Distribution Dependent Adaptive Learning

Vimal Bhatia

A thesis submitted for the degree of Doctor of Philosophy.
The University of Edinburgh.
August 2005

Abstract

The aim of this thesis is to develop algorithms and techniques for adaptive signal processing in non-Gaussian noise environment with applications to communications channel estimation and equalisation.

Practical communication systems are affected by thermal noise, caused by the thermal (Brownian) motion of particles in components. Thermal noise is readily modelled as a stationary independent Gaussian stochastic process popularly known as additive white Gaussian noise (AWGN). The adaptive signal processing for communication systems has been predominated by this AWGN assumption. Channel estimation and equalisation forms an integral part of communications receivers which are usually designed using the Gaussian noise assumption. In some communication channels, the observation noise exhibits impulsive, as well as Gaussian characteristics. On the other hand, recent increase in the use of wireless devices results in many such wireless devices operating in the vicinity of another wireless device. This has caused an increase in interference from the other wireless devices operating in the same band, which effectively causes the noise in the presence of interference to deviate from Gaussianity. Thus in practice the Gaussian noise assumption does not hold for practical communication systems and scenarios. To improve the performance of adaptive algorithms, we develop algorithms adapted on the noise characteristics rather than adapting only on second order statistics. The developments in this thesis can be classified in two major works.

First work is on developing a minimum bit-error rate (MBER) decision feedback equaliser (DFE) for impulsive noise modelled as an α-stable distribution. The development exploits the stable nature of the α-distribution and the concepts build on earlier work in a Gaussian noise environment. Further, a Wiener-filter-with-limiter solution is also presented and used as a performance bench mark. An improvement in convergence and BER performance is achieved by using a minimum bit error rate (MBER) cost function instead of a conventional least mean square (LMS) based design. The ability of least BER (LBER) equalisers based on a Gaussian noise assumption to operate in an α-stable noise environment is also highlighted.

In the second work, a block based maximum-likelihood algorithm using kernel density estimates to improve channel estimation in non-Gaussian noise environment is proposed. The

likelihood pdf is assumed unknown and is estimated by using a kernel density estimator at the receiver. Thereby combining log-likelihood as a cost function with a kernel density estimator provides a robust channel estimator, which could be used for various non-Gaussian noise environments without any modification. The performance of the proposed estimator is compared with the theoretical lower bounds for associated noise distribution. The simulations for impulsive noise and co-channel interference (CCI) in the presence of Gaussian noise, confirms that a better estimate can be obtained by using the proposed technique as compared to the traditional algorithms. The proposed algorithm is then applied to orthogonal frequency division multiplexing (OFDM) communication systems. A considerable performance improvement is observed when using a non-parametric channel estimator in conjunction with a symbol-by-symbol non-parametric maximum *a posteriori* probability (MAP) equaliser. Since, in practice, CCI is correlated in nature, a whitening filter based approach for channel estimation is proposed. In order to make the channel estimation technique robust to channel order mismatch, a novel technique to simultaneously adapt the channel order and channel coefficient is discussed.

to my parents

मेरे माता पिता

Acknowledgements

This thesis culminates the efforts of many people who have assisted me in tangible and intangible ways during my stay in, the beautiful, Edinburgh.

First and foremost I would thank Prof. Bernard (Bernie) Mulgrew, my supervisor, for his guidance, advice, encouragement and sharing his insight with me. I also like to thank him for showing confidence in me, supporting my ideas and taking time out for me from his busy-busy schedule. Under his guidance I have learnt not only research skills but also a new attitude towards life.

I wish to thank Prof. C. F. N. Cowan and Dr. Y. Gong (Alex) from Queen's University Belfast for fruitful meetings and sharing their experiences. I would also like to thank Alex for sharing his experience in my carrer development. I also wish to thank Prof. D. D. Falconer for inviting me to his research group over the summer 2004 and sharing his vast experience in equalisation and OFDM systems. I would like to thank the faculty members S. McLaughlin, J. S. Thompson, D. I. Laurenson, P. M. Grant, N. Goertz, J. Hopgood and the post-docs Y. Kopsinis, D. Blanco, A. K. Dinnis, N. H. Nedev for invaluable discussions and support. The wonderful support from the lecturers and collegues from school of mathematics also helped in my thesis work.

My studies at Edinburgh would not have been possible without the generous financial support from the UK-EPSRC and the Institue for Digital Communications. Also fellowship from IEE helped me complete one major part of my thesis by visiting Carleton University, Canada.

