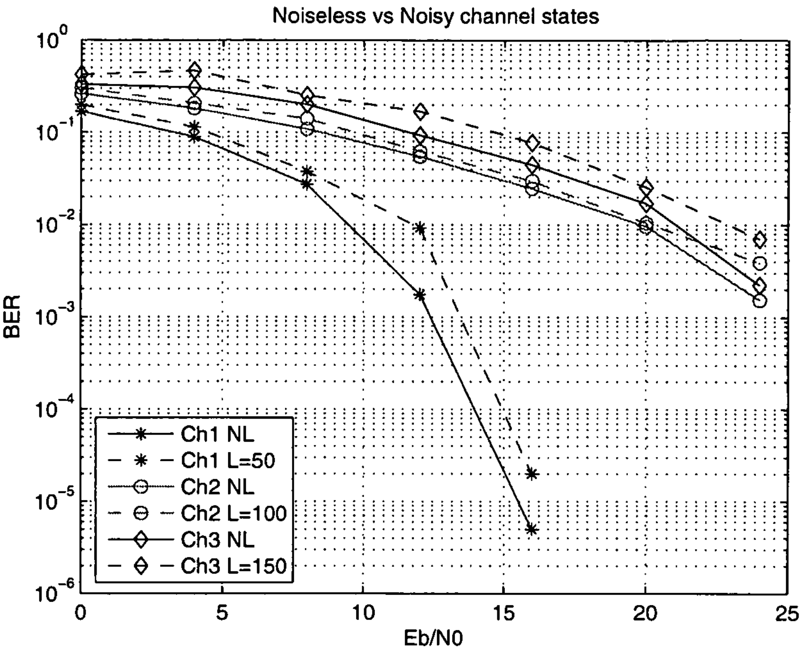


Figure 4.4: Ideal vs Noisy channels performance

4.3 General Machine Learning based Model for DS-UWB Systems

In this section, the general structure of the proposed receivers are introduced and described for DS-UWB system. Figure 4.5 depicts the block diagram of the proposed machine learning (ML) based receiver, this structure will be used throughout the research, and the subsequent work will investigate the learning methodology of the ML units in this structure. The receiver structure, therefore, consists of a bank of independent ML based classifiers with optimised parameters at the receiver in a block-by-block fashion. The M classifiers are arranged in parallel as shown at the receiver in Figure 4.5.

The principle of solving equalisation problem as a pattern recognition is developed to the spreading code space in DS-UWB systems. In the proposed receivers, the received discrete signal in Equation (2.15) is arranged in M parallel groups of N_c chip length each, as shown in Figure 4.5. The signal is passed through M classifiers, and the input vector to the m^{th} $(m = 1, ..., M)$ classifier from the j^{th} received block, $\mathbf{r}$, can be denoted

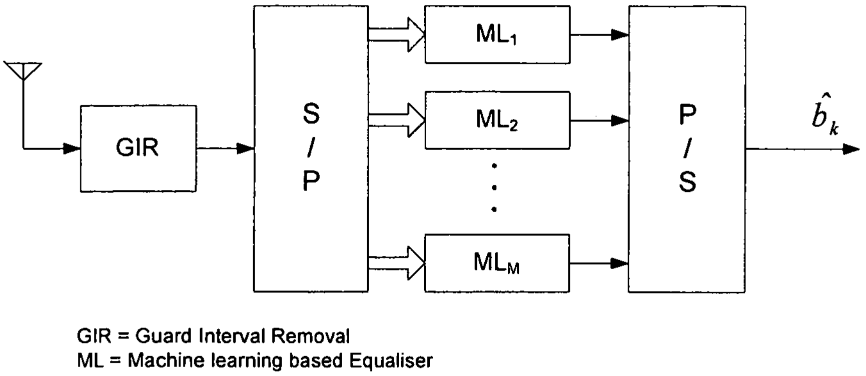

GIR
S / P
ML₁
ML₂
MLM
P / S
$\hat{b}_k$
GIR = Guard Interval Removal
ML = Machine learning based Equaliser

Figure 4.5: General machine learning based receiver for DS-UWB system

in machine learning convention as

$$\mathbf{x}_j^{(m)} = [\mathbf{r}[N_c(m-1)+1]\ \mathbf{r}[N_c(m-1)+2]\ \ldots\mathbf{r}[N_c(m-1)+N_c]]\,.^T \tag{4.6}$$

For training the ML based equalisers, *i.e.*, classifiers, P pilot blocks, each having of M symbols, are transmitted. The m^{th} pilot symbol in the j^{th} $(j = 1, ..., P)$ training block is given by $y_j^{(m)} = b_m(j)$.

In the detection (testing) mode, the estimated m^{th} symbol of the i^{th} $(i = 1, .., B)$ received data block are obtained from the m^{th} machine learning based classifier as

$$\hat{b}_m(i) = \text{sign}\{f_{Machine\ Learning}^{(m)}(\mathbf{x}_i^{(m)})\} \tag{4.7}$$

where *sign* is a decision function defined in Equation (2.20), and $f_{Machine\ Learning}^{(m)}$ is the classification function of the m^{th} classifier, which is generally defined as

$$f_{Machine\ Learning}^{(m)}(\mathbf{x}_i^{(m)}) = \sum_{\mathbf{x}_j^{(m)} \in V} \alpha_j^{(m)} y_j^{(m)} K(\mathbf{x}_i^{(m)}, \mathbf{x}_j^{(m)}) + \beta^{(m)} \tag{4.8}$$

where V is the resulting set of support vectors obtained by the learning process from all the training input vectors $\mathbf{x}_j^{(m)}$, and $y_j^{(m)}$ are the pilot symbols or class labels for $\mathbf{x}_j^{(m)}$. $\alpha's$ are classifier parameters which are calculated in the training mode via the training input vectors with a proper optimisation scheme. $\beta^{(m)}$ in Equation (4.8) is a threshold term that indicates how far the origin is from the hyperplane (or, affine offset [13]).

For simplicity of notation, the classifier's index m will be omitted in the subsequent chapters since the optimisation process applies independently to all the classifiers in the receiver.

$K(\mathbf{x}_a, \mathbf{x}_b)$ in Equation (4.8), is the kernel function of the machine learning classifiers that is discussed in Section 3.3. The natural choice of kernel function in most communication applications is the Gaussian radial basis function (RBF) [12], which takes the form in Equation (5.24) where σ is the kernel width parameter.

4.4 SVM based Equalisers for DS-UWB Systems

This work proposes and investigates the SVM based equalisation for DS-UWB systems by employing three types of SVM based classifiers in the equalisation units in the general receiver structure (*i.e.* the ML units in Figure 4.5). The three chosen classifiers are the SVC, least squares SVC (LS-SVC), and the sparse LS-SVC. To the best of our knowledge, this is the first work to apply the SVM technique to UWB communication systems considering practical UWB channels and scenarios. The following subsections describe these equalisers in details, and present and discuss the simulation results compared to conventional receivers in DS-UWB systems.

4.4.1 SVC based Equaliser

A powerful advantage of SVMs is that only some of the training vectors, referred to as support vectors (SVs), are used in the classification stage. The fundamental principle of SVC is to find a linear hyperplane ($\mathbf{w}$) in higher dimensional space that maximise the distance (margin) between two different patterns. Hence, the optimisation problem in its primal form is defined to minimise

$$W_p(\mathbf{w}, \boldsymbol{\xi}) = \frac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum_{i=1}^{P}\xi_i \tag{4.9}$$

subject to

$$y_j[\mathbf{w}^T\varphi(\mathbf{x}_j) + \beta] \geq 1 - \xi_j,\ j = 1, 2, ..., P, \tag{4.10}$$

where C is the regularisation parameter and ξ_i is the i^{th} slack variable. By introducing Lagrange multipliers (α's) and applying Karush-Kuhn-Tucker (KKT) conditions [82] for the above constrained optimisation [71], the primal objective function in Equation (4.9) with its constrains in Equation (4.12) is converted to the dual formulation. The optimisation process, according to [10], is then to find the values of $\alpha's$ of the m^{th} classifier that maximise the resulting dual objective function:

$$W_{\text{d}}(\boldsymbol{\alpha}) = \sum_{i=1}^{P} \alpha_i - \frac{1}{2} \sum_{i=1}^{P} \sum_{j=1}^{P} \alpha_i \alpha_j y_i y_j K(\mathbf{x}_i, \mathbf{x}_j) \tag{4.11}$$

subject to

$$\begin{aligned} \textstyle\sum_{i=1}^{P} \alpha_i y_i = 0 \\ 0 \le \alpha_i \le C. \end{aligned} \tag{4.12}$$

Again, C is a controlling parameter for the optimisation stability. The dual form facilitates the nonlinear separable data patterns to depend only on the size of the training set, not on the dimension of the high dimensional feature space. A well known procedure called quadratic programming (QP) [13] may be used to minimise $-W_d(\boldsymbol{\alpha})$, subject to the constraints of Equation (4.12). The optimal solution consists of those nonzero values of α's and the corresponding support vectors, which are used to construct the classifier in Equation (4.8).

Parameter selection for SVM

For a SVM, choosing a suitable kernel is imperative to the success of the learning process. The regularisation parameter C is also important, as it controls the tradeoff between the complexity of a SVM and the number of nonseparable points [11]. The most common model selection validation tests have been presented in Section 3.6, and some other testscan be used for SVM model selection as follows.

Leave-one-support-vector-out cross-validation

The leave-one-support-vector-out cross-validation (LOSVOCV) procedure is a modified LOO that has been introduced in the SVM context by [95]. The LOSVOCV algorithm has been proposed for estimating the optimal bandwidth of the kernel of support vector classifiers. Its generalisation performance and computational efficiency have been discussed in comparison with the conventional LOO algorithm [95]. It is initialised using a pre-determined value of kernel width parameter and the SVC is then trained with the whole training data to obtain a set of SVs. At each loop of the training process, after one of the current SVs is deleted, a new decision function is obtained by training the SVC with the remaining SVs. The new decision function is then used to classify the whole training set with errors defined as the total number of misclassification

Automatic tuning of SVM parameters

In [84], an approach for automatically tuning the kernel parameters has been proposed. This is based on the possibility of computing the gradient of various bounds on the generalisation error with respect to these parameters. By using smoothed gradient techniques, the search of kernel parameters space is performed with gradient descent algorithm.

4.4.2 LS-SVC based Equaliser

To reduce the computational complexity of the QP process in standard SVCs, we apply the LS-SVC technique [93] for equalisation, by modifying the inequality constraints in Equation (4.10) to equality constrains. The classification problem in LS-SVC, therefore, is formulated as (in a primal form) to minimise

$$W_{\mathrm{LS_p}}(\mathbf{w},\mathbf{e}) = \frac{1}{2}\mathbf{w}^{\mathrm{T}}\mathbf{w} + \frac{1}{2}\gamma\sum_{j=1}^{P} e_j^2 \tag{4.13}$$

subject to the equality constrains

$$y_j[\mathbf{w}^T\varphi(\mathbf{x}_j) + \beta] = 1 - e_j,\ j = 1, 2, ..., P \tag{4.14}$$

where $\mathbf{w}$ is the hyperplane coefficients vector, e_j is the misclassification error due to the equality constraint, and γ is a regularisation parameter that is predefined to control the error tolerance weight.

By introducing Lagrange multipliers, we construct a Lagrange function from Equation (4.13) as

$$W_{\mathrm{LS_d}}(\mathbf{w},\beta,\mathbf{e},\boldsymbol{\alpha}) = W_{\mathrm{LS_p}}(\mathbf{w},\mathbf{e}) - \sum_{j=1}^{P} \alpha_j \{y_j[\mathbf{w}^T\varphi(\mathbf{x}_j)+\beta] - 1 + e_j\} \qquad (4.15)$$

where $\alpha_i \in \mathbb{R}$, and the conditions for optimality become

$$\begin{array}{ll} \frac{\partial \mathcal{L}}{\partial \mathbf{w}} = 0 \Rightarrow & \mathbf{w} = \sum_{j=1}^{P} \alpha_j y_j \varphi(\mathbf{x}_j) \\ \frac{\partial \mathcal{L}}{\partial \beta} = 0 \Rightarrow & \sum_{j=1}^{P} \alpha_j y_j = 0 \\ \frac{\partial \mathcal{L}}{\partial e_j} = 0 \Rightarrow & \alpha_j = \gamma e_j,\ j = 1,2,...,P \\ \frac{\partial \mathcal{L}}{\partial \alpha_j} = 0 \Rightarrow & y_j[\mathbf{w}^T\varphi(\mathbf{x}_j)+\beta] - 1 + e_j = 0,\ j = 1,2,...,P. \end{array} \qquad (4.16)$$

It can be derived that Equation (4.16) can be expressed in a matrix form as

$$\begin{bmatrix} 0 & -\mathbf{y}^T \\ \mathbf{y} & \boldsymbol{\Omega}+\gamma^{-1}\mathbf{I} \end{bmatrix} \begin{bmatrix} \beta \\ \boldsymbol{\alpha} \end{bmatrix} = \begin{bmatrix} 0 \\ \mathbf{1} \end{bmatrix} \qquad (4.17)$$

where $\mathbf{y} = [y_1...y_P]^T$, and the element in the i^{th} row, j^{th} column of $\boldsymbol{\Omega}$ is defined as

$$\begin{aligned} \Omega_{ij} &= y_i y_j \varphi^T(\mathbf{x}_i)\varphi(\mathbf{x}_j) \\ &= y_i y_j K(\mathbf{x}_i, \mathbf{x}_j). \end{aligned} \qquad (4.18)$$

The linear equation in Equation (4.17) can be easily solved by many existing algorithms rather than the QP technique used in SVCs in Subsection 4.4.1. However, the concept of support vectors disappears in the LS-SVC case since the solution contains a spectrum of values rather than few nonzero values as in SVCs. This, therefore, introduces an increase in detection complexity. Once classifier coefficients $(\boldsymbol{\alpha},\beta)$ have been evaluated, the classification process for the information data can be accomplished by using Equation (4.7) and Equation (4.8), but with the whole range of training symbols.

4.4.3 Sparse LS-SVC based Equaliser

To alleviate the detection complexity resulted from the full spectrum of LS-SVC support values, some sort of sparseness can be imposed by using the pruning approach [96], whose procedure is as follows: train the classifier by the training data set; sort the spectrum of resulting classifiers' coefficients; remove the least important coefficients according to some acceptable degree in performance. This process can be terminated at this stage or extended to re-train the classifier by inputting the remaining corresponding data set. This pruning algorithm can be applied to the current application of the LS-SVC for DS-UWB equalisation.

As concluded from Subsection 4.4.2, a drawback of the LS-SVC in comparison with the original SVC formulation is that sparseness is lost in the LS-SVC case. This is because the support values are proportional to the errors at the data points, as can be seen in the third condition of Equation (4.16). However, by plotting the spectrum of the sorted support values, one can evaluate which data are the most significant for contribution to the LS-SVC classifier. Sparseness is imposed then by gradually omitting the least important data from the training set and re-estimating the LS-SVC. This algorithm can be summarised as [96]:

1. Train each of the M LS-SVCs on P pilot symbols.

2. Remove a predefined number (of sparseness ratio ρ) of pilot symbols that correspond to the smallest support values in the spectrum.

3. Re-train the LS-SVCs based on the reduced training set.

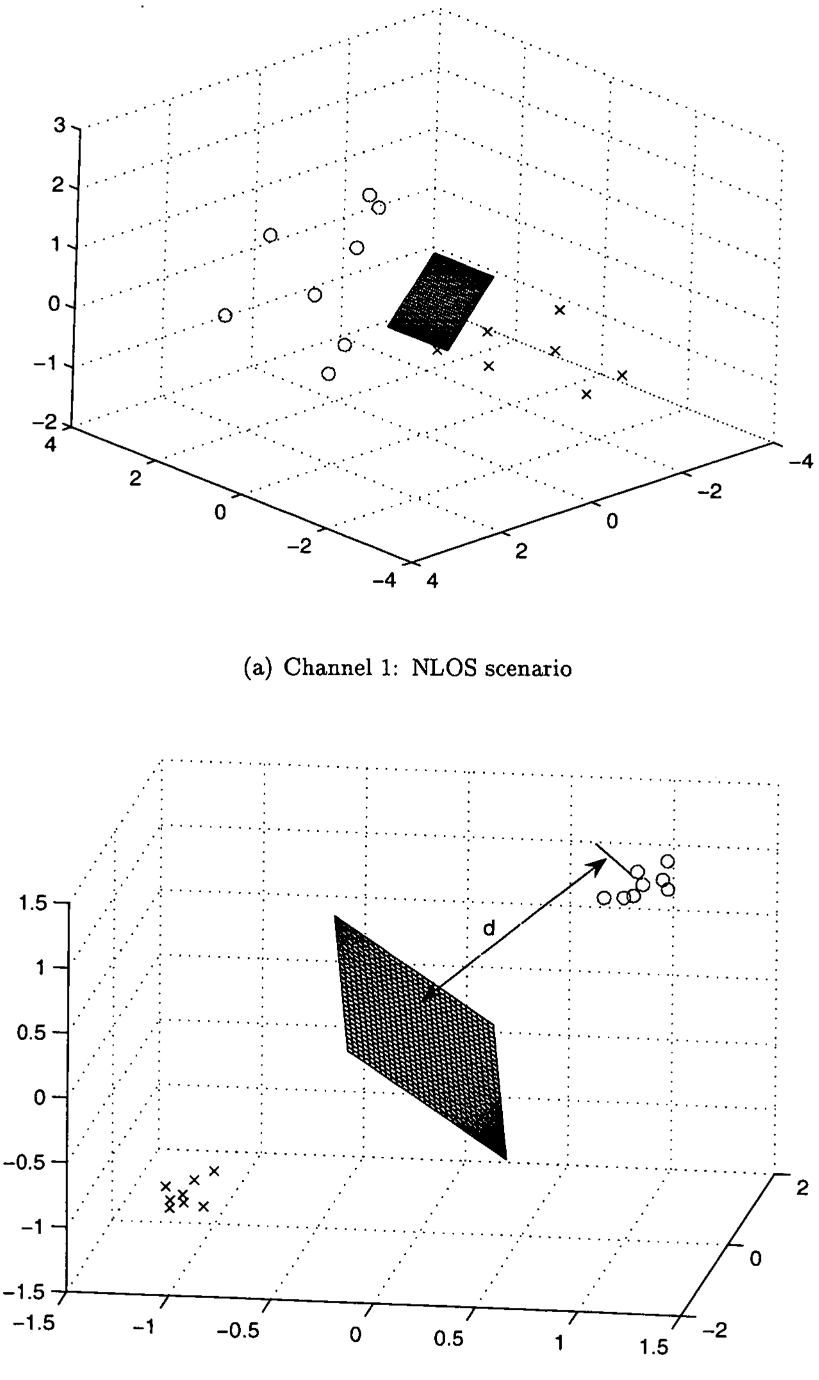
This procedure can also be implemented iteratively by removing a small part of training data until some preset performance index is reached. This represents pruning of the LS-SVC. As a result, the number of the classifier's coefficients becomes smaller, which reduces the detection complexity. It is worth noting that the pruning does not involve a computation of a Hessian matrix. Instead, it is immediately done based upon the physical meaning of the solution vector $\boldsymbol{\alpha}$.

4.4.4 Performance Analysis

In order to appreciate the performance advantages of the SVM based equalisation, we explain in this section the conceptual differences between conventional equalisation and equalisation as a pattern recognition solution. Considering the signal constellation for a digital communication signalling with ISI effects, the equalisation problem is to find the optimal mapping that returns the ISI corrupted signals to their original positions on the constellation. While in pattern recognition, the problem is to find the optimal decision boundary (a hyperplane in an appropriate higher dimensional space) that is used for symbol detection.

There are two main implications of the pattern recognition based equalisation. First, for an increasing number of observed symbols, a stable decision boundary can be obtained with a faster convergence speed, achieving a better detection accuracy. Second, the time-varying nature of multipath channels has small impact on the decision boundary, since the time variation can be viewed as a small noise added to the constellation in the detection regions of the learning machine.

A special case that clearly shows the superior performance of the SVM based equaliser is the line-of-sight (LOS) scenario, where most of the channel energy is concentrated on first components which results in two (for binary signalling) separable intensive clouds of observations in a high dimensional feature space with, most importantly, a proper kernel mapping. The detection then performs on average as a simple AWGN detection, if a well-chosen kernel function is used for the mapping. An illustrative example is shown in Figure 4.6, where two 4-tap FIR channel models with Rayleigh fading for each tap are used to visualise the signal constellation in the feature space for a binary signaling assuming no added noise. For an NLOS scenario (Channel 1), Figure 4.6(a), the average channel taps' powers are equal so that the signal centres are spread in the feature region. Hence, more classification errors occur potentially when the noise is added at the receiver. On the other hand, Figure 4.6(b) (Channel 2) shows the signal pattern for a LOS scenario where most of the channel power is concentrated on the first tap. With the assumption of unit average channel energy, the classification performance resembles the detection performance of the simple case of AWGN only, *i.e.*, the distance d in Figure 4.6(b) reflects the signal amplitude.



(a) Channel 1: NLOS scenario

(b) Channel 2: LOS scenario

Figure 4.6: An example of LOS vs NLOS signal constellation viewed in pattern space

4.4.5 Complexity Analysis

In this subsection, a complexity analysis is provided for the proposed SVM based equalisation, in terms of the number of multiplication operations in both the training and detection modes for a single training session, *i.e.,* one transmission packet. The analysis given in this section is normalised to one classifier, so that only a corresponding symbol of each block is considered.

Table 4.1 summarises the complexity of the proposed SVC, LS-SVC and sparse SVC based equalisers. The complexity of RAKE-MRC is also included as a benchmark. In the RAKE-MRC receiver, the training mode is used for channel estimation. The computational complexity of one training session is of $O(N_cPL_{est})$ multiplication operations. Whereas the detection complexity is of $O(N_cL_fB)$ operations for B detected symbols per packet.