Many thanks to all the collegues and friends for making my stay in Edinburgh wonderful and enjoyable and of course making me realise that there is (sometimes) life outside Room 2.12 and King's Building. Can't forget Amit for being my best man, Khamish and Somu for endless laughter and fun over the years. The excellent support facility from administration, finance, library and computing services has gone a long way to help ease the research pressure.

Lastly but not the least, support from my family has helped me get so far. Laughter from my two wee nephews' naugtiness, support from Anu (brother) and Shalu (his wife) can't be forgetten. I can't find words (in any language) to express my gratitude for my parents for their endless sacrifies and continuous encouragement.

Contents

List of figures

List of tables

Acronyms and abbreviations

3G	3rd generation
AIC	Akaike's information criterion
ASK	amplitude shift keying
AWGN	additive white Gaussian noise
BER	bit-error rate
CCI	co-channel interference
CDMA	code division multiple access
CPU	central processor unit
CRB	Cramér Rao bound
DAB	digital audio broadcasting
DFE	decision feedback equaliser
DFT	discrete Fourier transform
DS-CDMA	direct sequence code division multiple access
DSL	digital subscriber loop
DSP	digital signal processor
DVB	digital video broadcasting
EM	expectation maximization
FFT	fast Fourier transform
FIM	Fisher information matrix
FIR	finite impulse response
FPE	final prediction error
FSK	frequency shift keying
FT	Fourier transform
GHz	giga-Hertz
GSM	global system for mobile
i.i.d.	independent identically distributed
ICI	inter-carrier interference
IEEE	institute of electrical and electronics engineers

IFFT	inverse fast Fourier transform
ISI	inter-symbol interference
LBER	least bit-error rate
LMS	least mean square
LPE	linear prediction error
LS	least-squares
mph	miles per hour
MAP	maximum *a posteriori* probability
Mbps	mega bits per second
MBER	minimum bit-error rate
MDL	minimum description length
MEE	minimum error entropy
MHz	mega-Hertz
MIMO	multiple input multiple output
ML	maximum-likelihood
MLSE	maximum-likelihood sequence estimation
MMSE	minimum mean square error
MSE	mean square error
NMSE	normalised mean squares error
NPML	non-parametric maximum-likelihood
OFDM	orthogonal frequency division multiplexing
pdf	probability density function
P/S	parallel to serial
PAM	pulse amplitude modulation
PAPR	Peak to average power ratio
PSK	phase shift keying
QAM	quadrature amplitude modulation
QPSK	quadrature phase shift keying
RLS	recursive least squares
RV	random variable
SαS	symmetric α stable

S/P	serial to parallel
SAGE	space-alternating generalised expectation maximisation
SISO	single input single output
SIR	signal to interference ratio
SNR	aignal to noise ratio
VLSI	very large scale intergation
WiMAX	Worldwide Interoperability for Microwave Access
WLAN	wireless local area network
WSL	Wiener solution wth limiter
ZF	zero forcing

Nomenclature

$a(.)$	feed-forward equaliser taps
$b(.)$	feed-back equaliser taps
$\mathbb{C}$	complex number
C	level for PAM signalling
$d(.)$	coefficients of LPE filter
$e(.)$	error signal samples
$\epsilon(.)$	whitened error signal sample
$\mathbb{E}.$	expectation operator
$f(.)$	probability density function
$F_\alpha(.)$	alpha stable characteristic function
$g_p(.)$	pth interferring channel
$h(.)$	impulse response of a channel
$\mathbf{h}$	channel vector
$h_C(.)$	impulse response of a physical channel
$h_R(.)$	impulse response of a receiver matched filter
$h_T(.)$	impulse response of a transmitter modulation filter
$h_{TR}(.)$	combined impulse response of transmitter and receiver filter
$H(.)$	Fourier transform of $h(.)$
$\mathbf{H}$	channel matrix
$H_C(.)$	Fourier transform of $h_C(.)$
$H_R(.)$	Fourier transform of $h_R(.)$
$H_T(.)$	Fourier transform of $h_T(.)$
$H_{TR}(.)$	Fourier transform of $h_{TR}(.)$
$H_E(.)$	entropy
$I(.)$	interference in frequency domain
k	time index
κ_1, κ_2	constants
$K(.)$	Kernel function
log	natural logarithm