Table 4.1: Complexity Comparison (N_c–Code length, P– Pilot size, B– Number of data blocks, L_f–RAKE fingers, $\bar{N}_{SV}$–Average number of Support Vectors, ρ–Sparseness ratio, and L_{est}–Length of the estimated channel)

	Training Complexity	Detection Complexity
RAKE with MRC	$O(N_cPL_{est})$	$O(L_fN_cB)$
SVC	$O(P^3)$	$O(N_{SV}N_cB)$
LS-SVC	$O((N_c+1)P^2)$	$O(PN_cB)$
Sparse-LSSVC		$O((1-\rho)PN_cB)$

For the standard SVM method, solving the QP problem for optimisation in a SVC training for P pilot symbols requires $O(P^3)$ multiplication operations plus N_cP^2 multiplications in generating Hessian matrix $\mathbf{K}$ for non-linear mapping [69], where N_c represents the dimension of the detector input space that is equal to the spreading code length in this system. Thus, the total number of multiplication operations in training the SVC can be approximated by $(N_cP^2 + P^3)$.

The most important reason for using LS-SVC is to reduce the training complexity over the standard SVC. In order to evaluate the multiplication operations in LS-SVC training, the linear equations system in Equation (4.17) requires $\frac{1}{2}(P+1)^2$ operations when using Gaussian elimination algorithm. In addition, one should consider another P multiplications for adjusting $\mathbf{\Omega}$ by $\gamma^{-1}\mathbf{I}$. Moreover, the same number of K-generating operations is considered here. Therefore, the total number of operations in LS-SVC

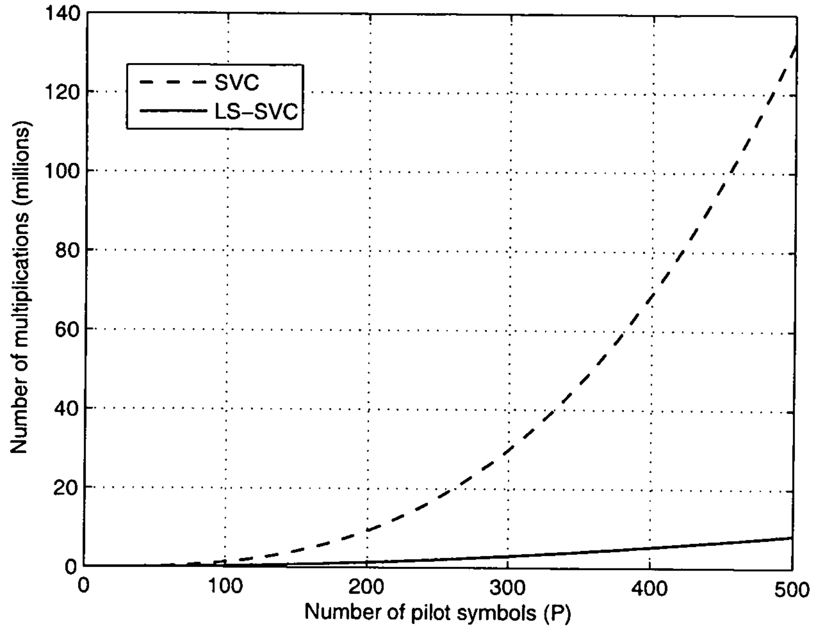


Figure 4.7: Training complexity comparison for SVM based equalisers

training would be on the order of ($N_cP^2 + P + \frac{1}{2}P^2 \approx (N_c + 1)P^2$).

For detection complexity, the numbers of multiplications for one detection session (B data symbols) can be calculated for each tested symbol as follows $O(\bar{N}_{SV}N_cB)$ for SVC and $O(PN_cB)$ for LS-SVC, where $\bar{N}_{SV}$ is the average number of support vectors of the SVC method. Since typically $\bar{N}_{SV} < P$, the detection computational complexity of the SVC is lower than that of the LS-SVC. However, practical experiments show that a large number of support vectors are obtained in DS-UWB systems.

Imposing sparseness will reduce the detection complexity of LS-SVC according to a predefined cutting ratio ρ - the ratio of the removed training points to the total training points. Hence, the detection complexity of the sparse LS-SVC is $O((1-\rho)PN_cB)$.

Figure 4.7 illustrates the training complexity comparison for both the SVC and LS-SVC based equalisers with respect to the number of pilot symbols.

For comparison purposes, define $\delta = \bar{N}_{SV}/P$ $(0 < \delta \leq 1)$. The overall complexity, *i.e.*, training and detection, saving of (sparse) LS-SVC over SVC can be evaluated by a ratio η, which is defined as

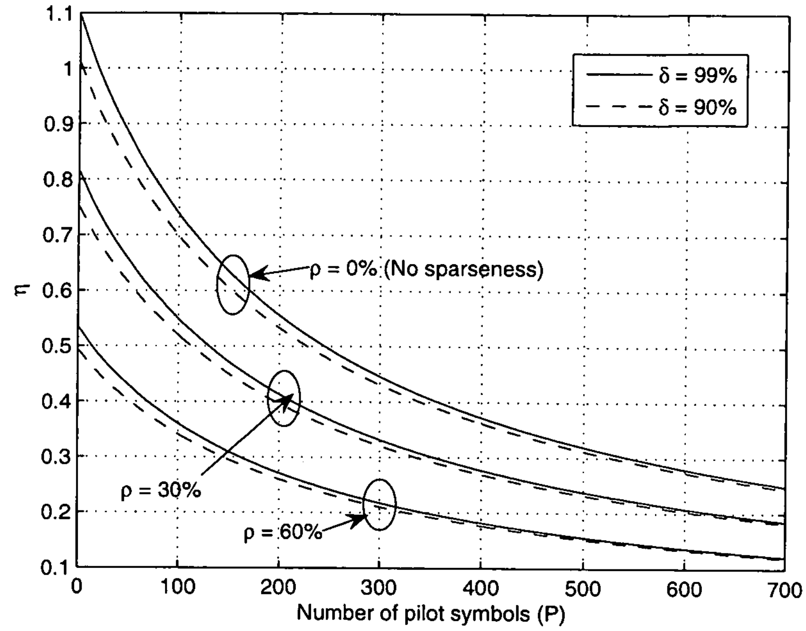


Figure 4.8: Complexity saving of LS-SVC over SVC, where δ is the average support vectors fraction and ρ is the removed pilot symbols ratio by imposing sparseness (sparseness ratio)

$$\eta = \frac{(Total\ Number\ of\ Multiplications)_{LS-SVC}}{(Total\ Number\ of\ Multiplications)_{SVC}}. \tag{4.19}$$

Figure 4.8 shows the reduction in the overall computational complexity for different numbers of pilot symbols considering two average numbers of support vectors and three cutting ratios of sparseness of $\rho = 0$ (no sparseness), 30% and 60%. The number of data symbols is assumed to be 6 times of the number of pilot symbols, *i.e.*, $B = 6P$.

4.4.6 Simulation Results

We use simulation results to demonstrate the performance of the proposed SVM based equalisers. The simulations were executed using two channel models proposed in IEEE802.15.3a [35]: CM1 for the case of line-of-sight (LOS), and CM3 with non-LOS (NLOS). We set the GI length to $L_{\mathrm{GI}} = 15$ for CM1, and $L_{GI} = 40$ for CM3, respectively, to match the maximum delay spreads of these two channels. The channels are assumed to be constant over one transmission session (one packet). We assume BPSK modulation with data rate varying from 138 Mbps to 163 Mbps. A ternary code [97] of length $N_c = 32$ is used

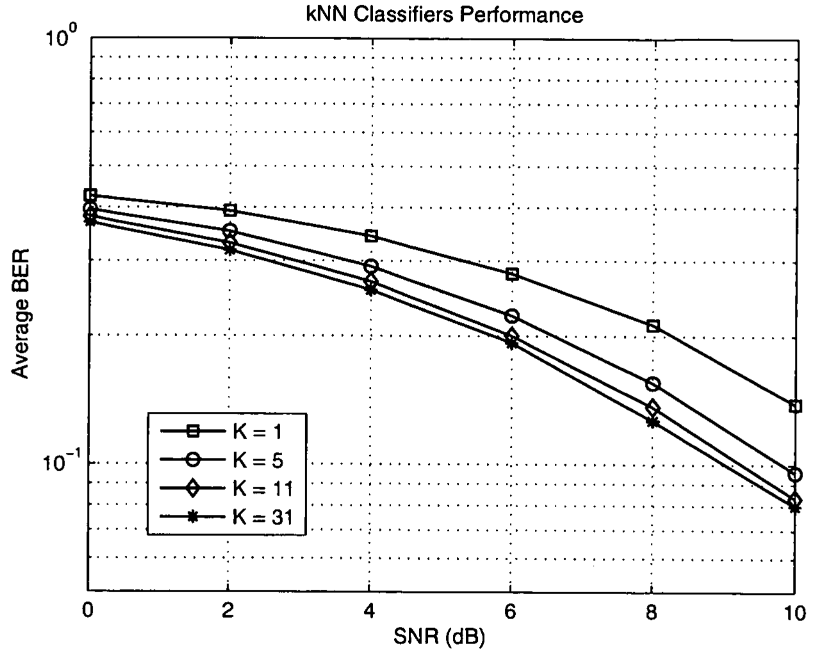


Figure 4.9: kNN based classification system performance for CM1

for spreading, with a chip width of $T_c = 0.167$ ns. The number of symbols per block is $M = 200$, and the sizes of pilot blocks per packet were chosen to be $P = 100$, 200, and 500. The number of data blocks per packet is $B = 2500$. The signal-to-noise ratio (SNR) is defined as the ratio between the average received signal power and the noise power.

For the sake of comparison, typical *k-Nearest Nieghbour* (kNN) based classifiers [70] were used as a benchmark reference of learning machine for CM1 and CM3 settings. Figure 4.9 shows the BER results of kNN based classifiers for CM1 with a range of values of K, *i.e.*, $1,\ 5,\ 11, \text{and}\ 31$. It can be clearly noticed from the results that better performance can be obtained for large values of K that represent the DS code length $(i.e., 31)$. Figure 4.10 shows the same results but for CM3 scenario. The optimal value of K in this scenario is varying depending on the SNR level, but, in general, we can choose $K = 31$ as an overall optimal value for later comparison. In both figures, it is obviously shown that classification in original attribute space is far worse than using nonlinear mapping through kernels as will be illustrated next.

Figure 4.11 depicts the BER performance of the proposed SVM based equalisers in comparison with the RAKE-MRC receiver and kNN based classifiers for CM1, where

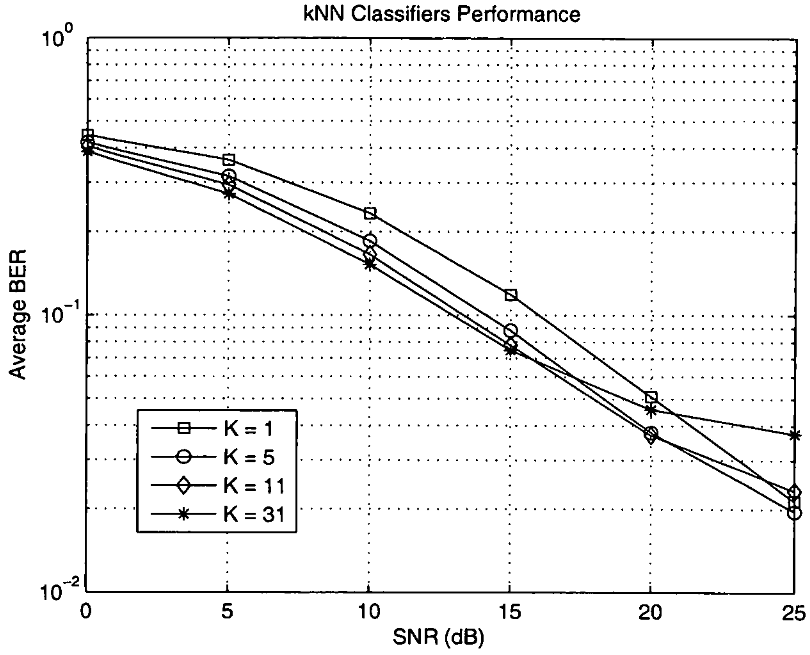


Figure 4.10: kNN based classification system performance for CM3

an LOS scenario is considered. The number of RAKE fingers was chosen to be the same as the number of pilot symbols of the classifiers, *i.e.*, $L_f = P = 100$, in order to fix the detection complexity for all equalisers. K was fixed to 31 for the kNN based classifiers. The SVM based equalisers significantly outperform the RAKE receiver and the kNN based classifiers. For instance, at $SNR = 10\,dB$, the average BER of the SVM based equalisers is around 2×10^{-5}, whereas it is 7×10^{-3} for RAKE receiver with perfect channel state information (CSI). As shown in Figure 4.11, the performance of the SVM based equalisers is nearly the same as the performance of an AWGN detector. This is because the data patterns (in high dimensional space) of the received chips are concentrated around far-apart centres that represent dominant LOS components, as described in Section 4.4.4.

The BER performance with CM3 is illustrated in Figure 4.12 where the number of RAKE fingers and the number of pilot symbols of the classifiers are set to be $L_f = P = 200$. Similar to Figure 4.11, the proposed SVM based equalisers outperform the RAKE receiver and the kNN based classifiers, at the same detection complexity. It is also inferred that, for medium-to-high SNR range, the cross-correlator channel estimator in the RAKE receiver results in an irreducible error floor at BER$\approx 10^{-3}$, while the SVM

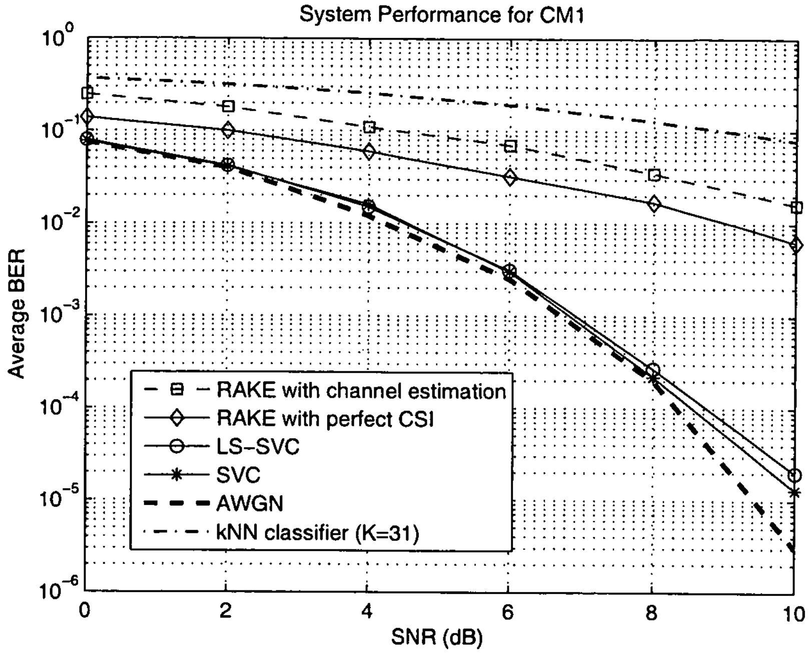


Figure 4.11: SVM system performance for CM1; Number of RAKE fingers L_f = Number of pilot symbols $P = 100$

based equalisers with implicit channel estimation even outperform the RAKE receiver with perfect CSI. Moreover, by introducing the LS-SVC based equalisers, the system performance is almost retained with the benefit of saving up to 50% of the training complexity, as shown in Figure 4.7.

The effect of imposing sparseness was examined for two levels of cutting ratios ($\rho =$ 30%, 60%) representing the detection complexity that could be saved. The effectiveness of the sparse LS-SVC based equaliser is illustrated in Figure 4.13 for three pilot sizes $P = 100, 200$ and 500. The results show that for a large enough size of pilot symbols ($P = 500$), sparse LS-SVC provides nearly the same performance as SVC, even with a reduction of 60% in detection complexity. The spectrum of the resulting average support values is depicted in Figure 4.14.

Choosing the appropriate number of symbols for training is a matter of a tradeoff between BER performance and bandwidth efficiency. Figure 4.15 illustrates the learning curve of the LS-SVC based equaliser with CM3, which can guide to choose the appropriate pilot size. For our simulations, we have chosen up to $P = 500$ pilot symbols so that no significant improvement is achieved beyond that size, compared to the entailed

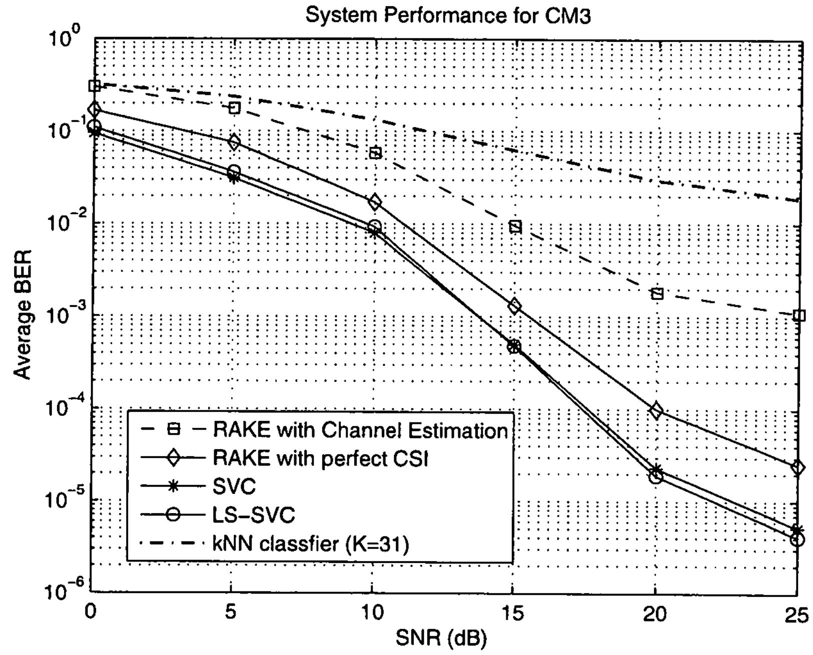


Figure 4.12: System performance for CM3; Number of RAKE fingers L_f = Number of pilot symbols $P = 200$

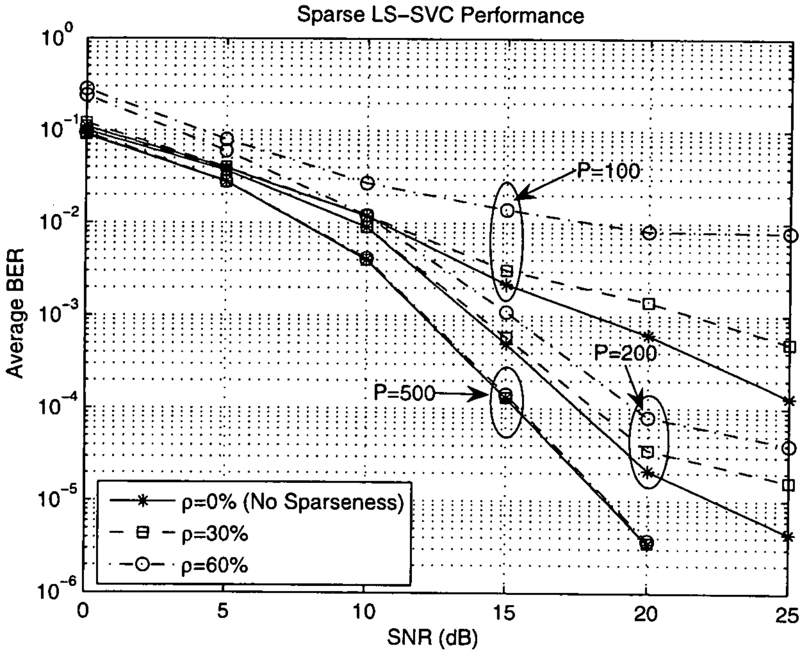


Figure 4.13: Sparse LS-SVC performance for different numbers of pilot symbols

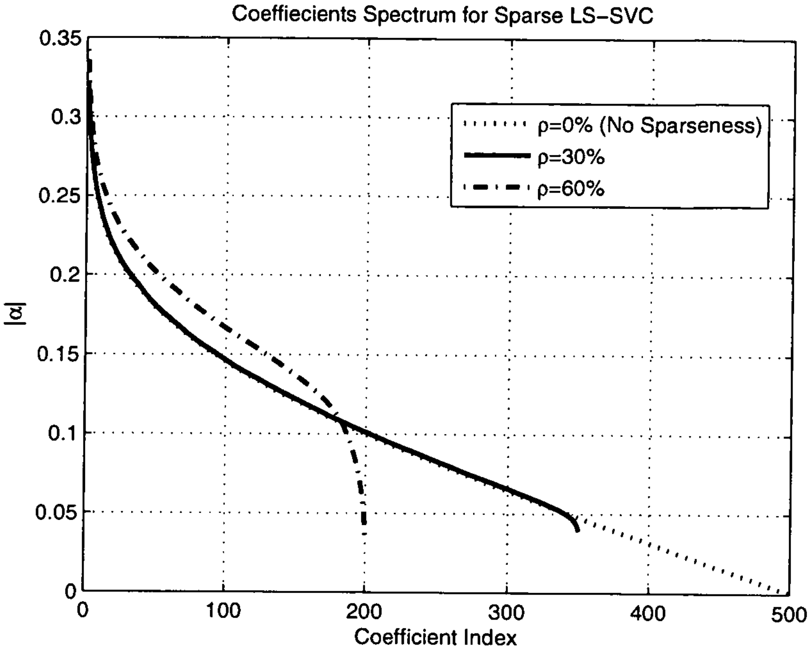


Figure 4.14: Support values spectrum

increase in complexity.

Due to the sensitivity of kernel parameters, an optimal width parameter (σ) value in the Gaussian RBF kernel is required. In the above simulations, empirical tests have been conducted to obtain the optimum value of σ. Figure 4.16 shows the effect of the value of σ on the performance of LS-SVC based equaliser with CM3, for different SNRs levels. It is interestingly noticed that the optimal value of σ is around $10^{0.5}$ for different SNR levels.

4.5 Summary

This chapter investigates the applications of SVM family of classification algorithms in the field of digital channel equalisation. The idea was developed from basic 2D scenario where an SVC based equaliser has been applied for digital wireless communication channel equalisation (*i.e.* symbol detection) and results have shown that the SVM approach is very effective in overcoming the ISI. Also, the nonlinear nature of signal constellation is effectively overcome by introducing kernel mapping.

Furthermore, the SVM techniques have been applied for DS-UWB channel equalisa-

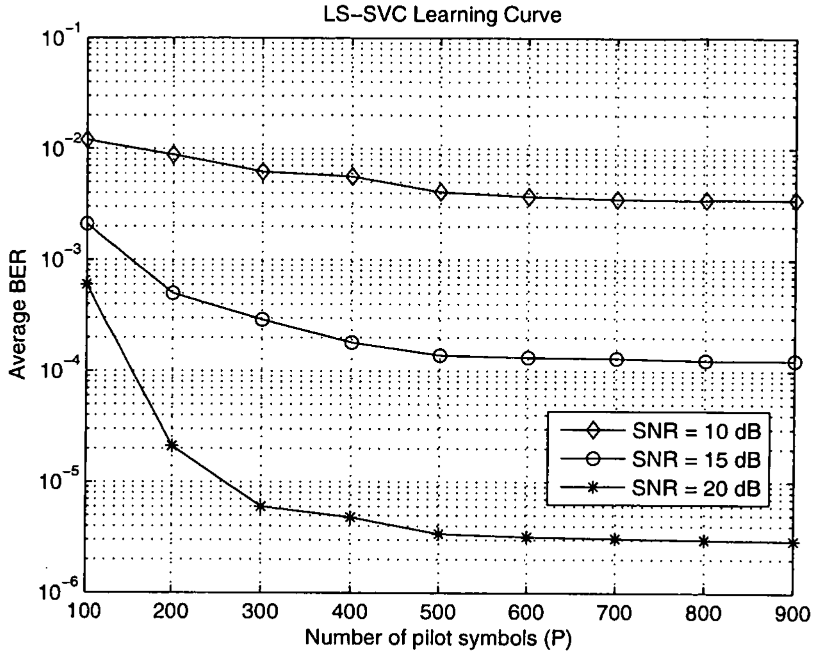


Figure 4.15: LS-SVC learning curve for CM3 at different SNR levels.

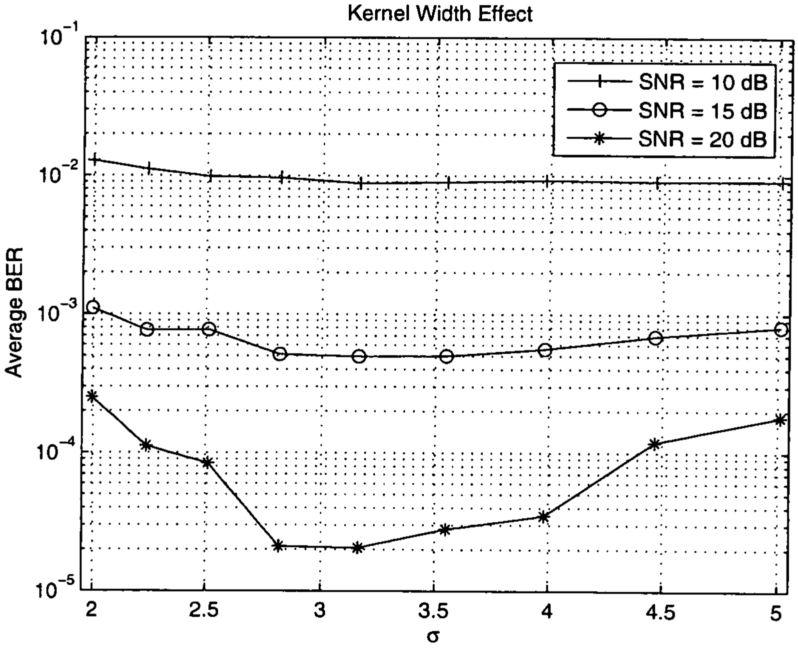


Figure 4.16: Choosing GRBF kernel parameter σ for SVM ($P = 200$)

tion. Results show superior performance of SVCs compared to the conventional RAKE receiver. In particular, the SVM based equalisation in the LOS scenario provides a close performance to the AWGN case. The LS-SVC based equalisers have been employed to reduce the training complexity of the standard SVC based receiver significantly, at a cost of small increase in detection complexity. The detection complexity, however, can be reduced by imposing some sparseness to the LS-SVC taps. Simulation results show that, with a relatively large number of pilot symbols, the proposed sparse LS-SVC based equaliser can save up to 60% of the detection complexity, while maintaining the superior BER performance of SVCs.

Chapter 5

Multi-Criteria Quadratic Programming based Equalisation

The optimisation in the SVM algorithm family is based on maximising only the margin that separates the classes, with some relaxation tolerance. In this chapter, a recent method called multi-criteria quadratic programming (MCQP) is investigated and applied to the channel equalisation and signal detection of communication systems. The organisation of this chapter is as follows. An introduction to the MCQP [98] is provided in Section 5.1, and detailed descriptions of MCQP learning model are presented in Section 5.2. An MCQP based equalisation method for nonlinear channels is proposed and presented in Section 5.3. Section 5.4 introduces and proposes the MCQP based equalisers for DS-UWB systems. Summary is provided in Section 5.5.

5.1 Introduction

The concept of MCQP methodology can be considered as an extension to the SVM methodology of the generalised learning theory presented in Chapter 3. MCQP enables a performance improvement in the optimisation module of the learning machine through the modification of the optimisation cost (*i.e.*, objective) function. In addition, MCQP is more computationally efficient, compared to standard SVMs, because the optimisation technique associated with MCQP only requires solving a set of linear equations. The preceding advantages raise the motivation to use MCQP for different digital channel equalisation applications.

5.2 MCQP Learning Model

The MCQP uses training data, similar to SVM, to estimate the decision function for the detection stage. The estimated function is, then, used to perform the classification to the testing transmission data. For a binary classification problem, the idea of MCQP model is based on maximising the external distance between the two classes' groups and minimising the internal distance within the same class group. This model has two significant advantages. The first is its relatively low complexity since it only needs to solve a set of linear equations. The second advantage is the performance enhancement due to the introduction of internal distance to the optimisation object function. Furthermore, kernel functions can also be used to solve nonlinear patterns. The following subsections develop the model formulation for linearly and nonlinearly separable patterns.

5.2.1 MCQP for Linearly Separable Patterns

Same as that in the SVC, the data patterns are separated by a hyperplane of direction represented by $\mathbf{w} = [w_1, w_2, ..., w_n]^T$, where n is the data pattern dimension, and a scalar distance β from the origin. Considering the training set $\{\bar{\mathbf{r}}_i, y_i\}_{i=1}^{P}$, the MCQP model is formulated as to

$$\begin{aligned} &\textit{minimise} \quad \tfrac{1}{2}\|\mathbf{w}\|^2 + A\textstyle\sum_{i=1}^{P}\alpha_i^2 - B\textstyle\sum_{i=1}^{P}\gamma_i \\ &\textit{subject to} \quad y_i(\bar{\mathbf{r}}_i^T\mathbf{w} - \beta) = -\alpha_i + \gamma_i, \ \ (i = 1, 2, ..., P) \end{aligned} \tag{5.1}$$

where A and B are arbitrary pre-defined model parameters that control the optimisation objectives. $\alpha_i, \gamma_i \geq 0$ represent the slack distances for misclassification errors and the distances of correctly classified points from the hyperplane, respectively. In other words, the aim of the optimisation problem in Equation (5.1) is to maximise the margin between different classes and to minimise the internal distance within the same class.

Assuming $\alpha_i = 0$ for correctly classified points and $\gamma_i = 0$ for misclassified points, and by introducing $\eta_i = \alpha_i - \gamma_i$, model of Equation (5.1) can be rewritten as

$$\begin{array}{ll} minimise & \frac{1}{2}\|\mathbf{w}\|^2 + \frac{1}{2}A\sum_{i=1}^{P}\eta_i^2 - B\sum_{i=1}^{P}\eta_i + \frac{1}{2}K\beta^2 \\ subject\,to & y_i(\bar{\mathbf{r}}_i^T\mathbf{w} - \beta) = -\eta_i, \ \ (i = 1, 2, ..., P), \end{array} \tag{5.2}$$

the new term $\left(\frac{1}{2}K\beta^2\right)$ in Equation (5.2) is introduced to add strong convexity to the objective function [98]. The weight K is an arbitrary positive number. By introducing Lagrange multipliers θ_i, The Lagrange function for the constrained optimisation in Equation (5.2) can be obtained as follows:

$$\begin{aligned} L_1(\mathbf{w}, \beta, \boldsymbol{\eta}, \boldsymbol{\theta}) = \ & \frac{1}{2}\|\mathbf{w}\|^2 + \frac{1}{2}A\sum_{i=1}^{P}\eta_i^2 - B\sum_{i=1}^{P}\eta_i \\ & + \frac{1}{2}K\beta^2 - \sum_{i=1}^{P}\theta_i[y_i(\bar{\mathbf{r}}_i^T\mathbf{w} - \beta) + \eta_i]. \end{aligned} \tag{5.3}$$

A matrix-vector notation is adopted for simplicity where $\boldsymbol{\theta} = [\theta_1, \theta_2, ..., \theta_P]^T$, and $\boldsymbol{\eta} = [\eta_1, \eta_2, ..., \eta_P]^T$. Further define the P-dimensional column vector $\mathbf{e} = [1, 1, ..., 1]^T$, the $P \times n$ matrix $\mathbf{R} = [\bar{\mathbf{r}}_1, \bar{\mathbf{r}}_2, ..., \bar{\mathbf{r}}_P]^T$ and the diagonal matrix $\mathbf{Y} = diag(y_1, y_2, ..., y_P)$. Then, the optimal solution of Equation (5.3) can be obtained by setting the derivatives of the Lagrange function to zeros, namely,

$$\begin{aligned} \tfrac{\partial}{\partial \mathbf{w}}L_1 &= \mathbf{w} - \mathbf{R}^T\mathbf{Y}\boldsymbol{\theta} = 0, \\ \tfrac{\partial}{\partial \beta}L_1 &= K\beta - \mathbf{e}^T\mathbf{Y}\boldsymbol{\theta} = 0, \\ \tfrac{\partial}{\partial \eta}L_1 &= A\boldsymbol{\eta} + B\mathbf{e} - \boldsymbol{\theta} = 0, \\ \tfrac{\partial}{\partial \theta}L_1 &= \mathbf{Y}(\mathbf{R}\mathbf{w} - \beta\mathbf{e}) + \boldsymbol{\eta} = 0. \end{aligned} \tag{5.4}$$

Thus, by simple manipulation to the equations in Equation (5.4), the optimal Lagrange multiplier vector can be expressed as

$$\boldsymbol{\theta} = \left[\frac{1}{A}\mathbf{I} + \mathbf{Y}\left(\mathbf{R}\mathbf{R}^T + \frac{1}{K}\mathbf{e}\mathbf{e}^T\right)\mathbf{Y}\right]^{-1}\left[\frac{B}{A}\mathbf{e}\right]. \tag{5.5}$$

Therefore, the optimal solution can be obtained by substituting Equation (5.5) into the first two equations of Equation (5.4)

$$\begin{aligned} \mathbf{w} &= \mathbf{R}^T\mathbf{Y}\boldsymbol{\theta} \\ \beta &= \tfrac{1}{K}\mathbf{e}^T\mathbf{Y}\boldsymbol{\theta} \end{aligned} \tag{5.6}$$

5.2.2 MCQP for Nonlinearly Separable Patterns

For nonlinear separable clouds of data, the kernel mapping [85] is utilised. By applying nonlinear mapping, through the transformation function $\phi(.)$, the original linearly nonseparable input data space is transformed to a high dimension linearly separable feature space. The kernel mapping, however, can be realised as the inner product of the higher-dimensional feature vectors without explicitly knowing ϕ. Therefore, the kernel mapping is defined as

$$K(\bar{\mathbf{r}}_i, \bar{\mathbf{r}}_j) = \phi(\bar{\mathbf{r}}_i)^T \phi(\bar{\mathbf{r}}_j), \tag{5.7}$$

and

$$\mathbf{K}(\mathbf{R}, \mathbf{R}^T) = \begin{bmatrix} K(\bar{\mathbf{r}}_1, \bar{\mathbf{r}}_1) & K(\bar{\mathbf{r}}_1, \bar{\mathbf{r}}_2) & \cdots & K(\bar{\mathbf{r}}_1, \bar{\mathbf{r}}_P) \\ K(\bar{\mathbf{r}}_2, \bar{\mathbf{r}}_1) & \ddots & & \vdots \\ \vdots & & \ddots & \vdots \\ K(\bar{\mathbf{r}}_P, \bar{\mathbf{r}}_1) & & \cdots & K(\bar{\mathbf{r}}_P, \bar{\mathbf{r}}_P) \end{bmatrix}. \tag{5.8}$$

From the model of Equation (5.2) and its optimal conditions in Equation (5.4), and by substituting $\mathbf{w} = \mathbf{R}^T\mathbf{Y}\boldsymbol{\theta}$, and replacing $\mathbf{R}\mathbf{R}^T$ by $\mathbf{K}(\mathbf{R}, \mathbf{R}^T)$, we can reformulate the model as in Equation (5.9) below

$$\begin{aligned} &minimise \quad \tfrac{1}{2}\left\|\boldsymbol{\theta}\right\|^2 + \tfrac{1}{2}A\textstyle\sum_{i=1}^P \eta_i^2 - B\sum_{i=1}^P \eta_i + \tfrac{1}{2}K\beta \\ &subject\, to \quad \mathbf{Y}\left(\mathbf{K}(\mathbf{R}, \mathbf{R}^T)\mathbf{Y}\boldsymbol{\theta} - \beta\mathbf{e}\right) = -\boldsymbol{\eta}. \end{aligned} \tag{5.9}$$

The Lagrange function of this model can be expressed as

$$\begin{aligned} L_2(\boldsymbol{\theta}, \beta, \boldsymbol{\eta}, \boldsymbol{\rho}) = \; & \tfrac{1}{2}\left\|\boldsymbol{\theta}\right\|^2 + \tfrac{1}{2}A\textstyle\sum_{i=1}^P \eta_i^2 - B\sum_{i=1}^P \eta_i + \tfrac{1}{2}K\beta^2 \\ & -\boldsymbol{\rho}^T\left[\mathbf{Y}\left(\mathbf{K}(\mathbf{R}, \mathbf{R}^T)\mathbf{Y}\boldsymbol{\theta} - \beta\mathbf{e}\right) + \boldsymbol{\eta}\right], \end{aligned} \tag{5.10}$$

where $\boldsymbol{\rho} = [\rho_1, \rho_2, ..., \rho_P]^T$are Lagrange multipliers for nonlinear scenario. The optimality conditions of the model of Equation (5.10) are then expressed by

$$\begin{aligned}
\tfrac{\partial}{\partial \boldsymbol{\theta}} L_2 &= \boldsymbol{\theta} - \mathbf{Y}\left(\mathbf{K}(\mathbf{R}, \mathbf{R}^T)\right)^T \mathbf{Y}\boldsymbol{\rho} = 0, \\
\tfrac{\partial}{\partial \beta} L_2 &= K\beta + \mathbf{e}^T \mathbf{Y}\boldsymbol{\rho} = 0, \\
\tfrac{\partial}{\partial \boldsymbol{\eta}} L_2 &= A\boldsymbol{\eta} - B\mathbf{e} - \boldsymbol{\rho} = 0, \\
\tfrac{\partial}{\partial \boldsymbol{\rho}} L_2 &= \mathbf{Y}\left(\mathbf{K}(\mathbf{R}, \mathbf{R}^T)\mathbf{Y}\boldsymbol{\theta} - b\mathbf{e}\right) + \boldsymbol{\eta} = 0.
\end{aligned} \tag{5.11}$$

Hence, the optimal solution is given by

$$\begin{aligned}
&\boldsymbol{\rho} = \left[\tfrac{1}{A}\mathbf{I} + \mathbf{Y}\left(\mathbf{K}(\mathbf{R}, \mathbf{R}^T)\left(\mathbf{K}(\mathbf{R}, \mathbf{R}^T)\right)^T + \tfrac{1}{K}\mathbf{e}\mathbf{e}^T\right)\mathbf{Y}\right]^{-1}\left[\tfrac{B}{A}\mathbf{e}\right], \\
&\beta = \tfrac{1}{K}\mathbf{e}^T\mathbf{Y}\boldsymbol{\rho}
\end{aligned} \tag{5.12}$$

5.3 MCQP based Equaliser for Nonlinear Channels

In this section, the low complexity MCQP based approach is proposed for nonlinear channel equalisation or, more appropriately, signal detection in wireless communications. The proposed system model is presented with simulations to show the performance and complexity improvements over SVM. Simulation results confirm the performance enhancement of the proposed MCQP over standard SVM based equaliser. It also provides a performance close to that of the optimal Bayesian detector. Furthermore, the MCQP based equaliser considerably demonstrates its robustness to the time variation effects of channel coefficients.

5.3.1 System Model

The communication system used in this experiment is shown in Figure 5.1. Assuming baseband transmission and perfect symbol matching filtering associated with real valued data, a discrete-time real channel can be considered to fit the learning based equalisers adopted. The channel model, according to [13], consists of a deterministic term $y_p(k)$ and random process term $v(k)$ which represent AWGN samples. The deterministic term, Equation (5.14), is a polynomial combination of order P_c of a linear FIR filter with length L, which is defined in Equation (5.13). Hence, for a transmit symbol

$d(k) \in \{+1, -1\}$, the output of a general form of nonlinear channel can be modelled as follows

$$y_l(k) = \sum_{l=0}^{L-1} h_l(k) d(k-l) \tag{5.13}$$

and

$$y_p(k) = \sum_{i=0}^{P_c} c_i y_l^i(k), \tag{5.14}$$

where c_i is a real coeffiecient. The channel output can be modelled as

$$r(k) = y_p(k) + v(k). \tag{5.15}$$

The channel output in Equation (5.15) can be grouped into vectors of length n as

$$\mathbf{r}(k) = [r(k),\ r(k-1), ..., r(k-n+1)]^T \tag{5.16}$$

where n is the dimension of received signal vectors that is chosen to match the length of the channel so that the equaliser output in Figure 5.1 is dependent on the length of the ISI channel (*i.e.* $n = L$). This means that the number of channel states (signal constellation) for the binary detection is 2^{n+1} if no AWGN is added.

Optimal Bayesian detector

The utilised optimal detector in this study is the Bayesian or maximum a posterior (MAP) equaliser. The binary decision role for the Bayesian detector is presented here. Bayesian equaliser works in a symbol-by-symbol manner, with the aim of maximising the posterior of symbol $d(k)$ is being transmitted, given the likelihood and the priori of the observed signal [99].

Given a set of noise-free received vectors (*i.e.* channel states) $\{\mathbf{r}_i^+, \mathbf{r}_i^-\}$, the decision rule is to choose the optimal Bayesian symbol ($\hat{d}(k)$) for a noisy received vector ($\mathbf{r}(k)$, *or* $\mathbf{r}$ for simple notation). $\hat{d}(k)$ is estimated by

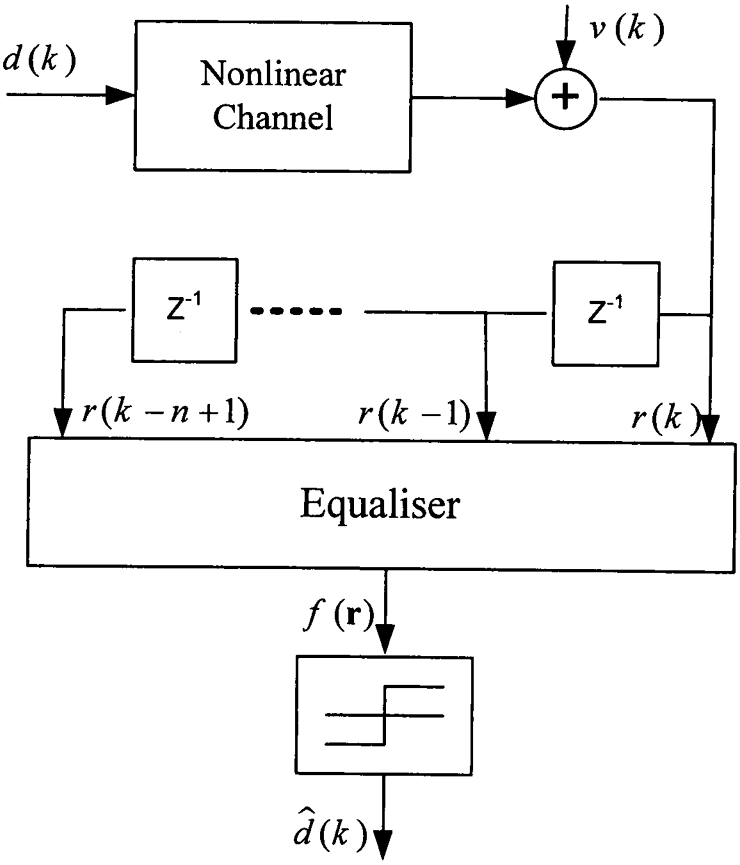


Figure 5.1: Discrete-time system model

$$\hat{d}(k) = \text{sign}\{f_{BAYES}(\mathbf{r})\} = \begin{cases} +1, & f_{BAYES}(\mathbf{r}) \geq 0 \\ -1, & f_{BAYES}(\mathbf{r}) < 0 \end{cases} \tag{5.17}$$

where sign denotes the decision function, and the optimal Bayesian function is given by

$$\begin{aligned} f_{BAYES}(\mathbf{r}) = & \textstyle\sum_{i=1}^{N^+} \exp\left(-\left\|\mathbf{r}-\mathbf{r}_i^+\right\|^2/2\sigma^2\right) \\ & -\textstyle\sum_{i=1}^{N^-} \exp\left(-\left\|\mathbf{r}-\mathbf{r}_i^-\right\|^2/2\sigma^2\right) \end{aligned} \tag{5.18}$$

where $\mathbf{r}_i^\pm = [y_p(k), y_p(k-1), ..., y_p(k-n+1)]$ for $d(k) = \pm 1$, and $1 \leq i \leq N^\pm$ respectively. N^+ and N^- in Equation (5.18) refer to the number of channel states for +1, -1 symbols (in this application, $N^+ = N^- = 2^n$). σ^2 denotes the AWGN power. The Bayesian decision function in Equation (5.18) assumes equiprobable a priori probabilities and a binary decision solution.