M	block length
$n(.)$	additive white Gaussian noise
N	noise free channel states
N_M	number of mixtures
N_T	channel length
$N(.)$	additive white Gaussian noise in frequency domain
$\mathcal{N}$	normal distribution
$\mathbf{p}$	time averaged cross-correlation vector
$P(.)$	probability distribution function
$P_E(.)$	probability of error
P_I	number of interferers
$Q(.)$	generalised error function
$\mathbf{R}$	time averaged auto-correlation vector
T	time period of transmitted signal
$u_p(.)$	interferring user input
$x(.)$	input sequence in time domain
$\mathcal{X}$	random variable
$X(.)$	input sequence in frequency domain
$y(.)$	received sequence in time domain
$Y(.)$	received sequence in frequency domain
$\bar{y}(.)$	noiseless received sequence
$\hat{y}(.)$	estimated received sequence
$z(.)$	output of DFE
$(.)^*$	conjugate operator
$\theta(.)$	phase response of the channel
f_c	upper cut-off frequency
β	excess bandwidth factor
μ	learning rate/step-size
$(.)^T$	transpose operator
$\Re\{.\}$	Real part of complex number
$\Im\{.\}$	Imaginary part of complex number
$\|\|.\|\|$	Euclidean norm
∇	gradient vector

Chapter 1
Introduction

The field of adaptive signal processing has seen several advances in recent years primarily due to explosive growth in digital communications. The demand for increasing bit throughput for multimedia and data services is driving advances in communication technology. Various new modulation, diversity, coding techniques have been developed in recent years to match the demands of today's bit hungry applications. The operating frequencies for communication systems have also been going up from MHz to GHz. Thus bandwidth efficiency is a major concern in the communication research community today.

In real life, communication systems are affected by channel noise and interference from different users. To improve the performance of communication systems in noisy and/or interference limited varying channel conditions the use of adaptive signal processing techniques is highly desirable. This thesis deals with developing such adaptive algorithms and techniques which perform better than those based on Gaussian stationary noise assumptions used in communication systems.

The chapter begins with an exposition of the principal motivation behind the work undertaken in this thesis. Section 1.2 outlines the contributions made in this thesis. Lastly, the thesis layout is described in section 1.3.

1.1 Motivation for work

The revolution in digital communication techniques can be attributed to the invention of the automatic linear adaptive equaliser in the late 1960's [1]. From the modest start, adaptive equalisers have gone through many stages of development and refinement in the last 40 years. Early equalisers were based on linear adaptive filter algorithms [2] with or without a decision feedback. Alternatively, the maximum likelihood sequence estimator (MLSE) [3] were implemented using the Viterbi [4][5] algorithm. The MLSE requires knowledge of the channel, which was readily available from using a least squares estimation. These methods have been applied for several years primarily for two main reasons.

Firstly, both forms of equalisers provide two extremes in terms of performance achieved. The linear equaliser has low complexity but has poor performance in severe conditions. An infinitely length MLSE has better performance in severe channel conditions, however the computational complexity is quite high. Until recent years Gaussian noise was considered to be a major impediment to the communication receiver's performance, where the above two equalisers perform fairly well, depending on the channel conditions.

Secondly, rapid advances in digital signal processing (DSP) techniques have provided scope for very large scale integration (VLSI) implementation. The DSP chips specialise in signal processing functions like multiply and add much faster than other central processing units (CPUs). The power of DSPs has been increasing and their cost has come down rapidly, thanks to advances in VLSI technology.

Owing to the aforementioned reasons the rapid growth of communication systems both in wireline and wireless communications took off. In wireline communications, digital subscriber line (DSL) technology has been gaining popularity as a high speed network access technology, capable of the delivery of multimedia services [6]. A major impairment for DSL is impulse noise in the telephone line. In wireless communications, the interference from co-channel and adjacent channel are major impairments [7]. The co-channel interference (CCI) in presence of Gaussian noise is successfully modelled as non-Gaussian noise. It is well known that non-Gaussian noise can cause significant performance degradation in traditional communication systems designed under the Gaussian stationary linear assumption [8] [9] [10]. In [8] it is shown that by using non-parametric techniques and relaxing the Gaussian noise assumption the performance of global system for mobile (GSM) receiver can be improved in interference limited channels. In [9] and [10] it is shown that for impulsive noise channels Gaussian assumption based signal processing is not viable, and other statistically based signal processing algorithms lead to improvement in performance. Also a well known example is the matched filter for coherent reception of deterministic signals in Gaussian white noise. If the noise statistics deviate from the Gaussian model, serious degradation in performance occurs, such as increased false alarm rate or error probability [11] [12].