5.3.2 Symbol Detection by MCQP based Equaliser

For digital communication channel equalisation, the training stage is accomplished through transmitting pilot symbols ($\mathbf{Y}$), and receiving their corresponding channel outputs ($\mathbf{R}$). Thus, by applying Equation (5.12), the equaliser parameters are evaluated. Then, the symbol estimation, for an observed received channel output ($\mathbf{r}$), is evaluated by the function in Equation (5.20) for detection process. First, define

$$\mathbf{K}(\mathbf{r}, \mathbf{R}^T) = [K(\mathbf{r}, \bar{\mathbf{r}}_1)\; K(\mathbf{r}, \bar{\mathbf{r}}_2) \;\; K(\mathbf{r}, \bar{\mathbf{r}}_P)], \tag{5.19}$$

then, the evaluation function can be expressed as

$$f_{MCQP}(\mathbf{r}) = \left(\mathbf{K}(\mathbf{r}, \mathbf{R}^T)\left(\mathbf{K}(\mathbf{r}, \mathbf{R}^T)\right)^T + \frac{1}{K}\mathbf{e}^T\right)\mathbf{Y}\rho \tag{5.20}$$

where, for our binary signalling, the estimate of $d(k)$ decision is given by

$$\hat{d}(k) = \text{sign}\{f_{MCQP}(\mathbf{r})\} = \begin{cases} +1, & f_{MCQP}(\mathbf{r}) \geq 0 \\ -1, & f_{MCQP}(\mathbf{r}) < 0. \end{cases} \tag{5.21}$$

5.3.3 Simulation Results

In this subsection, we present the computer simulation configurations and results of the proposed system. To simplify visualisation, a channel model of 2 FIR taps were used (*i.e.* $L = n = 2$), and the general channel output is defined as [100]

$$r(k) = y(k) + \mu y^3(k) + v(k), \tag{5.22}$$

where

$$y(k) = h_0(k)d(k) + h_1(k)d(k-1). \tag{5.23}$$

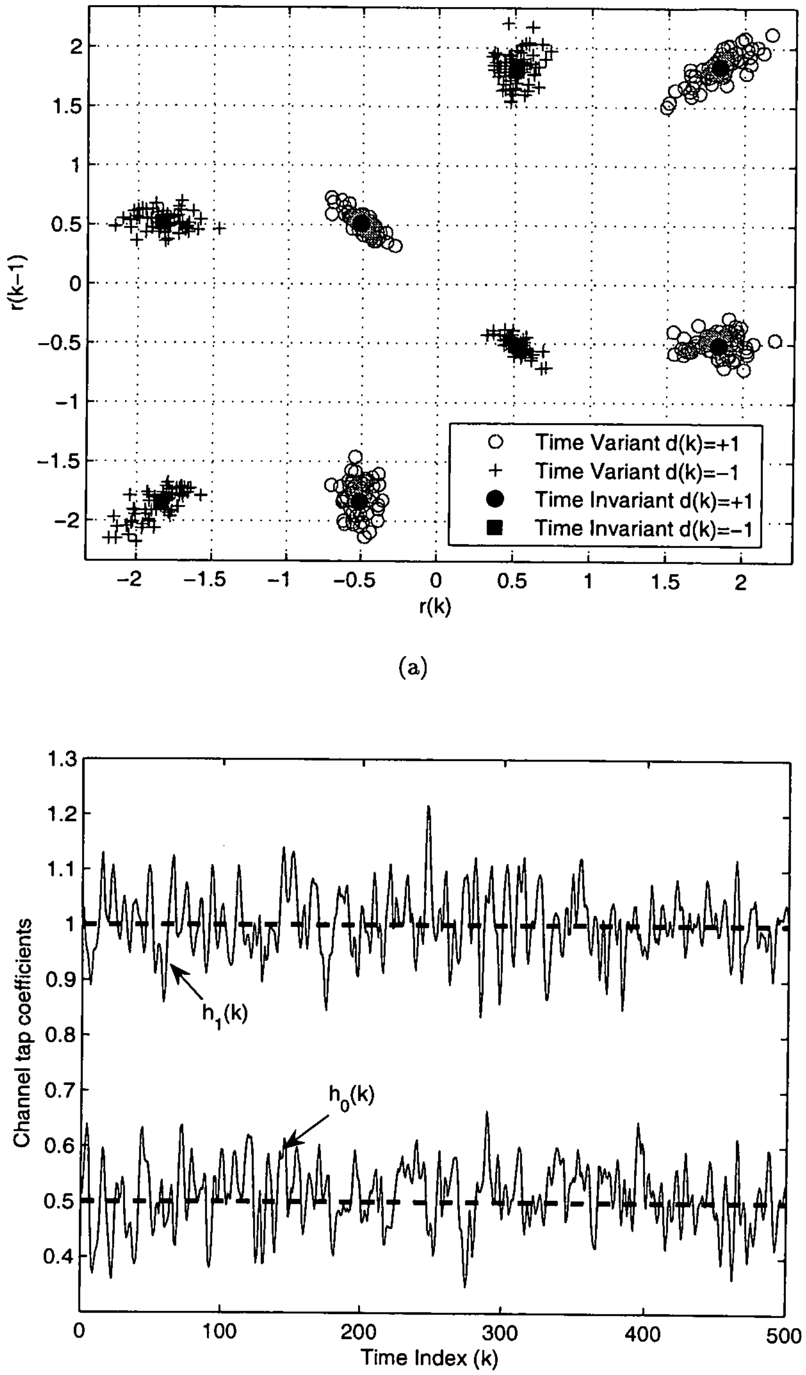
For all simulations, μ is empirically chosen to be 0.1 and $[h_0(k),\ h_1(k)] = [0.5,\ 1]$ for time invariant model. For time variant scenario, and according to [101], $h_0(k)$, $h_1(k)$ are two time-varying coefficients. These coefficients were generated by passing AWGN of variance $\sigma_n^2 = 0.01$, and centred around [0.5, 1], through a Butterworth low pass filter. The normalised cuttoff frequency (f_D) is 0.15 representing a Doppler shift relative to symbol rate. The noise-free channel states, for both time invariant and time variant, are shown in Figure 5.2(a). The FIR channel coefficients variations are plotted in Figure 5.2(b).

The MCQP parameters settings for simulations are as follows: $A = 0.9$, $B = 0.1$and $K = 0.5$ based on empirical tests. The number of training symbols (pilots) is set to $P = 100$. For nonlinear transformation by kernel function, the GRBF is used as a preferred kernel for communication applications [12]. The GRBF is rewritten here, for convenience, as

$$K(\mathbf{r}_a, \mathbf{r}_b) = \exp\left(\frac{-\left\|\mathbf{r}_a - \mathbf{r}_b\right\|^2}{2\sigma^2}\right) \tag{5.24}$$

where σ is the kernel width parameter that is proportional to standard deviation of the AWGN. The SVC is used for comparison. The controlling parameter for the SVC (C) is empirically set to be 10. The kernel function for the SVC is chosen to be the same one used in the proposed MCQP for fair comparison.

The computational complexity of the proposed equaliser for training is tested in terms of computer execution time. Figure 5.3 shows a comparison between the proposed



(a)

(b)

Figure 5.2: Channel models; (a) Signal constellation for noise free channel states. (b) Channel taps coefficients.

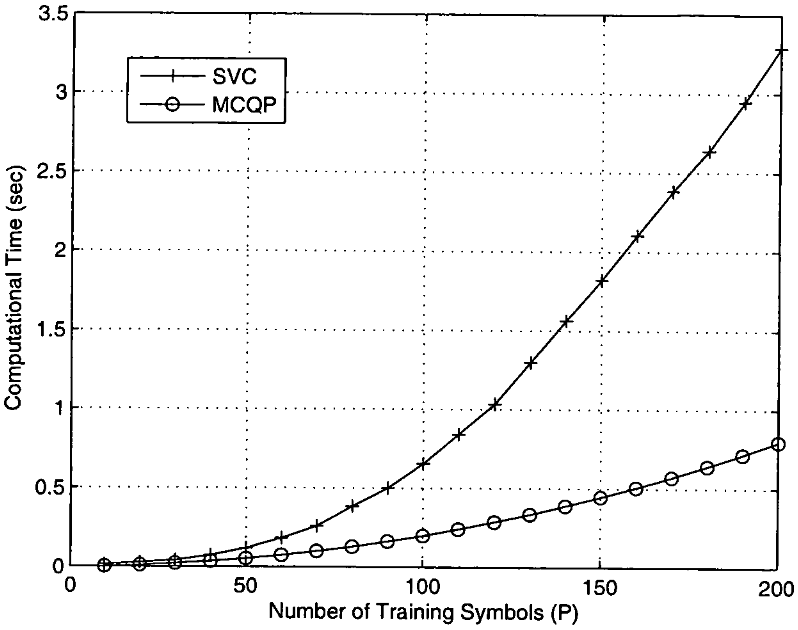


Figure 5.3: Training computational complexity comparison.

equaliser and SVC equaliser training times for different pilot sizes. Results in Figure 5.3 confirm the massive reduction in the training complexity by the proposed equaliser, in comparison with the SVC equaliser. The training complexity of the MCQP based equaliser increases almost linearly with respect to the training pilot size (*i.e.* $\approx O(P)$), while the training complexity of the SVC is quadratic This is a massive reduction compared to that of SVC where the trend follows a quadratic in the training pilot size (*i.e.* $O(P^2)$). On the other hand, the detection complexity of the MCQP based equaliser is higher than that of the SVC.

The resulting BER curves of the Bayesian, SVC and MCQP based equalisers are depicted in Figure 5.4. The results show the superb performance of the proposed equaliser and its convergence to the optimal Bayesian detector, especially for SNR levels over 10 dB. It is also shown that the high capability of tracking the time variation with the proposed equaliser, where the MCQP performance for time varying channel, with fast channel variation, is almost identical to the SVC performance for the corresponding static channel.

Figure 5.5 shows the learning curves of the proposed equaliser. By stating learning curve, it is meant to describe finding the sufficient number of training data that

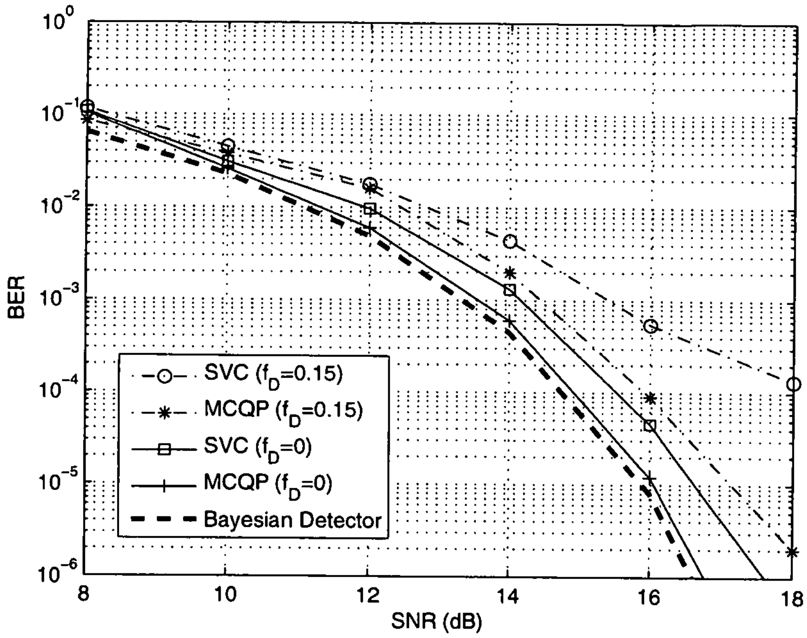


Figure 5.4: BER performance of MCQP equaliser

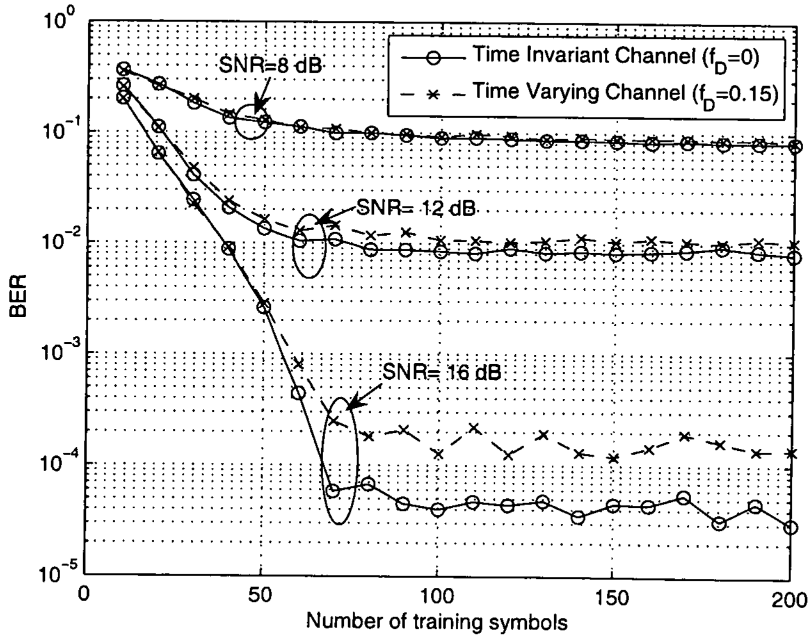


Figure 5.5: MCQP learning curves

guarantees the best performance for a particular condition. The results in Figure 5.5 consider three levels of SNR (8, 12 and 16 dB) for both the time invariant and time variant scenarios. Learning curves converge after approximately 80 training pilot symbols. Hence, a pilot of size 100 was chosen in the simulations. Moreover, curves confirm the comparable performance between time invariant and time variant scenarios.

5.4 MCQP based Equalisers for DS-UWB Systems

This section proposes and investigates the MCQP based equalisation for DS-UWB systems. Here, the ML units in Figure 4.5 are replaced by MCQP based classifiers. A sparse version of the MCQP based equaliser is also proposed in this investigation. The following subsections describe these equalisers in details, and present and discuss the simulation results compared to conventional receivers in DS-UWB systems.

5.4.1 MCQP based Equaliser

As mentioned in the general receiver structure in Section 4.3, a group of M parallel MCQP based equalisers are used in the receiver. In training mode, and for each equaliser, the input to the m^{th} equaliser can be defined as $\bar{\mathbf{r}}_j = \mathbf{x}_j^{(m)}$, as in Equation (4.6). Now, we construct $\mathbf{R} = [\bar{\mathbf{r}}_1, \bar{\mathbf{r}}_2, ..., \bar{\mathbf{r}}_P]^T$, $\mathbf{K}(\mathbf{R}, \mathbf{R}^T)$ as in Equation (5.8) and $\mathbf{Y} = diag(y_1, y_2, ..., y_P)$. y_i represents the pilot symbols of this m^{th} equaliser. The aim of training in MCQP based equalisation is to estimate the coefficients vector ($\boldsymbol{\rho}$) from Equation (5.12) and β from Equation (5.11). The resulting estimates are then applied in Equation (5.20) and the detected symbol is given by Equation (5.21).

5.4.2 Sparse MCQP based Equaliser

The sparseness imposing procedure in the MCQP based equaliser is identical to that in the sparse LS-SVC based equaliser, which is described in Subsection 4.4.3. Here, the resulting MCQP based equaliser coefficients vector ($\boldsymbol{\rho}$) is sorted according to absolute values and truncated to the specified level (cutting ratio).

5.4.3 Complexity Analysis

In this subsection, the complexity analysis of the MCQP based equaliser is provided. As in Section 4.4.5, the analysis is also in terms of the number of multiplication operations in both the training and detection modes for a single training session. All analyses are normalised to one classifier, so that only a corresponding symbol of each block is considered.

The MCQP based equaliser is trained by solving a linear equation system as discussed in Subsection 5.2.2. This is similar to the case in the LS-SVC based equaliser. However, an additional matrix multiplication term is added in Equation (5.12), for the constructed kernel matrix resulting in additional $O(p^2)$ operations, compared to to the LS-SVC. Therefore, the training complexity of the proposed MCQP based equaliser is approximately $O([N_c + \frac{3}{2}]P^2)$.

For detection complexity, the numbers of multiplications for one detection session (B data symbols) can be calculated for each tested symbol in MCQP as $O(PN_cB)$, which is the same as the detection complexity of the LS-SVC based equaliser. Similarly to the sparse LS-SVC based equaliser, imposing sparseness will reduce the detection complexity of MCQP according to a predefined cutting ratio ρ. Hence, the detection complexity of sparse MCQP is also $O((1-\rho)PN_cB)$.

5.4.4 Simulation Results

This section presents simulation settings and results to demonstrate the performance of the proposed MCQP based receivers for DS-UWB system. For fair comparison purposes, similar settings and assumptions to the proposed SVM based receiver in Section 4.4 are considered. We will briefly repeat the most important settings for convenience. A BPSK signalling is adopted for CM1 and CM3 of the IEEE802.15.3a report [35]. The GI were set to 15 for CM1 and 40 for CM3. A similar ternary code of length $N_c = 32$ is used for spreading. The number of symbols per block is $M = 200$, and the sizes of pilot blocks per packet were chosen to be $P = 100$, 200, and 500. The number of data blocks per packet is $B = 2500$, representing a period of 16 μs, and the channel is assumed to be constant within this period. The MCQP parameters were empirically

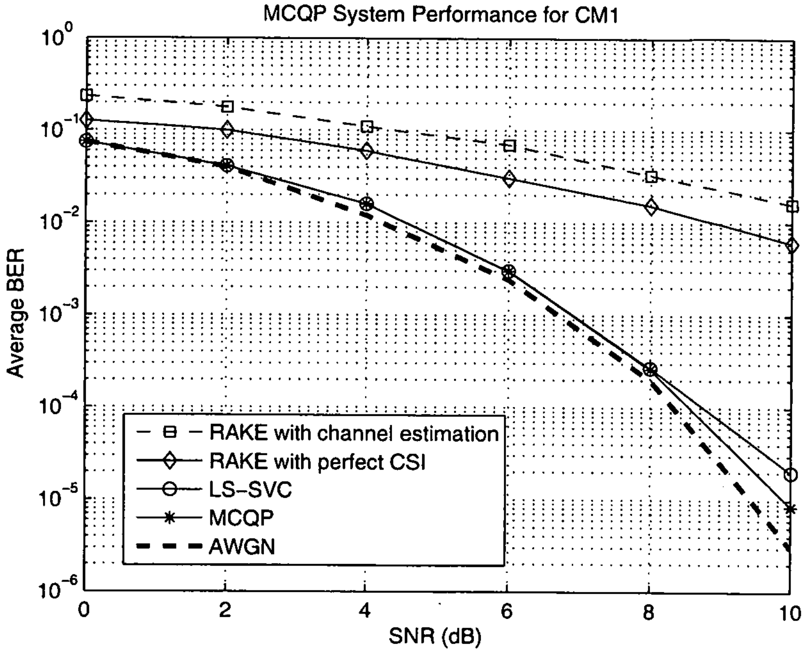


Figure 5.6: MCQP system performance for CM1; Number of RAKE fingers L_f = Number of pilot symbols $P = 100$

set to: $A = 0.85$, $B = 0.15$, and $K = 0.5$, although changing these values does not considerably affect the performance.

For the LOS scenario of CM1 settings, Figure 5.6 depicts the BER performance of the MCQP based receiver in comparison with the RAKE receiver for CM1, in both perfect CSI and with channel estimation. Also, the performance of LS-SVC based receiver is shown. For same detection complexity settings, the number of RAKE fingers was chosen to be the same as the number of pilot symbols of the classifiers, *i.e.*, $L_f = P = 100$. The proposed MCQP based receiver has also shown superior performance, as in LS-SVC based receiver case, that is very close to the pure AWGN channel. It even outperforms the LS-SVC based receiver for higher SNR levels (*e.g.* at $10\,dB$).

For the NLOS scenario of CM3, the BER performance of the MCQP based receiver is illustrated in Figure 5.7 where the number of RAKE fingers and the number of pilot symbols of the classifiers are set to be $L_f = P = 200$. By comparison to the conventional RAKE receiver and the FDE, with and without channel estimation, the MCQP based receiver is obviously excellent. The MCQP based receiver even better outperforms the RAKE receiver with perfect CSI if similar complexity settings were considered, *i.e.*, the

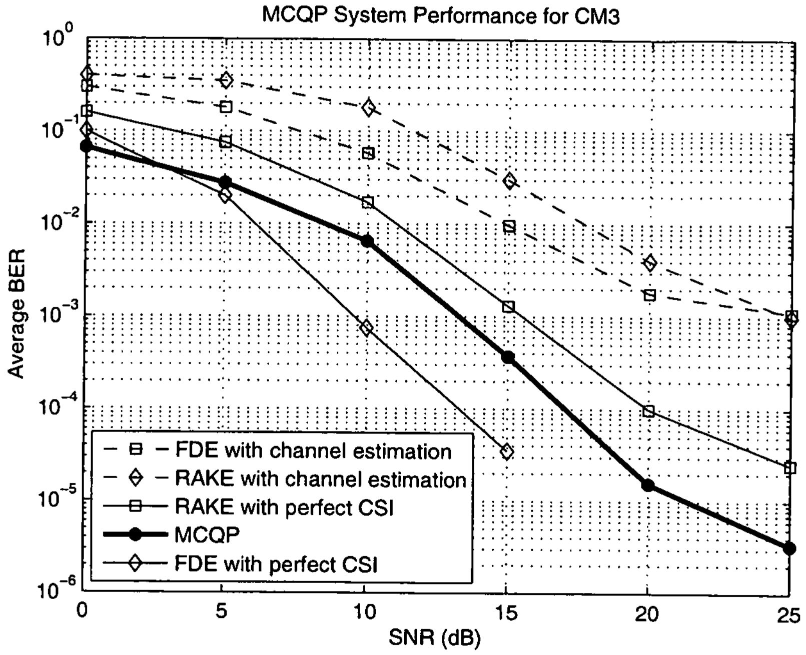


Figure 5.7: MCQP system performance for CM3; Number of RAKE fingers L_f = Number of pilot symbols $P = 200$

number of RAKE fingers = the number of pilot symbols. The optimal performance reference is for the FDE with perfect CSI.

Figure 5.8 illustrates the effects of imposing sparseness to the MCQP based receiver for CM3 considering the same two levels of cutting ratios ($\rho = 30\%$, 60%) of the sparse LS-SVC based receiver. Also, the sparseness effectiveness was examined for three pilot sizes $P = 100, 200$ and 500. The results show that imposing sparseness is negatively, comparing to the sparse LS-SVC, affecting the performance. Even for a large enough size of pilot symbols ($P = 500$), where the sparse MCQP exhibits performance degradation, unlike the case of the sparse LS-SVC based receiver.

Learning curves provide good guidance to the appropriate number of training symbols needed. Figure 5.9 illustrates the learning curves of the MCQP based receiver for CM3, under three SNR levels. The results of Figure 5.9 show that the optimal number of training symbols for the MCQP based receiver is approximately 400. This should be compared with the results of Figure 4.15 for the LS-SVC based receiver, which indicates that the optimal number of training symbols for the LS-SVC based receiver is approximately 600.

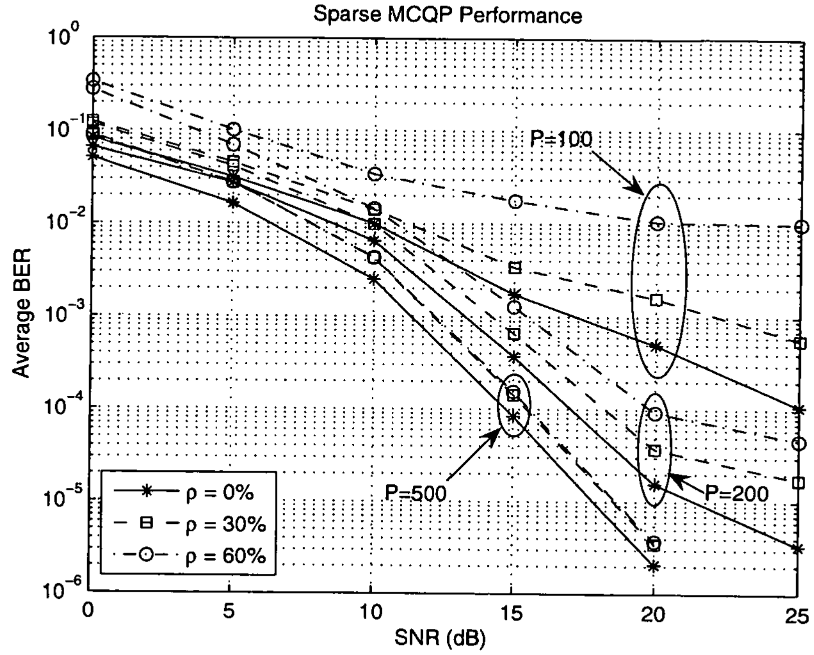


Figure 5.8: Sparse MCQP performance for different numbers of pilot symbols

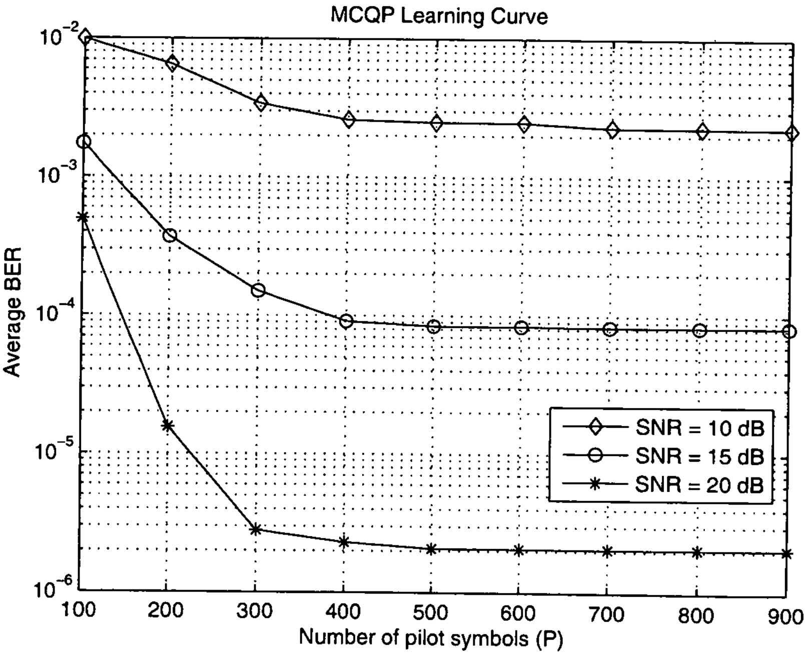


Figure 5.9: MCQP learning curve for CM3 at different SNR

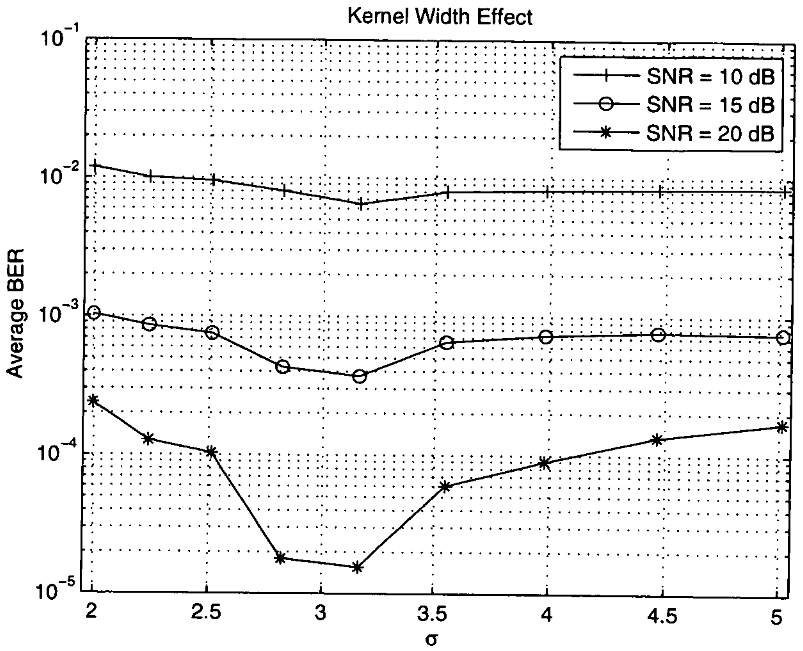


Figure 5.10: Choosing GRBF kernel parameter σ for MCQP ($P = 200$)

Figure 5.10 depicts the MCQP based receiver sensitivity to the kernel width parameter (σ) of the GRBF function for CM3 with different SNRs levels. The optimal values of σ is almost the same to that in SVC, *i.e.* $10^{0.5}$. The most significant observation in this context is the high sensitivity to the values outside these optimal values. In other words, the optimal MCQP based receiver performance can be obtained only for small range of kernel parameter.

5.5 Summary

The MCQP method of classification has been investigated and applied for various channel equalisation applications. In the first application, an MCQP based equaliser is proposed for nonlinear channel equalisation, which demonstrates a performance close to that of the optimal Bayesian detector. Compared to the SVM based equaliser, the proposed MCQP based approach has two advantages. First, it introduces the internal distance criteria to the objective function which improves the equalisation performance. Second, the optimisation in MCQP requires solving a linear set of equations, hence, a considerable reduction in the training computational complexity can be attained. Simulation results show that the proposed MCQP based equaliser outperforms, in terms of BER, the standard SVM based equaliser as well as reducing the training computational complexity significantly. Furthermore, the proposed equaliser has shown superb robustness to the time variation of the communication channels.

In the second application, the MCQP based receiver is proposed and discussed for DS-UWB channel equalisation. Results show superior performance of the proposed MCQP based receiver compared to the previous SVM based receivers. Therefore, for CM1, a close performance to the AWGN case is also obtained. A sparse version of MCQP based receiver is investigated to reduce the detection complexity and it is found that sparsity in MCQP based receiver is less effective than that in sparse LS-SVC. The learning convergence rate of the MCQP based receiver is smaller than the LS-SVC receiver. In terms of sensitivity to kernel parameter, the MCQP based receiver has shown high sensitivity to the optimal choice of σ.

Chapter 6

Relevance Vector Machine based Equalisation

The learning models of the previous proposed receivers in Chapters 4 and 5 can be regarded as "Deterministic" in terms of a single optimal solution is found for the equaliser. In this chapter, a sparse probabilistic learning model called relevance vector machine (RVM) is investigated and applied to the channel equalisation and signal detection of communication systems. The organisation of this chapter, consistently with previous chapters, is as follows. An introduction to the RVM methodology is provided in Section 6.1, and detailed descriptions of RVM learning model are presented in Section 6.2. An RVM based equalisation for nonlinear channels is proposed and presented in Section 6.3. Section 6.4 introduces and proposes the RVM based receiver for DS-UWB systems. Discussions and comparisons among all the proposed receivers in this PhD study are provided in Section 6.5. The summary of this chapter is presented in Section 6.6.

6.1 Introduction

The aim of supervised learning is to find a model that establishes the dependency of target values or labels with respect to input data. This model is constructed from a priorly given training set of data ($\{\mathbf{x}_i, y_i\}_{i=1}^N$, *where* N is the size of the training set, and y_i is the label for the input $\mathbf{x}_i$), hence the name "learning". The corresponding targets could be real values, for regression, or labels, for classification. The latter is considered in this study. As in the standard SVM methodology, the target estimation of an input point ($\mathbf{x}$) can be evaluated from the general functional expression

$$f(\mathbf{x}) = \sum_{i=1}^{N} w_i K(\mathbf{x}, \mathbf{x}_i) + w_0 \tag{6.1}$$

where $w_i's$ are model weights, w_0 represents the bias or the model shift from origin (*i.e.* β in the previous models). $K(.,.)$ is the typical kernel basis function. The SVM outputs a hard binary decision in classification. The RVM, however, is a probabilistic sparse kernel model identical in functional form to the SVM, introduced by [88]. The probabilistic estimates captures the uncertainty degrees to the resulting predictions.

The following sections presents the advantages of using RVM over standard SVM. The detailed description of the RVM model is also provided. Originally, RVM approach is aimed and developed for regression application, and by introducing integral approximations, it can be extended to perform classification tasks. The application of RVM based equalisation is investigated and discussed in the latter sections.

Advantages of RVM in comparison with SVM

The RVM methodology does not suffer from any of the following disadvantages that standard SVM do:

- Although relatively sparse, SVMs make unnecessarily liberal use of basis functions since the number of support vectors required typically grows linearly with the size of the training set.

- Predictions are not probabilistic. Ideally, it is desired to estimate the conditional distribution $p(y|\mathbf{x})$ in order to capture uncertainty in our prediction.

- In SVM, it is necessary to estimate the error/margin trade-off parameter 'C'. This generally entails a cross-validation procedure, which is wasteful both of data and computation.

- The kernel function $K(.,.)$ must satisfy Mercer's condition. That is, it must be the continuous symmetric kernel of a positive integral operator [87].

6.2 RVM Learning Model

Given a block of N training data $\{\mathbf{x}_i, y_i\}_{i=1}^N$, consider the estimation expression in Equation (6.1). The relevance vector (RV) approach for classification can readily be applied to construct the classifier [102]. Denote $\mathbf{y} = [y_1, ..., y_N]^T$ and $\mathbf{w} = [w_1...w_N]^T$. The posterior probability of $\mathbf{w}$ is

$$p(\mathbf{w}|\mathbf{y}, \boldsymbol{\alpha}) = \frac{P(\mathbf{y}|\mathbf{w}, \boldsymbol{\alpha})p(\mathbf{w}|\boldsymbol{\alpha})}{P(\mathbf{y}|\boldsymbol{\alpha})} \tag{6.2}$$

where $p(\mathbf{w}|\boldsymbol{\alpha})$ is model parameters prior with $\boldsymbol{\alpha} = [\alpha_1, ..., \alpha_N]^T$denoting the vector of hyperparameters. $P(\mathbf{y}|\mathbf{w}, \boldsymbol{\alpha})$ is the likelihood, and $P(\mathbf{y}|\boldsymbol{\alpha})$ is called the evidence. Considering the probabilistic binary encoding scheme to the targets labels, *i.e.*, $y \in \{0, 1\}$, the following likelihood of the Bayesian classification framework can be defined as

$$P(\mathbf{y}|\mathbf{w}, \boldsymbol{\alpha}) = \prod_{i=1}^{N} \{\sigma(f(\mathbf{x}_i))\}^{y_i}[1 - \sigma(f(\mathbf{x}_i))]^{1-y_i} \tag{6.3}$$

where $\sigma(.)$ is the logistic sigmoid function that is defined in Equation (3.16). The model parameters prior, $p(\mathbf{w}|\boldsymbol{\alpha})$, is chosen to be Gaussian with hyperparameters as expressed in the sparse Bayesian model in Equation (3.59), so that

$$p(\mathbf{w}|\boldsymbol{\alpha}) = \prod_{i=1}^{N} \left(\frac{\alpha_i}{2\pi}\right)^{1/2} \exp\left(-\frac{\alpha_i w_i^2}{2}\right). \tag{6.4}$$

As the marginal likelihood $P(\mathbf{y}|\boldsymbol{\alpha})$ cannot be obtained analytically by integrating out the weights from Equation (6.3), an iterative procedure is necessitated.

6.2.1 MAP-RVM Prediction

By initializing the hyperparameter vector $\boldsymbol{\alpha}$, a Gaussian approximation can be built to the posterior distribution in Equation (6.2) and thereby obtain an approximation to the marginal likelihood [87]. Maximisation of this approximate marginal likelihood then leads to a re-estimated value for $\boldsymbol{\alpha}$, and the process is repeated until convergence.

By considering the Laplace approximation for this model, with a fixed given $\boldsymbol{\alpha}$, the

MAP solution $\mathbf{w}_{MAP}$ can be obtained by maximising

$$\ln(p(\mathbf{w}|\mathbf{y},\boldsymbol{\alpha})) = \ln\{P(\mathbf{y}|\mathbf{w},\boldsymbol{\alpha})p(\mathbf{w}|\boldsymbol{\alpha})\} - \ln P(\mathbf{y}|\boldsymbol{\alpha}), \tag{6.5}$$

or, equivalently, by minimising the following cost function [102]:

$$J(\mathbf{w}|\mathbf{y},\boldsymbol{\alpha}) = \frac{1}{2}\mathbf{w}^T\mathbf{A}\mathbf{w} - \sum_{i=1}^{N}\left(y_i \log(\sigma_i) - (1-y_i)\log(1-\sigma_i)\right), \tag{6.6}$$

where $\mathbf{A} = diag\{\alpha_1, ..., \alpha_N\}$ and $\sigma_i = \sigma(f(\mathbf{x}_i))$. The iterative reweighted least squares (IRLS) [70] can be used to solve this optimisation problem in Equation (6.6). To do this, the gradient and the Hessian of the cost function J in Equation (6.6) with respect to $\mathbf{w}$ are evaluated, so that

$$\nabla J = \mathbf{A}\mathbf{w} - \mathbf{K}(\mathbf{y} - \boldsymbol{\sigma}), \tag{6.7}$$

$$\mathbf{H} = \nabla^2 J = \mathbf{K}^T\mathbf{B}\mathbf{K} + \mathbf{A}, \tag{6.8}$$

where $\boldsymbol{\sigma} = [\sigma(f(\mathbf{x}_1)), ..., \sigma(f(\mathbf{x}_N))]^T$, the matrix $\mathbf{K}$ is the kernel mapping matrix that is defined in Equation (5.8) , and $\mathbf{B} = diag\{\sigma_1(1-\sigma_1), ..., \sigma_N(1-\sigma_N)\}$.

At convergence of the IRLS algorithm, the Hessian represents the inverse covariance matrix for the Gaussian approximation to the posterior distribution around $\mathbf{w}_{MAP}$. The MAP-RVM solution ($\mathbf{w}_{MAP}$) can be then obtained by equating the gradient in Equation (6.7) to zero, so that the mean of the approximation is

$$\mathbf{w}_{MAP} = \mathbf{A}^{-1}\mathbf{K}(\mathbf{y} - \boldsymbol{\sigma}), \tag{6.9}$$

and the covariance matrix is

$$\mathbf{C} = \mathbf{H}^{-1}. \tag{6.10}$$

The hyperparameters $\boldsymbol{\alpha}$ are updated using [102]

$$\alpha_i^{new} = \frac{1 - \alpha_i^{old} c_{i,i}}{\mu_i^2} \tag{6.11}$$

with $c_{i,i}$ being the diagonal elements of $\mathbf{C}$. The new values of $\boldsymbol{\alpha}$ are iteratively applied to the cost function in Equation (6.6) until a convergence to the hyperparameters is achieved. During this process, many of the $\alpha's$ are driven to very large values and the corresponding weights to small near zero values. A zero threshold can then be set to remove these near-zero-value weights, leading to a sparse model.

6.2.2 MRVM Prediction

Having found the hyperparameters values $\boldsymbol{\alpha}$ that maximise the marginal likelihood, we can evaluate the marginalised RVM (MRVM) predictive distribution over $\hat{y}$ for a new input $\mathbf{x}$. Using marginalisation, according to [70], this is given by

$$P(\hat{y}|\mathbf{x}, \mathbf{y}, \boldsymbol{\alpha}) = \int P(\hat{y}|\mathbf{x}, \mathbf{w}) p(\mathbf{w}|\mathbf{y}, \boldsymbol{\alpha}) d\mathbf{w}, \tag{6.12}$$

where we assume the first quantity $P(\hat{y}|\mathbf{x}, \mathbf{w})$ represents the logistic sigmoid function, and the the second quantity $p(\mathbf{w}|\mathbf{y}, \boldsymbol{\alpha})$ is the approximated Gaussian posterior that is been found in the previous section of the MAP-RVM.

In the case of an MAP-RVM with the basis functions centred on data points, the model will therefore become increasingly certain of its predictions when extrapolating outside the domain of the data, which of course is undesirable. The predictive distribution in MRVM does not suffer from this problem. However, the computational cost of making predictions with an MRVM is typically much higher than with an MAP-RVM.

6.3 RVM based Equaliser for Nonlinear Channels

In this section, a number of simulations has been reported for the application of RVM based classifiers for nonlinear channel equalisation. In particular, the MAP-RVM model is used for convenience and simplicity of illustration. Some other learning machines based equalisers were implemented, namely the SVC and the MCQP [103], for comparison purposes.

6.3.1 System Model

The system model adopted in this experiment is identical to that in Subsection 5.3.1 for the MCQP based equalisation for nonlinear channel. Thus, the communication system block diagram is same to which is shown in Figure 5.1. And the same signalling settings and channel model are used for comparison purposes. Only time invariant scenario is considered in this work for convenience. The initial settings for the proposed RVM based equaliser's were $\mathbf{w} = \boldsymbol{\alpha} = 10^8$.

6.3.2 Simulation Results

In this section, the computer simulation configurations and settings are presented for the proposed system followed by a discussion to the obtained results. The nonlinear channel model of this experiment, according to those in Subsection 5.3.3, is given by the two transmit-receive relations of Equation (5.23) and Equation (5.22).

The signal constellation of output (received) signal is illustrated in Figure 6.1 for SNR=12 dB showing the nonlinear pattern to be passed through our classifiers. Three types of classifiers were used; RVM, SVC and MCQP based equaliser. The kernel width (σ) is fixed to 1 for all classifiers, and the SVC regularisation parameter (C) is empirically chosen to be 4. A, B and K of MCQP parameters were chosen as 0.1, 0.9, and 0.5 respectively.

The performance criterion used is the BER, and results are as shown in Figure 6.2. For low to moderate SNR levels, the RVM based receiver performs closely to the optimal Bayesian detector [103].

The most significant feature of RVM is its sparsity where the number of nonzero coefficients (relevant vectors or RVs) are the lowest amongst the other classifiers (*i.e.* equalisers) as shown in Figure 6.3. Moreover, the number of RVs in RVM does not considerably change with increased number of training symbols. This can be obviously seen in Figure 6.4 which can be interpreted as that the RVM based learning can predict the clusters of the underlying model, *i.e.*, in our constellation (Figure 6.1) we have 8 centred clusters of symbols and the number of resulting RVs is almost 8 whatever the training set size. Also, the number of resulting RVs is not significantly affected by the

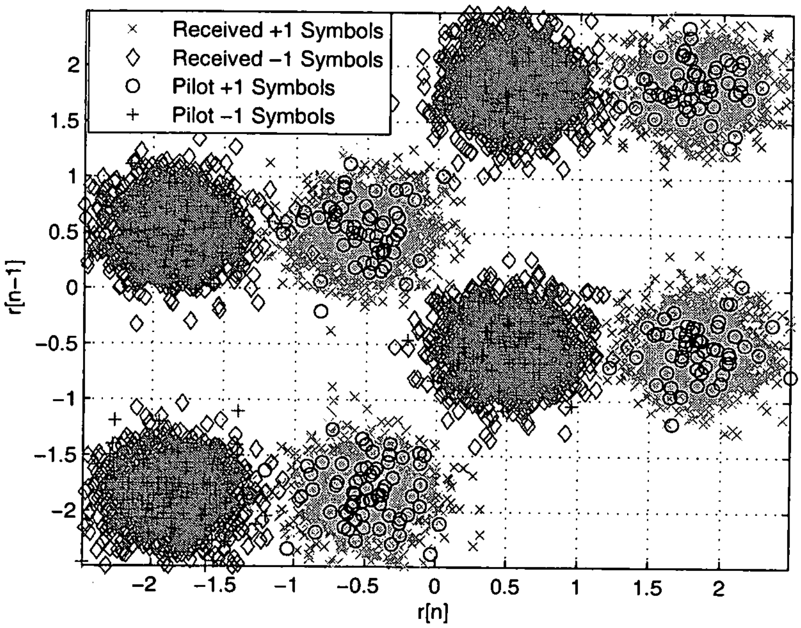


Figure 6.1: Received signal constellation for both pilot and detection, with SNR=12 dB.