That means, when the performance degradation due to the ideal Gaussian assumption in a non-Gaussian environment can not be tolerated, the underlying signal processing methods must be revisited and redesigned taking into account the non-Gaussian noise statistics. Thus finding better signal processing techniques based on exploiting this non-Gaussian phenomenon motivates

the research in this thesis. Some of the advantages of using such approaches in equalisation has already been witnessed in [8] [13] [14] and [15].

1.2 Thesis contributions

In this thesis, new signal processing algorithms for channel equalisation and estimation are proposed. The proposed algorithms exploit the non-Gaussian behaviour of noise in channel estimation and symbol detection/equalisation.

The first part of the thesis is concerned with the development of a minimum bit error rate (MBER) decision feedback equaliser (DFE) working in impulsive noise environments. The impulsive noise is modelled as an α-stable distribution as in [13]. In [13] it was observed that the Bayesian equaliser working in an α-stable environment performs better than the Gaussian noise based algorithms in similar environments. Moreover, an MBER version of algorithms for Gaussian noise channels was shown to perform better than least mean square (LMS)-based algorithms in [14] [15]. In this thesis, an MBER equaliser for α-stable distribution is proposed. The comparison of the proposed equaliser with the traditional LMS-based equaliser and Gaussian assumption based MBER equaliser is also explored.

The later part of the thesis concentrates on the development of non-parametric techniques for channel estimation and equalisation. An adaptive non-parametric channel estimator for non-Gaussian noise is proposed. The proposed channel estimator is found to be robust for both impulsive noise and co-channel interference limited communication channels. Both the impulsive noise and co-channel interference, type of non-Gaussianity is modelled as a mixture of finite Gaussian processes. Analysis on the step-size selection for the proposed channel estimator is also developed in this thesis. Channel estimation in the time domain based communication system is considered first in this thesis. The performance of the proposed algorithm is compared with the theoretical lower bounds defined by the Cramér Rao bound. The concept of totally adaptive channel estimator where the channel tap length is also dynamically adapted along with the tap coefficients is also explored in the thesis.

Since CCI is correlated in practice, a whitening filter based solution is presented. An error whitening based technique is used to reduce the correlation and then estimate the channel based on this whitened error log-likelihood function. The proposed channel estimation algorithm is then modified for the frequency domain based communication systems. A non-

parametric symbol-by-symbol maximum *a posteriori* probability (MAP) equaliser for orthogonal frequency domain multiplexing (OFDM) communication systems is proposed. Considerable performance gains are achieved by using this MAP equaliser with the non-parametric channel estimator.

1.3 Thesis outline

The rest of the thesis is organised as follows:

Chapter 2 presents background and puts the work described in the thesis in perspective. A brief review on adaptive signal processing with applications to communications channel estimation and equalisation are the focus of this chapter. The models and notations used throughout the thesis are also explained in this chapter.

Chapter 3 is devoted to development of the minimum bit error rate adaptive decision feedback equaliser for impulsive noise environment. A Wiener solution for the said simulation environment is formulated for performance analysis. Comparison with the MBER algorithm for Gaussian distribution and LMS-algorithm is also presented in this chapter.

Chapter 4 discusses an adaptive non-parametric channel estimator for non-Gaussian noise environments. Two specific cases of non-Gaussian noise are analysed throughout the chapter. A comparison with popular techniques for channel estimation in non-Gaussian noise environments is drawn in this chapter. The performance of the algorithm is evaluated for the impulsive noise and (correlated and uncorrelated) the co-channel interference in the presence of Gaussian noise.

Chapter 5 is an extension of the algorithm developed in the previous chapter. Here the case of channel estimation with whitening filter in co-channel interference limited communication systems is considered. Monte Carlo simulations support the viability of the framework discussed in this chapter.

Chapter 6 provides a non-parametric channel estimator and a non-parametric symbol-by-symbol MAP equaliser for OFDM communication systems. As the OFDM is increasingly used in next generation of mobile devices, interference amongst different communication standards working in-band and in vicinity have become a major concern. The proposed algorithm is tested for in-band interference limited communication system.

Chapter 7 summarises the work undertaken in this thesis and points to possible directions for future research.