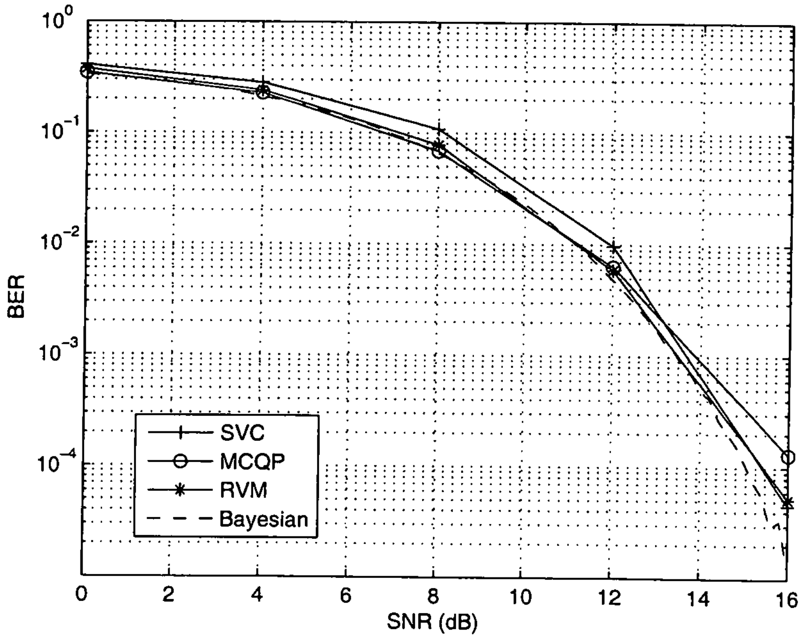


Figure 6.2: BER performance vs SNR

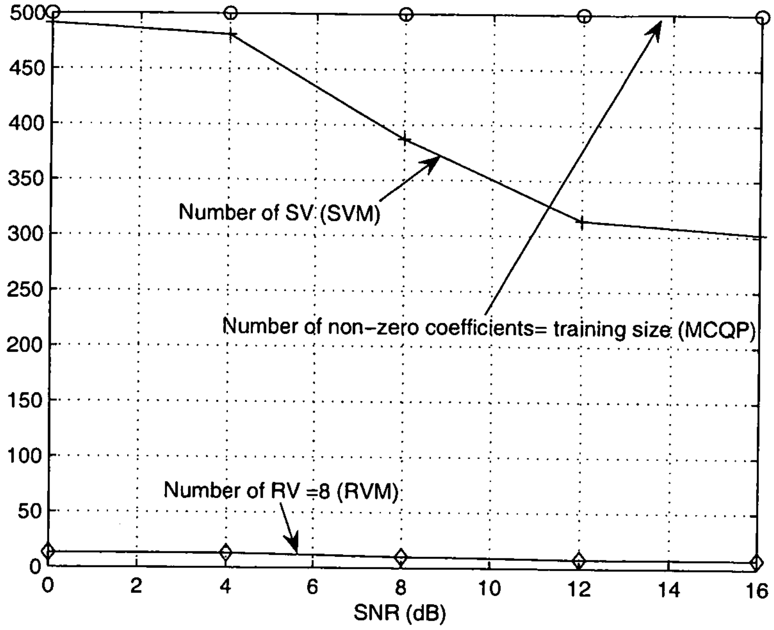


Figure 6.3: Number of nonzero coefficients vs SNR

kernel parameter (σ) as shown in Figure 6.5.

Another important feature worth to mention in RVM is that the BER performance of RVM classifier is not significantly changing with kernel parameter unlike other classifiers. Figure 6.6 illustrates this precisely.

6.4 RVM based Equalisers for DS-UWB Systems

In a consistent and similar way to the proposed receivers in the previous chapters. This section proposes and investigates the probabilistic RVM based equalisation for DS-UWB systems. The RVM based classification modules are used in the the ML units of the proposed DS-UWB receiver in Figure 4.5. The MAP-RVM based equaliser were used these experiments, and an MRVM based equaliser is investigated as well. The following subsections describe these equalisers in details, and present and discuss the simulation results compared to the conventional and the previously proposed receivers in DS-UWB systems.

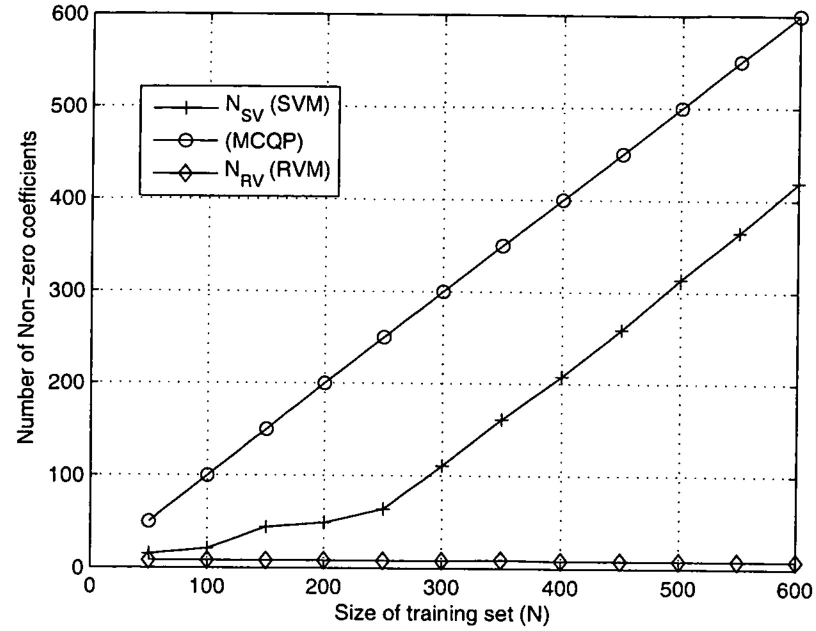


Figure 6.4: The sparsity change with number of training symbols

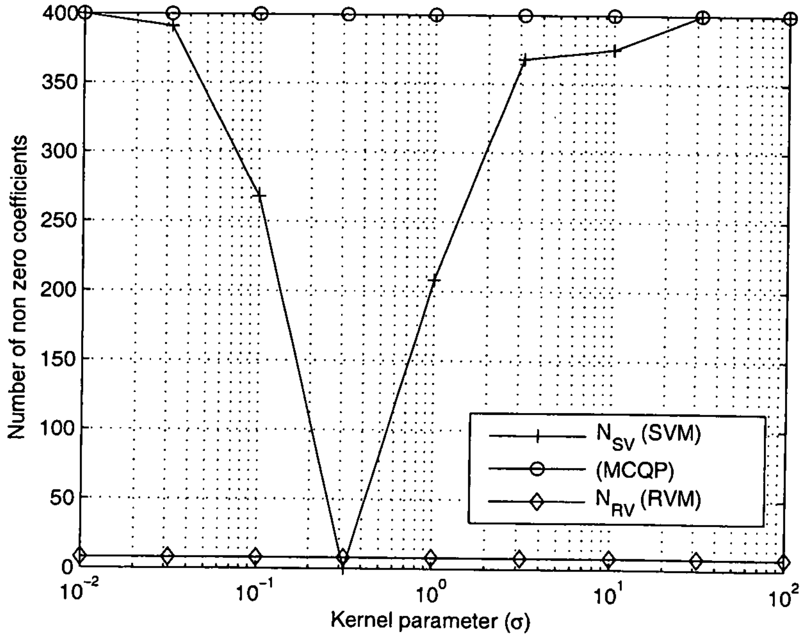


Figure 6.5: The effect of kernel parameter (σ) on number of nonzero coefficients

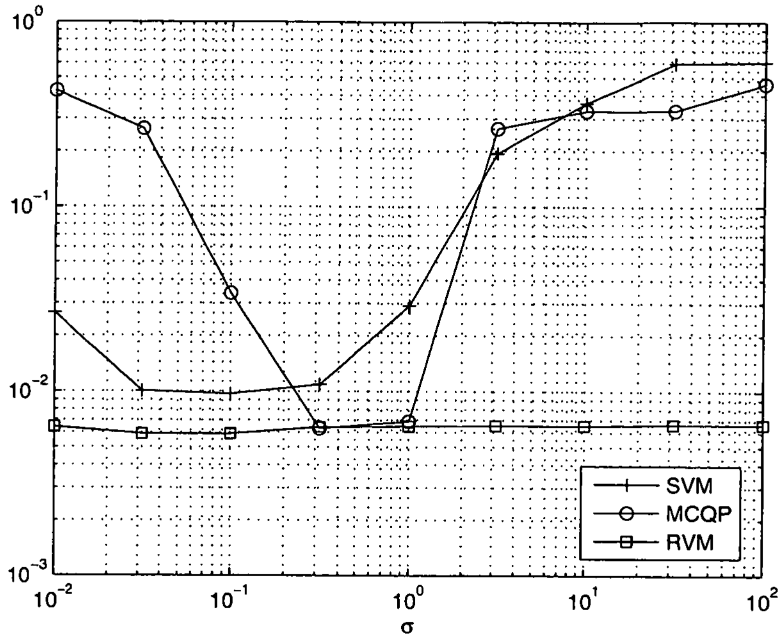


Figure 6.6: The sensitivity to kernel width (σ) for nonlinear channel (SNR=12 dB)

6.4.1 MAP-RVM based Equaliser

As described in Section 4.3, a group of M parallel MAP-RVM based equalisers are used at the receiver. In the training mode, and for each equaliser, the input to the m^{th} equaliser can be defined as $\mathbf{x}_i = \mathbf{x}_i^{(m)}$, as in Equation (4.6). Now, we construct $\mathbf{R} = [\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_P]^T$, $\mathbf{K}(\mathbf{R}, \mathbf{R}^T)$ as in Equation (5.8) and $\mathbf{y} = [y_1, y_2, ..., y_P]^T$. y_i represents the pilot symbols of this m^{th} equaliser, and they are converted to $\{0, 1\}$ coding for probabilistic convenience. The learning aim in the MAP-RVM based equalisers is to find the optimum coefficients ($\mathbf{w}_{MAP}$) according to the iterative procedure discussed in Subsection 6.2.1. So that, and after initialisation, the cost function in Equation (6.6) is evaluated, then the Hessian and the covariance matrices are calculated from Equation (6.8) and Equation (6.10) respectively. The $\mathbf{w}_{MAP}$ solution is evaluated from Equation (6.9), and the hyperparameters are then updated according to the Equation (6.11). This process is repeated until convergence. The resulting optimum $\mathbf{w}_{MAP}$ solution is used for signal estimation in Equation (6.1). The symbols are then detected by applying a threshold of 0.5 to the signal estimates as following

$$\hat{b}_m = \begin{cases} +1 & f(\mathbf{x}) \geq 0.5 \\ -1 & f(\mathbf{x}) < 0.5 \end{cases} \tag{6.13}$$

6.4.2 MRVM based Equaliser

The optimisation in the MRVM based equaliser is the same as in the MAP-RVM case, the difference lays in the detecting process. The resulting Gaussian approximations of the converged parameters from MAP-RVM, *i.e.,* Equation (6.9) and Equation (6.10), are used in detection (*i.e.,* prediction). This is by integrating out equalisers coefficients according to the form in Equation (6.12). In fact, this integration is not tractable analytically, therefore a simple numerical method is employed to approximate the integration. The basic Simpson's rule [104] for multivariables is used for simplicity, considering five points around the $\mathbf{w}_{MAP}$ solution. Symbol detection is then performed by thresholding the integration approximation to the detector in Equation (6.13).

6.4.3 Complexity Analysis

Computational complexity is the main drawback of the RVM methods. However, some pruning techniques [70] are usually applied to reduces P to a manageable size in most problems. In short, the pruning procedure truncates the resulting zeros coefficients and their corresponding data points from the matrices after each iteration.

For RVM based equalisers training complexity, the update rules for the hyperparameters depend on computing the posterior weight covariance matrix, which requires an inverse operation of order $O(P^3)$ complexity [87]. This is carried out in an iterative manner, with a number of iterations equal to N_{IT}. The overall RVM based equaliser training complexity is then of order of $O(N_{\text{IT}}P^3)$, which, of course, leads to extended training times, although the disadvantage of this is significantly offset by the lack of necessity to perform cross-validation over nuisance parameters, such as the nuisance parameter C in the SVM.

For detection complexity, the numbers of multiplications for one detection session (B data symbols) of the MAP-RVM are of order $O(\bar{N}_{\text{RV}}N_cB)$, where $\bar{N}_{\text{RV}}$ is the average

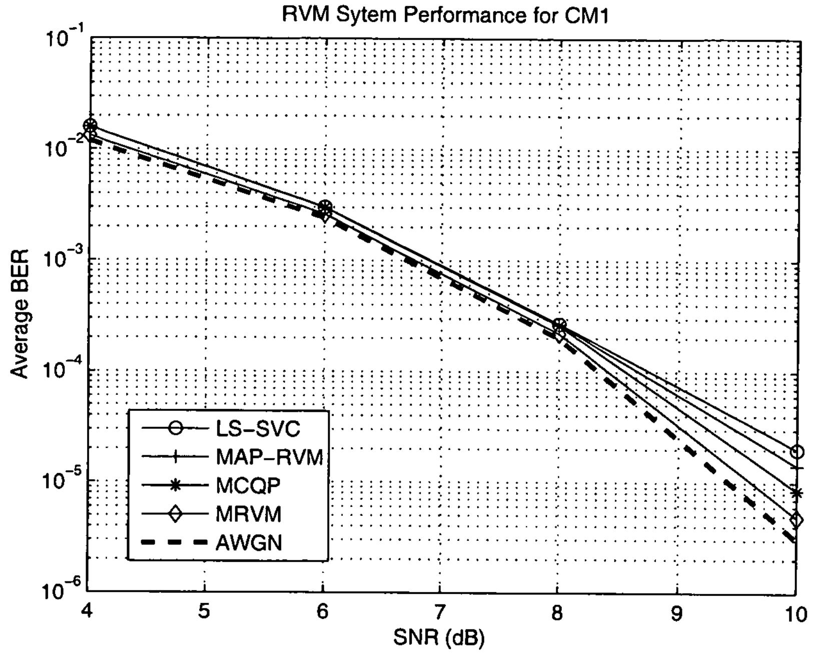


Figure 6.7: RVM system performance for CM1; Number of pilot symbols $P = 100$

number of the resulting relevance vectors of the MAP-RVM based equalisers. For the MRVM based equalisers, applying numerical approximations to perform the integration in Equation (6.12) results in an prohibitive detection complexity. So, for a resulting RV of size $\bar{N}_{\mathrm{RV}}$ and five points of integration periods, the detection complexity can be of order $O(\bar{N}_{\mathrm{RV}} 5^P N_c B)$. However, it is shown that the number of RVs in the MAP-RVM based equalisers is much less than the number of SVs in the SVM based equalisers, and therefore the complexity reduction of MAP-RVM over SVM is up to 70%.

6.4.4 Simulation Results

This subsection presents simulation settings and results to demonstrate the performance of the proposed RVM based receivers for DS-UWB systems. For the sake of consistency in the comparison, the simulation settings and assumptions of Subsection 4.4.6 are considered. They are also described in Subsection 5.4.4. The initial values of the equalisers' coefficients are set to zero, and the RVM hyperparameters are set initially to a large value of 10^8. For the MRVM based receiver, five points of equal spaces of 0.1 are centred around the optimum $\mathbf{w}_{MAP}$ solution.

For the LOS scenario of CM1 settings, Figure 6.7 depicts the BER performance of

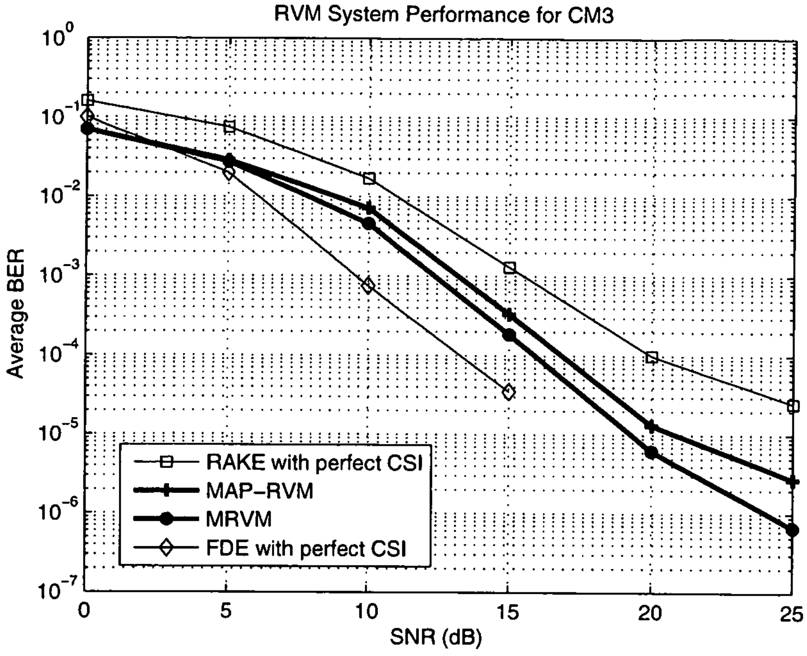


Figure 6.8: RVM system performance for CM3; Number of RAKE fingers L_f = Number of pilot symbols $P = 200$

the RVM based receivers in comparison with the LS-SVC and MCQP based receivers. For the same training complexity settings, *i.e.,* $P = 100$, the proposed RVM based receiver has also shown similar superior performance to the LS-SVC and the MCQP based receivers. In particular, the MRVM based receiver outperforms all other receivers in higher SNR levels (*e.g.,* at $10\,dB$).

For the NLOS scenario of CM3, the BER performance of the RVM based receivers is illustrated in Figure 6.8 where the number of RAKE fingers and the number of pilot symbols of the classifiers are set to be $L_f = P = 200$. By a comparison to the conventional RAKE receiver and the FDE with perfect CSI, it is obvious the significant performance of the RVM based receivers over the conventional receivers. The MAP-RVM and MRVM based receivers outperform the RAKE receiver with the perfect CSI under similar complexity settings. As expected, the MRVM receivers outperforms the MAP-RVM since a range of equalisers coefficients were employed for integration.

Figure 6.9 illustrates the effects of increasing the pilot size on the performance of both the MAP-RVM and the MRVM based receivers. This was examined for three pilot sizes $P = 100, 200$ and 500 for CM3. And as expected for most probabilistic models,

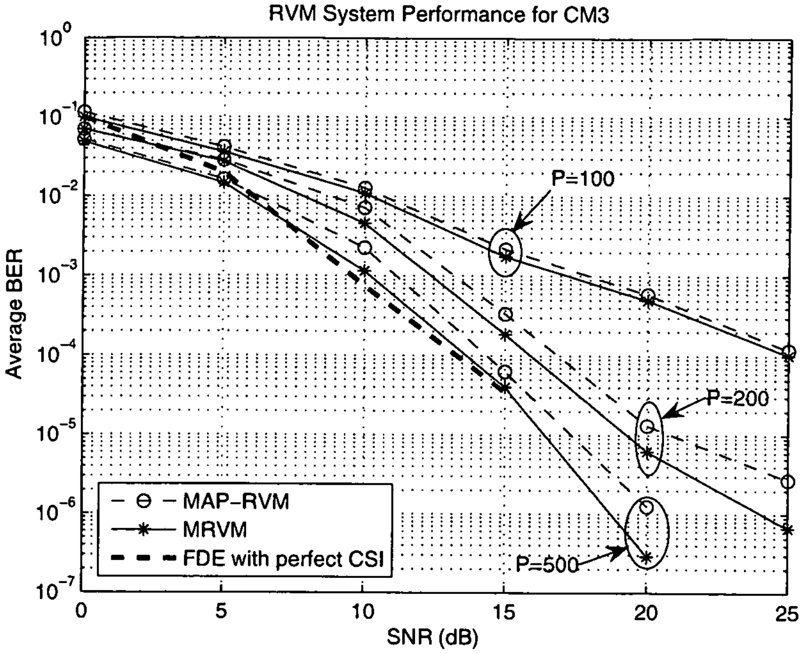


Figure 6.9: RVM performance for different numbers of pilot symbols

increasing the number of observed data significantly improves the model performance. This can be confirmed by the close performance to the ideal FDE case at $P = 500$. Also, the performance improvement due to the increase in pilot size is more obvious in the MRVM based receiver case.

The learning curves of the MRVM based receiver is illustrated in Figure 6.10. Again, three SNR levels were considered in these tests for CM3. Results demonstrates that although a significantly better BER performance can be achieved by the MRVM based receiver, in comparison with the previously proposed receivers, it requires a large training pilot size to do so. A different observation in the RVM learning curve, as compared-with the previously proposed receivers, is that the optimal pilot size increases as the SNR level increases.

An important feature in RVM methodology is its sparsity, that is, only a few training pilot data are used in detection. This can be confirmed in Figure 6.11 where a comparison between standard SVC and RVM based receivers at $P = 500$ is illustrated. The numbers of nonzero coefficients represent the average numbers of the SVs and RVs for the SVC and RVM, respectively. The results show the reduced number of the resulting RVs, in comparison with the number of SVs, which therefore means a reduced detection

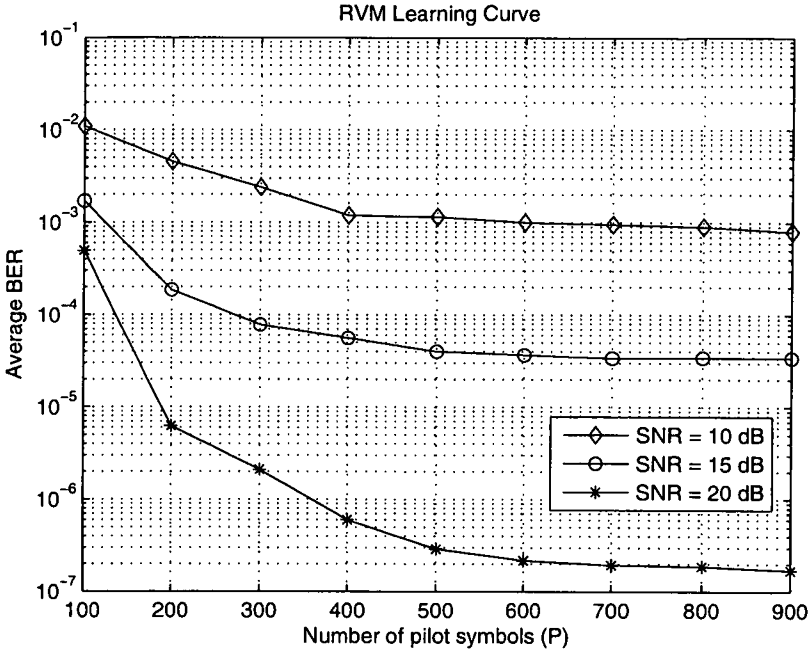


Figure 6.10: MRVM learning curve for CM3 at different SNR levels.

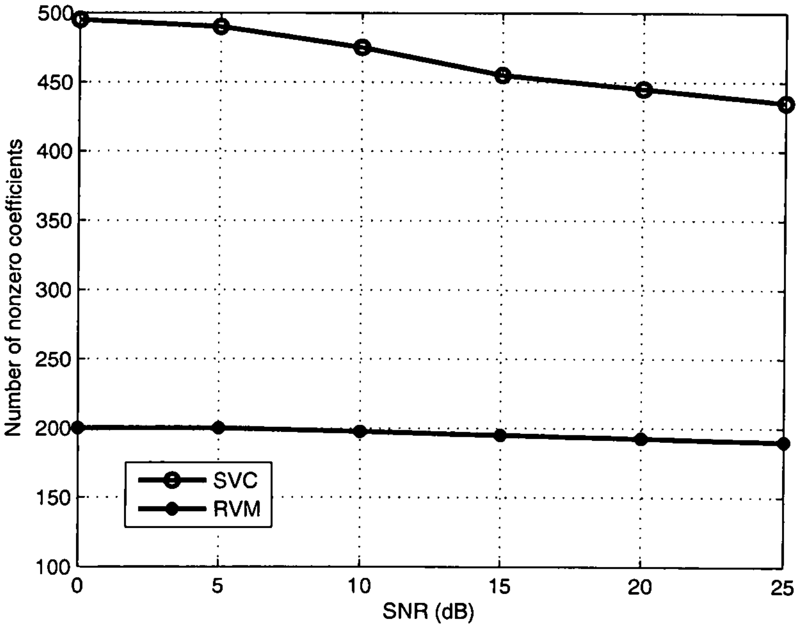


Figure 6.11: Number of nonzero coefficients vs SNR for CM3 in DS-UWB system

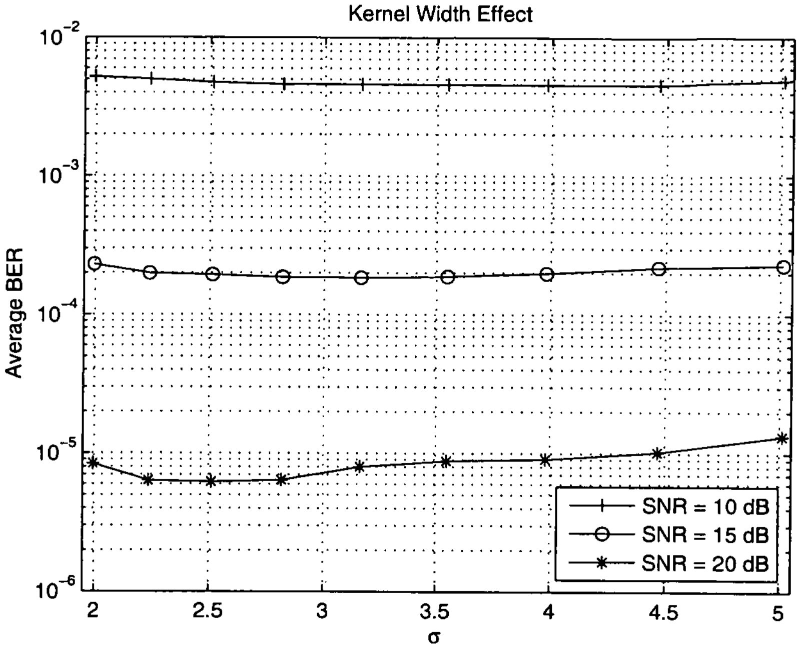


Figure 6.12: Choosing GRBF kernel parameter σ for RVM ($P = 200$)

complexity for the RVM based receiver. Also, the robustness of the RVM, in terms of the number of RVs, to the SNR level is an advantage of this receiver.

Figure 6.12 depicts the RVM based receiver sensitivity to the kernel width parameter (σ) of the GRBF function for CM3 with different SNRs levels. The results show that the BER performance of the RVM based receiver is insensitive to the kernel width parameter value used, which is an advantage of this receiver.

6.5 Comparisons and Discussions

This section discusses the proposed receivers, and provides comparisons among them in terms of their performance, complexity, sparsity, learning convergence and sensitivity to kernel parameter. Selected results were rearranged and illustrated to show significances in the related aspect of comparison.

6.5.1 Performance Comparisons

In terms of BER performance, Figure 6.13 shows the BER results of all the proposed receivers for CM3 with a training pilot of size 200. The RAKE receiver and FDE with

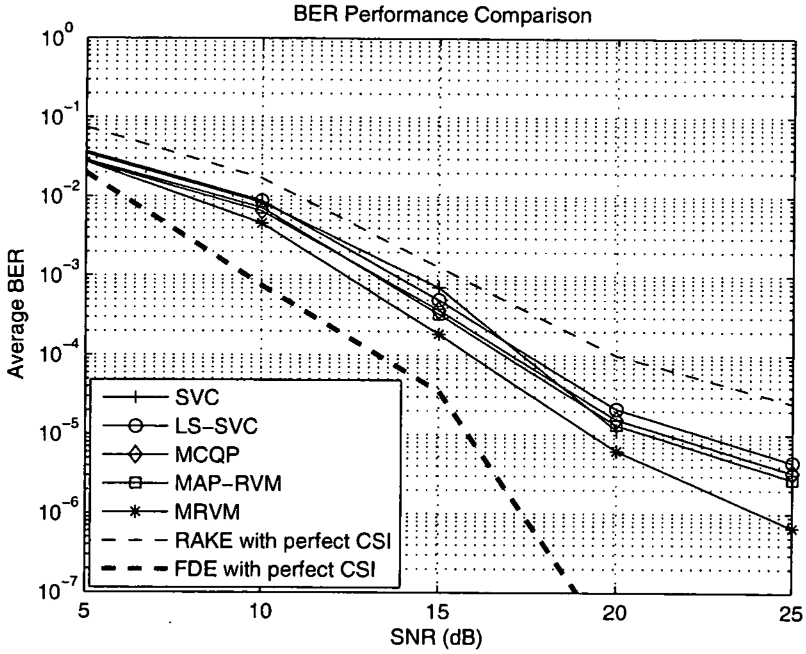


Figure 6.13: Performance comparisons of the machine learning based systems for CM3; Number of RAKE fingers L_f = Number of pilot symbols $P = 200$

perfect knowledge of the same channel are also shown as reference. Results confirm the significance of the proposed receivers that is even better than the ideal RAKE if same complexity is assumed. Among the proposed receivers, they show close performance at this pilot size with notable performance improvement for the MRVM receiver. Also, the MCQP and MAP-RVM based receivers are almost identical and they outperform the LS-SVC and SVC based receivers slightly. The SVC based receiver, however, approach the MCQP and MAP-RVM at higher SNR levels.

The sparsity of an equaliser is referred to the fact that only a subset of the training data are used in detection. Therefore, it is highly demanded in designing the receiver. SVC and RVM techniques provide this feature by their nature. In practice, the proposed RVM based equalisers have shown better model sparseness than the SVC based counterparts. This can be confirmed in the results of Figure 6.11. In the LS-SVC and MCQP based receivers, the sparseness can be imposed by pruning as discussed in the previous chapters. Figure 6.14 depicts the BER performance of the sparse LS-SVC vs sparse MCQP based receivers for CM3 with a pilot size of 100. Results inform that the sparse LS-SVC based receiver outperforms the sparse MCQP based receiver, although

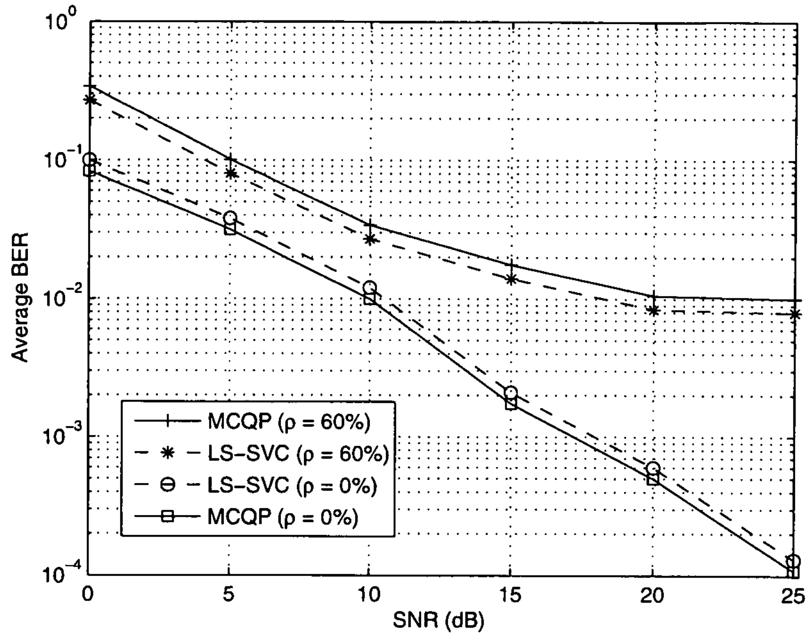


Figure 6.14: Sparsity comparison of LS-SVC and MCQP based systems for CM3; Number of pilot symbols $P = 100$

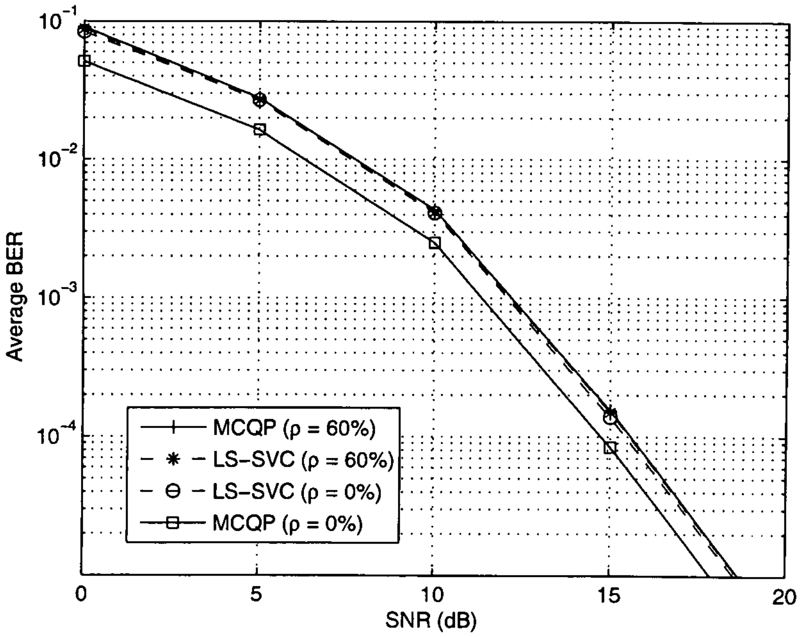


Figure 6.15: Sparsity comparison of LS-SVC and MCQP based systems for CM3; Number of pilot symbols $P = 500$

Table 6.1: Performance comparison between sparse MCQP vs sparse LS-SVC

Pilot size	Sparseness ratio		
	$\rho = 0\%$ (No Sparseness)	$\rho = 30\%$	$\rho = 60\%$
$P = 100$	20%	-13%	-26%
$P = 200$	35%	-9%	-16%
$P = 500$	63%	-2%	-3%

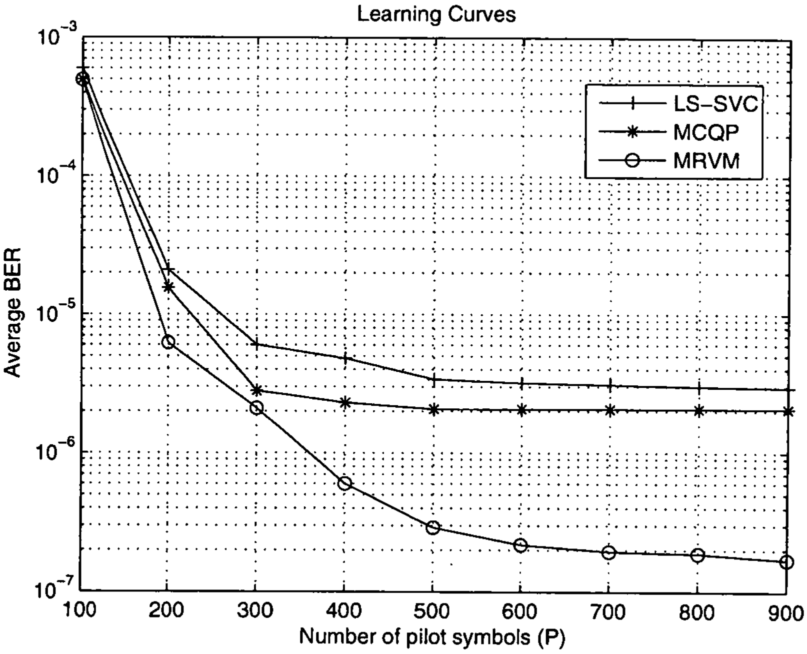


Figure 6.16: A comparison of different machine learning based equalisers learning curves for CM3 at SNR level of $20dB$.

the non-sparse receiver are vice versa. This can be shown obviously in a much larger pilot size as in Figure 6.15, where for $P = 500$, both the sparse and non-sparse LS-SVC based receivers perform almost identically. On the other hand, the MCQP based receiver retains the performance advancement over its sparse version. Table 6.1 provides the MCQP performance improvement as a percentage over the sparse LS-SVC (negative values mean better sparse LS-SVC performance). In other words, the percentage of BER performance improvement is evaluated from

$$\frac{BER_{\text{sparse LS-SVC}} - BER_{\text{sparse MCQP}}}{BER_{\text{sparse LS-SVC}}} \times 100. \tag{6.14}$$

Figure 6.16 illustrates the learning curves of the proposed receivers in the training for CM3 at SNR level of $20dB$. The trends show that the MCQP based receiver has the

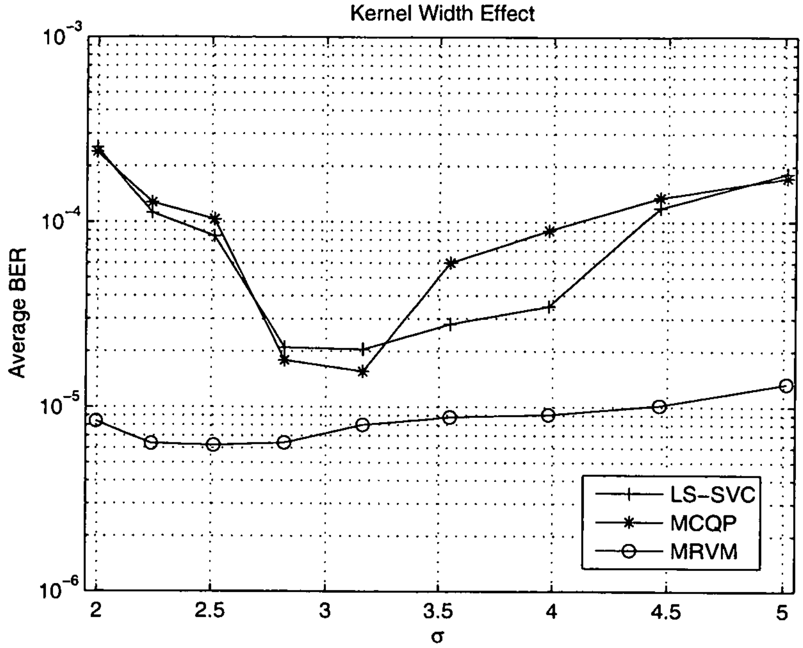


Figure 6.17: A comparison of GRBF kernel width sensitivity for CM3 ($P = 200, SNR = 20\,dB$)

fastest convergence rate, such that it can achieve its steady state just after 300 symbols at this SNR level. Whereas RVM based receiver is the slowest to converge to its steady state, yet it provides the best performance.

The sensitivity to the kernel width parameter (σ) is illustrated in Figure 6.17. This is by testing the BER performance of the proposed receivers for CM3 at the pilot size of 200 and the SNR level of $20\,dB$. The tests were carried out for a range of σ values, from $10^{0.3}$ to $10^{0.7}$, that contains the optimal σ. These results show the highest sensitivity to σ for MCQP based receiver where the optimum BER performance can be achieved within a small range around the optimum σ. On the other hand, the RVM based receiver proves its advantage since the BER trend is almost flat within the specified range of σ.

6.5.2 Complexity Comparisons

Table 6.2: Computational complexity comparisons for all proposed equalisers

	Training Complexity	Detection Complexity
SVC	$O(P^3)$	$O(\bar{N}_{SV} N_c B)$
LS-SVC	$O((N_c + 1)P^2)$	$O(PN_cB)$
Sparse LS-SVC		$O((1-\rho)PN_cB)$
MCQP	$O((N_c + \frac{3}{2})P^2)$	$O(PN_cB)$
Sparse MCQP		$O((1-\rho)PN_cB)$
MAP-RVM	$O(N_{IT}P^3)$	$O(\bar{N}_{RV} N_c B)$
MRVM		$O(\bar{N}_{RV} 5^P N_c B)$

Table 6.2 summarises the computational complexity to the proposed equalisers in terms of the number of multiplications required, where P is the pilot size, N_c is spreading code length, B is the number of data symbols for detection, $\bar{N}_{SV}$ and $\bar{N}_{RV}$ are the average numbers of resulting support and relevance vectors, respectively, and N_{IT} represents the number of iterations in RVM training. As regarding the training requirements, the LS-SVC based equaliser has shown the lowest complexity, while the MCQP based equaliser has the second lowest complexity with slight increase in the training requirements, compared with the LS-SVC. The RVM based equaliser, however, has the highest complexity which is approximately N_{IT} times more complex than the SVC based equaliser.

For detection, the computational complexity is dependent on either a predesigned factor such as the sparseness ratio (ρ) in the sparse LS-SVC and MCQP cases, or on a resulting number of non-zero coefficients or weights (Lagrange multipliers). In practice, $\bar{N}_{RV}$ is the lowest among the others, in terms of resulting non-zero vectors, and can be of 70% reduction to the pilot size, while $\bar{N}_{SV}$, on the other hand, is the highest, and no more than 13% of reduction was obtained. Among all, the detection complexity of the MRVM based equaliser is the highest due to the numerical integration part in it.

6.6 Summary

In this chapter, the probabilistic method of RVM has been investigated and applied for various channel equalisation applications. In the first application, an RVM based equaliser is proposed for nonlinear channel equalisation, which demonstrates a performance close to that of the optimal Bayesian detector. Compared to the SVM and MCQP based equalisers, the proposed RVM based approach has three advantages. Firstly, It does not required any hyperparameter to be set via cross-validation method. Second, it provides an impressive few number of model parameter which mean lots of reduction in detection time (the sparsity). Third, The kernel parameter almost does not affect the performance of RVM which provides robustness and less sensitivity to kernel choice.

The RVM based receivers are also proposed and discussed for DS-UWB channel equalisation. Two variants of RVM receivers, according to their prediction strategy, were considered. The MAP-RVM and the MRVM. The simulation settings were fixed to those in the previous chapters. Results show similar, and even superior, performance of the proposed RVM based receiver compared to the previous proposed receivers. In particular, the MRVM based receiver outperforms all other receivers. For large pilot size, the RVM based receivers provide a performance close to the ideal FDE case. Sparsity is an impressive feature of applying the RVM and a reduction of up to 62% of pilot symbols can be obtained, which outperforms the SVC based receiver. The learning convergence rate of the RVM based receivers are slower than the MCQP and LS-SVC receiver. In terms of sensitivity to the kernel parameter, the RVM based receivers has shown the least sensitivity to the optimal choice of σ, with slight dependence to the SNR level.

The chapter has been concluded by providing discussions and comparisons among all the proposed receivers, in terms of performance, sparsity, learning convergence, sensitivity to kernel parameter, and computational complexity.

Chapter 7

Conclusions and Further Research

7.1 Conclusions

The thesis has applied machine learning algorithms for channel equalisation of single-user wideband wireless communication systems, in particular the DS-UWB, which aim to combat the severe frequency selective channels thus the high ISI, and to improve the throughput of the systems. This is by mapping the estimation and equalisation problem into a pattern recognition solution in detection process. The significant performance of the machine learning algorithms, through kernel-induced function, is also incorporated in tackling unknown channels by observing distorted samples from known pilot data.

Chapter 4 has proposed a bank of SVM based equalisers in the receiver structure for DS-UWB systems. The proposed block-wise SVC based receiver achieves a significant BER performance over the conventional RAKE receiver, even for the case that a perfect knowledge of channel is available. For LOS UWB channels, the SVC based receiver provides the performance which is close to that of pure AWGN. Although detection complexity in SVC is low due to the fewer support vectors, the optimisation processing in SVC is accomplished via QP procedure, which increases the training complexity of the receiver. Hence, a lower training complexity LS-SVC based equalisers have been proposed to replace the SVC based ones. The LS-SVC based receiver provides almost same performance of the SVC. The detection complexity in LS-SVC, however, is raised. The latter issue can be alleviated by imposing sparseness to the LS-SVC coefficients up to specific performance tolerance index. The performance and complexity analyses of the proposed receivers have been investigated. Also, the sensitivity to the kernel function parameter was investigated and it is found that kernel parameter in SVM based receivers are moderately sensitive to specific values, and not much sensitive to the SNR level.

Chapter 5 has investigated a lower complexity machine learning algorithm that uses multi-criteria in optimisation processing modules. One of the contributions of this chapter is to propose the MCQP based equaliser for nonlinear channel systems, for both time-invariant and time-variant scenarios. While achieving a significant complexity reduction, the MCQP based receiver provides improved BER performance, with nearly the same performance as the typical optimal Bayesian detector in time-invariant scenario. By utilising a similar receiver structure to that in SVM based system, MCQP based equalisers are also proposed for DS-UWB system at the receiver end. The MCQP based receiver outperforms its LS-SVC counterpart in terms of BER performance. A sparse version of MCQP based receiver is also proposed to reduce the detection complexity, but the performance is negatively affected comparing to the sparse LS-SVC counterpart. By increasing the number of pilot symbols for training, MCQP based receiver achieves a steady-state BER with a higher convergence speed. A drawback to the proposed MCQP is the high sensitivity to the choice of kernel parameter.

Chapter 6 has investigated the probabilistic learning algorithms for classification and their sparse Bayesian inferred models represented by the RVM technique. An RVM based equaliser is, therefore, proposed for the nonlinear channel system obtaining better BER performance than the previously proposed equalisers. RVM based equalisers are arranged in a similar way to construct the RVM based receiver for DS-UWB system. Two variants of RVM based equalisers, according to their predicting criteria, were proposed: the MAP-RVM using a single set of optimised coefficients, and the MRVM using integration over the coefficients distribution. As expected, the MRVM is found to significantly outperform its MAP-RVM counterpart. In fact it outperforms all the proposed receivers in this study. Furthermore, the proposed RVM based receivers show efficiency to the kernel parameter choice, *i.e.*, the performance does not change much for different parameter values. Another impressive advantage of using RVM is its high sparsity and robustness in the detection process, hence, a very low detecting complexity can be achieved. The cost is the significant high training complexity. Also, the RVM based receivers have a slow convergence to the steady-state when increasing the number of pilot symbols.

Among all the proposed receivers for DS-UWB, it can be concluded that, in terms of BER performance, the MRVM based receiver outperforms all the others in cost of very high training computational complexity. In terms of training complexity, LS-SVC based receiver show the lowest training complexity but near to MCQP. The sparsity in detection is accredited to the RVM based receivers with a stable number of relevance vectors. However, imposed sparsity can be obtained for both LS-SVC and MCQP, and the former shows better performance than the latter. RVM based receivers are the least sensitive to the choice of kernel parameter. In terms of training convergence, *i.e.*, learning curves, MCQP based receivers are the fastest to converge whereas the RVM based receivers are the slowest.

7.2 Further Research

The field of machine learning algorithms is really a rich and unlimited horizon of motivations and potential contributions, considering their promising performance and elegant developments. For this reason, they can be extensively investigated and developed to be applied as solutions to many of the current signal processing challenges in communication systems. To continue the research having been done, the following ideas can be suggested for future research activity, which will still focus on developing machine learning algorithms for communication systems.

- The channel estimation and equalisation using machine learning processing modules can help improve the performance of the wireless communication systems in time domain through observed data samples. The same idea can be applied to frequency domain signal processing, which is to mitigate the effect of additive noise and phase noise in low to medium SNR levels in the existing frequency domain signal processors.

- The BER performance of the proposed systems can be improved further by systematically optimising the kernel mapping function rather than the existing empirical and validation tests. The similar kernel mapping scheme can be applied to different machine learning algorithms. The comparison will reveal how differently the same kernel mapping scheme works on these algorithms.

- Despite the fact that some of the proposed receivers reduce the training or/and detecting complexity, there is still a vital requirement of developing lower complexity learning algorithms. The potential algorithms should consider linear training with minimum number of pilot symbols as well as to have huge sparsity for detection.

- The proposed systems consider single user single input single output (SISO) scenario. This idea can be elaborated to include MUD or/and multi-input multi-output (MIMO) systems. By such developments, the observation input domain may be perceived in space diversity or in any diversity scheme.

Bibliography

[1] M. Z. Win and R. Scholtz, "Impulse radio: How it works," *IEEE Communications Letters*, vol. 2, no. 2, pp. 36–38, Feb. 1998.

[2] J. H. Reed, Ed., *An Introduction to Ultra Wideband Communication Systems.* Pearson Education, Inc., 2005.

[3] H. Arslan, Z. N. Chen, and M. D. Benedetto, Eds., *Ultra Wideband Wireless Communication.* Wiley Blackwell, 2006.

[4] S. Wood and D. R. Aiello, *Essentials of UWB*, ser. Cambridge wireless essentials series. Cambridge University Press, 2008.

[5] D. Porcino and W. Hirt, "Ultra-wideband radio technology: Potential and challenges ahead," *IEEE Communications Magazine*, pp. 66–74, Jul. 2003.

[6] R. Aiello and A. Batra, *Ultra Wideband Systems Technologies and Applications*, ser. Communications Engineering Series. Newnes, 2006.

[7] A. Molisch, J. Foerster, and M. Pendergrass, "Channel models for ultrawideband personal area networks," *IEEE Wireless Communications*, vol. 10, no. 6, pp. 14–21, Dec. 2003.

[8] R. S. Michalski, I. Bratko, and M. Kubat, Eds., *Machine Learning and Data Mining Methods and Applications.* John Wiley & Sons, ltd., 1998.

[9] S. Haykin, *Neural Networks and Learning Machine.* Pearson Education, Inc., 2009.

[10] V. Vapnik, *The nature of statistical learning theory.* Springer, 2001.

[11] V. N. Vapnik, *Statistical Learning Theory.* NewYork Wiley, 1998.

[12] S. Chen, S. Gunn, and C. Harris, "Decision feedback equalizer design using support vector machines," in *IEE Proc. Vision, Image and Signal Processing*, 2000.

[13] D. J. Sebald and J. A. Buklew, "Support vector machine technique for nonlinear equalization," *IEEE Transactions on Signal Processing*, vol. 48, no. 11, pp. 3217–3226, Jun. 2000.

[14] S. Chen, A. K. Samingan, and L. Hanzo, "Support vector machine multiuser receiver for DS-CDMA signals in multipath channels," *IEEE Nueral Netwroks*, vol. 12, pp. 604–611, May 2001.

[15] D. P. Agrawal and Q. Zeng, *Introduction to Wireless and Mobile Systems.* Thomson, Brooks/Cole, 2003.

[16] X. Gu and L. Taylor, "Ultra-wideband and its capabilities," *BT Technology Journal*, vol. 21, no. 3, pp. 56–66, July 2003.

[17] S. Roy, J. Foerster, V. Somayazulu, and D. Leeper, "Ultrawideband radio design: The promise of high-speed, short-range wireless connectivity," *PROCEEDINGS OF THE IEEE*, vol. 92, no. 2, pp. 295–311, February 2004.

[18] M. Ghavami, L. B. Michael, and R. Kohno, *Ultra Wideband signals and systems in communication engineering.* John Wiley & Sons, Ltd,, 2005.

[19] L. Yang and G. Giannakis, "Ultra-wideband communications," *IEEE Signal Processing Magazine*, pp. 26–54, Nov. 2004.

[20] T. Barrett, "History of ultra wideband communications and radar: Part 1, uwb communications," *Microwave Journal*, pp. 22–56, January 2001.

[21] R. J. Fontana, "A brief history of uwb communications," website. [Online]. Available: http://www.multispectral.com/

[22] R. Scholtz, "Multiple access with time-hopping impulse modulation," *IEEE MILCOM*, vol. 2, pp. 447–450, October 1993.

[23] [Online]. Available: http://www.timedomain.com/

[24] [Online]. Available: http://www.uwbforum.org/

[25] [Online]. Available: http://www.wimedia.org/

[26] A. Mehbodniya and S. Aissa, "Effects of mb-ofdm system interference on the performance of ds-uwb," *Vehicular Technology, IEEE Transactions on*, vol. 58, no. 8, pp. 4665–4669, Oct. 2009.

[27] Y. Li, X. Xia, R. Yao, and W. Zhu, "Coding assisted iterative channel estimation for impulse radio ultra-wide band communication systems," ser. ICASSP, vol. 3, 2005, pp. 330–332.

[28] M. G. D. Benedetto and G. Giancola, *Understanding Ultra Wide Band Radio Fundamentals*, ser. Prentice Hall Communications Engineering and Emerging Technologies. Prentice Hall, 2004.

[29] S. Erkucuk, D. I. Kim, and K. S. Kwak, "Code shift keying impulse modulation for uwb communications," *Wireless Communications, IEEE Transactions on*, vol. 7, no. 9, pp. 3285–3291, September 2008.

[30] R. C. Qiu, H. Liu, and X. Shen, "Ultra-wideband for multiple access communications," *IEEE Communications Magazine*, pp. 80–87, February 2005.

[31] D. Wu, P. Spasojevic, and I. Seskar, "Ternary zero correlation zone sequences for multiple code uwb," *Proc. of 38th Conference on Information Sciences and Systems, Princeton, NJ,*, pp. 939–943, March 2004.

[32] M. Laughlin, M. Welborn, and R. Kohno, "Summary presentation of the xtreme spectrum proposal," *IEEE P802.15-03/334r5 WPAN*, 2003.

[33] V. Somayazulu, "Multiple access performance in uwb systems using time hopping vs. direct sequence spreading," in *Proc. 2002 IEEE Conf. Wireless Communications and Networking,*, vol. 2, March 2002, pp. 522–525.

[34] T. S. Rappaport, *Wireless Communications principles and practice.* Prentice-Hall, Upper saddle River, NJ 07458, 2002.

[35] J. Foerster, "Channel modeling subcommittee report final," IEEE802.15-02/490, Tech. Rep., October 2003.

[36] A. Saleh and R. Valenzuela, "A statistical model for multipath propagation," *IEEE journal on selected areas in communications*, vol. SAC-5, no. 2, pp. 128–137, February 1987.

[37] M.-G. DiBenedetto, T. Kaiser, A. F.Molisch, I. Oppermann, C. Politano, and D. Porcino, Eds., *UWB communication systems: A Comprehensive Overview.* EURASIP Book Serieson Signal Processing and Communications,, 2006.

[38] X. Chu and R. D. Murch, "Performance analysis of DS-MA impulse radio communications incorporating channel-induced pulse overlap," *IEEE Transactions on Wireless Communications*, vol. 5, no. 4, pp. 948–959, Apr. 2006.

[39] Z. Wang and B. Giannakis, "Wireless multicarrier communications," *IEEE Signal Processing Magazine*, vol. 17, no. 3, pp. 29–48, May 2000.

[40] M. Welborn, "System considerations for ultra-wideband wireless networks," in *Proc. IEEE 2001 Radio and Wireless Conf.*, 2001, pp. 5–8.

[41] M. Win and R. Scholtz, "Characterization of ultra-wide bandwidth wireless indoor channels: a communication theoretic view," *IEEE J. Selected Areas in Communications*, vol. 20, no. 9, pp. 1613–1627, December 2002.

[42] V. Lottici, A. D. Andrea, and U. Mengali, "Channel estimation for ultra-wideband communications," *IEEE Journal on Selected Areas in Communications*, vol. 20, no. 9, pp. 1638–1645, Dec. 2002.

[43] B. Mielczarek, M. Wessman, and A. Svensson, "Performance of coherent uwb rake receivers with channel estimators." in *IEEE VTC Fall*, Oct. 2003, pp. 1880–1884.

[44] H. Sato and T. Ohtsuki, "Frequency domain channel estimation and equalisation for direct sequence-ultra wideband (DS-UWB) system," *IEE Proceeding-Communication*, vol. 153, no. 1, pp. 93–98, 2006.

[45] A. D'Amico, U. Mengali, and M. Morelli, "Multipath channel estimation for the uplink of a ds-cdma system," in *Proc. ICC 2002,*, vol. 1, 2002, pp. 16–20.

[46] A. Tonello and R. Rinaldo, "A frequency domain approach to channel estimation, detection, and interference cancellation for impulse radio systems," *ICASSP*, vol. 3, pp. 613–616, 2005.

[47] Y. Wang and X. Dong, "Frequency-domain channel estimation for SC-FDE in UWB communications," *IEEE Trans. Communications*, vol. 54, no. 12, pp. 2155–2163, Dec. 2006.

[48] B. Sklar, *Digital Communications: Fundamentals and Applications*, 2nd ed. Prentice Hall, 2001.

[49] J. Proakis, *Digital communications*, 4th ed. McGroaw Hill, New York, NY, 2000.

[50] A. Rajeswaran, V. Somayazulu, and J. Foerster, "RAKE performance for a pulse based UWB system in a realistic UWB indoor channel," in *Proc. ICC 2003,*, vol. 4, May 2003, pp. 2879–2883.

[51] Y. Li, A. F. Molisch, and J. Zhangr, "Channel estimation and signal detection for uwb," in *Proc. IEEE WPMC*, 2003.

[52] S.-L. Chiou and M.-X. Chang, "Analysis of DS-UWB with MMSE equalization in dispersive channels," in *Proc. 2006 Int. Conf. Wireless Communications, Networking and Mobile Computing,*, Sept. 2006, pp. 1–4.

[53] A. Parihar, L. Lampe, R. Schober, and C. Leung, "Equalization for DS-UWB systems-part I: BPSK modulation," *IEEE Transactions on Communications*, vol. 55, no. 6, pp. 1164–1173, Jun. 2007.

[54] L. Zhiwei, A. Premkumar, and A. Madhukumar, "Matching pursuit-based tap selection technique for uwb channel equalization," *Communications Letters, IEEE*, vol. 9, no. 9, pp. 835–837, Sep 2005.

[55] K. B. Toh and S. Tachikawa, "On equalization for direct sequence-ultra wideband system using received response code sequence," *IEICE Trans. Fundam. Electron. Commun. Comput. Sci.*, vol. E91-A, no. 9, pp. 2637–2645, 2008.

[56] D. Morgan, "Wideband equalization using multiple antennas," in *Proc. 2008 Conf. Wireless Communications and Networking,*, March 31- April 3 2008, pp. 634–639.

[57] E. Torabi, J. Mietzner, and R. Schober, "Pre-equalization for pre-rake MISO DS-UWB systems," in *Proc. ICC 2008,*, May 2008, pp. 4861–4866.

[58] Y. Ishiyama and T. Ohtsuki, "Performance evaluation of uwb-ir and ds-uwb with mmse-frequency domain equalization (fde)," in *Global Telecommunications Conference, 2004. GLOBECOM '04. IEEE*, vol. 5, Nov.-3 Dec. 2004, pp. 3093–3097 Vol.5.

[59] P. Kaligineedi and V. K. Bhargava, "Frequency-domain turbo equalization and multiuser detection for DS-UWB systems," *IEEE Transactions on Wireless Communications*, vol. 7, no. 9, pp. 3280–3284, September 2008.

[60] A. Rajeswaran, V. C. Somayzullu, and J. R. Foester, "Rake performance for a pulse based UWB system in a realistic UWB indoor channel," in *IEEE ICC2003*, May 2003, pp. 2879–2883.

[61] D.Cassioli, M. Win, F. Vatalaro, and A. Molisch, "Performance of low-complexity RAKE reception in a realistic UWB channel," in *IEEE ICC2002*, vol. 2, no. 28, May 2002, pp. 763 - 767.

[62] H. Sato and T. Ohtsuki, "Computational complexity and performance of rake receivers with channel estimation for ds-uwb," *IEICE Trans. Fundamentals*, vol. 88-A, no. 9, pp. 2318–2326, Sep. 2005.

[63] D. Falconer, S. Ariyavisitakul, A. Benyamin-Seeyar, and B. Eidson, "Frequency domain equalization for single-carrier broadband wireless systems," *IEEE Communication Magazine*, pp. 58–66, april 2002.

[64] H. Sari, G. Karam, and I. Jeanclaude, "Frequency-domain equalization of mobile radio and terrestrial broadcast channels," *IEEE Global Telecommunications Conference*, vol. 1, pp. 1–5, 1994.

[65] J. Suykens, T. V. Gestel, J. D. Brabanter, B. D. Moor, and J. Vandewalle, *Least Squares Support Vector Machines.* World Scientific, 2002.

[66] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, 2nd ed. John Wiley & Sons, Inc., 2001.

[67] S. Chen, "Adaptive equalization of finite non-linear channels using multilayer perceptrons," *Signal Processing*, vol. 20, no. 2, pp. 107–119, 1990. [Online]. Available: http://ci.nii.ac.jp/naid/10012703690/en/

[68] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory.* printice hall, 1993.

[69] C. M. Bishop, *Neural Networks for Pattern Recognition.* Clarendon Press, 1995.

[70] ——, *Pattern Recognition and Machine Learning.* Springer, 2007.

[71] N. Christiani and Shawe-Taylor, *An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods.* Cambridge University Press, 2000.

[72] P. Domingos and M. J. Pazzani, "On the optimality of the simple bayesian classifier under zero-one loss," *Machine Learning*, vol. 29, no. 2-3, pp. 103–130, 1997. [Online]. Available: citeseer.ist.psu.edu/domingos97optimality.html

[73] J. Neter, M. Kutner, C. Nachtsheim, and W. Wasserman, *Applied Linear Statistical Models*, 4th ed. Irwin, Chicago, IL, 1990.

[74] [Online]. Available: http://www.dtreg.com/svm.htm

[75] T. M. Cover, "Geometrical and statistical properties of systems of linear inequalities with applications in pattern recognition," *IEEE Trans. Electronic Computers,*, vol. EC-14, no. 3, pp. 326–334, June 1965.

[76] C. J. C. Burges, "A tutorial on support vector machines for pattern recognition," *Data Mining and Knowledge Discovery*, vol. 2, no. 2, pp. 121–167, 1998.

[77] J. Shawe-Taylor and N. Cristianini, *Kernel Methods for Pattern Analysis.* Cambridge University Press, 2004.

[78] A. Aizerman, E. M. Braverman, and L. I. Rozoner, "Theoretical foundations of the potential function method in pattern recognition learning," *Automation and Remote Control*, vol. 25, pp. 821–837, 1964.

[79] J. Mercer, "Functions of positive and negative type and their connection with the theory of integral equations," *Philos. Trans. Roy. Soc. London*, 1909.

[80] V. Vapnik, "An overview of statistical learning theory," *Neural Networks, IEEE Transactions on*, vol. 10, no. 5, pp. 988–999, Sep 1999.

[81] B. Boser, I. Guyon, and V. Vapnik, "A training algorithm for optimal margin classifiers," in *Proc. 5th Annual Workshop Computational Learning Theory.* (New York, USA): ACM, 1992, pp. 144–152.

[82] B. Schlkopf and A. Smola, *Learning with Kernels.* The MIT press, 2002.

[83] R. Fletcher, *Practical methods of optimization (2nd Edition).* New York, USA: Wiley-Interscience, 1987.

[84] O. Chapelle, V. Vapnik, O. Bousquet, and S. Mukherjee, "Choosing multiple parameters for support vector machines," *Machine Learning*, vol. 46, pp. 131–159, 2002.

[85] K.-R. Müller, S. Mika, G. Rätsch, K. Tsuda, and B. Schölkopf, "An introduction to kernel-based learning algorithms," *IEEE Transactions on Neural Networks*, vol. 12, no. 2, pp. 181–201, March 2001.

[86] P. Craven and G. Wahba, "Smoothing noisy data with spline functions: estimating the correct degree of smoothing by the method of generalized cross-validation," *Numerische Mathematik*, vol. 31, no. 4, pp. 377–403, December 1978.

[87] M. E. Tipping, "Sparse bayesian learning and the relevance vector machine," *Journal of Machine Learning Research*, vol. 1, pp. 211–244, June 2001. [Online]. Available: http://www.jmlr.org/papers/volumel/tipping01a/tipping01a.pdf

[88] M. Tipping, "The relevance vector machine," *Advances in Neural Information Processing Systems*, 2000.

[89] D. J. MacKay, "The evidence framework applied to classification networks," *Neural Computation*, vol. 4, pp. 720–736, 1992.

[90] H. Chen, P. Tino, and X. Yao, "Probabilistic classification vector machines," *IEEE Trans. Neural Networks,*, vol. 20, no. 6, pp. 901–914, June 2009.

[91] L. Wang, Ed., *Support Vector Machines : Theory and Applications.* Berlin : Springer, 2005.

[92] K. W. Lau and Q. H. Wu, "Online training of support vector classifier," *Pattern Recognition*, vol. 36, no. 8, pp. 1913–1920, Aug. 2003.

[93] J. A. Suykens and J. Vandewalle, "Least squares support vector machine classifiers," *Neural Processing Letters*, vol. 9, pp. 293–300, 1999.

[94] Y. Zhao, L. Wang, J. Frigon, C. Nerguizian, K. Wu, and R. Bosisio, "UWB positioning using six-port technology and a learning machine," in *Proc. 2006 Conf. Mediterranean Electrotechnical,*, May 2006, pp. 352–355.

[95] K. W. Lau and Q. H. Wu, "Leave one support vector out cross validation for fast estimation of generalization errors," *Pattern Recognition*, vol. 37, no. 9, pp. 1835–1840, 2004.

[96] J. A. K. Suykens, L. Lukas, and J. Vandewalle, "Sparse least squares support vector machine classifiers," in *European Symposium on Artificial Neural Networks (ESANN 2000)*, Apr. 2000, pp. 37–42.

[97] M. M. Laughlin, M. Welborn, and R. Kohno, "Summary presentation of xtreme spectrum proposal," IEEE P802.15 WPANS, Tech. Rep., 2003.

[98] Y. Peng, G. Kou, Y. Shi, and Z. Chen, "A multi-criteria convex quadratic programming model for credit data analysis," *Decis. Support Syst.*, vol. 44, no. 4, pp. 1016–1030, 2008.

[99] S. Chen, B. Mulgrew, and P. Grant, "A clustering technique for digital communications channel equalization using radial basis function networks," *IEEE Transactions on Neural Networks*, vol. 4, no. 4, pp. 570–590, Jul 1993.

[100] Q. Liang and J. M. Mendel, "Equalization of nonlinear time-varying channels using type-2 fuzzy adaptive filters," *IEEE Transactions on Fuzzy Systems*, vol. 8, no. 5, pp. 551–563, October 2000.

[101] C. F. N. Cowan and S. Semnani, "Time-variant equalization using a novel nonlinear adaptive structure," *International Journal of Adaptive Control and Signal Processing*, vol. 12, no. 2, pp. 195–206, 1998.

[102] S. Chen, S. Gunn, and C. Harris, "The relevance vector machine technique for channel equalization application," *IEEE Trans. Neural Networks*, vol. 12, no. 6, pp. 1529–1532, Nov 2001.

[103] M. Musbah and X. Zhu, "Multi-criteria quadratic programming based low complexity nonlinear channel equalisation," in *European Signal Processing Conference, EUSIPCO 2009*, August 2009.

[104] R. L. Burden and J. D. Faires, *Numerical Analysis*, 8th ed. Thomson, Brooks/Cole, 2005.