Chapter 2
Background

The term *estimator* or *filter* is commonly used to refer to a system that is designed to extract information about a prescribed quantity of interest from noisy data. With such a broad aim, estimation (filter) theory finds applications in many diverse fields: communications, radar, sonar, navigation, seismology, biomedical engineering, and financial engineering. In this thesis the focus is adaptive channel estimation and equalisation for digital communication systems where the noise is considered as impulsive and in the presence of interference from other sources. This chapter gives an overview of adaptive signal processing, for detailed discussion on adaptive signal processing the reader is referred to work in [16] [17] [18]. In order to present the work in this thesis in perspective and introduce the nomenclature used throughout the thesis, a background survey on adaptive signal processing is presented in this chapter with digital communications as an application area.

First a brief description of linear optimal filters and adaptive filters is presented. Some typical classes of adaptive filters are discussed in section 2.3. Two application areas of adaptive signal processing are considered in section 2.4, where first the equalisation problem, followed by the channel estimation problem are discussed.

2.1 Linear optimal filter

In this section, a brief overview about linear optimal filtering is presented. Consider a filter

$$y(k) = \sum_{i=0}^{\infty} h(i)x(k-i), \;\; k = 0, 1, 2, ... \tag{2.1}$$

where $y(k)$ is the linear convolution sum of input $x(k)$ and filter $h(i)$ with i being the index to the number of taps (or delay elements). From Figure (2.1),

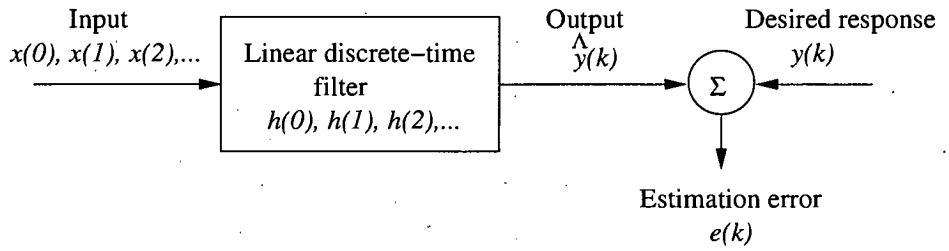


Figure 2.1: *Block diagram representation of linear filtering problem*

$$
\begin{aligned}
x(k) &= \text{input signal applied to the adaptive filter;} \\
y(k) &= \text{received, desired signal;} \\
\hat{y}(k) &= \text{output of adaptive filter;} \\
e(k) &= y(k) - \hat{y}(k) = \text{estimation error}
\end{aligned}
$$

At some discrete time k, the filter produces an output $\hat{y}(k)$. This output is used to provide an estimate of a desired response designated by $y(k)$. In particular, the estimation error, denoted by $e(k)$, is defined as the difference between the desired response $y(k)$ and the estimated response $\hat{y}(k)$. The requirement is to make the estimation error $e(k)$ "as small as possible" in some statistical sense. The filter is assumed to be linear to make the mathematics simpler. Also it is assumed that the filter operates in discrete time to make its implementation on digital computer hardware or software possible.

The purpose of the filter in Figure 2.1 is to produce an estimate of the desired response $y(k)$. We assume that the filter input and the desired response are single realisations of a jointly wide sense stationary stochastic process, with zero mean. To optimise the filter design, we choose to minimise the mean-square value of $e(k)$. We thus define the cost function as the mean-square error

$$J = \mathbb{E}\{e^2(k)\} \tag{2.2}$$

where $\mathbb{E}\{.\}$ denotes the statistical expectation operator. The requirement is therefore to determine the operating conditions under which J attains its minimum value. For the cost function J to attain its minimum value, all the elements of the gradient vector ∇J must be simultaneously

equal to zero; that is

$$\nabla_i J = 0, \quad i = 0, 1, 2, ... \tag{2.3}$$

Under this set of conditions, the filter is said to be optimum in the mean-square error sense, where

$$\nabla_i J = 2\,\mathbb{E}\left\{ e(k)\frac{\partial e(k)}{\partial h(i)} \right\} \tag{2.4}$$

Thereby taking gradient of eq. (2.2) and then cancelling common terms, it finally yields

$$\nabla_i J = -2\,\mathbb{E}\{x(k-i)e(k)\} \tag{2.5}$$

that is equivalent to

$$\mathbb{E}\{x(k-i)e_o(k)\} = 0 \tag{2.6}$$

which represents the special value of e_o that results when the filter operates in its optimum condition. The above equation forms the powerful *principle of orthogonality*.

From the Wiener-Hopf equations we see that,

$$\mathbb{E}\{x(k-i)(y(k) - \sum_{v=0}^{\infty} h_o(v)x(k-v))\} = 0 \tag{2.7}$$

where $h_o(v)$ is the vth coefficient in the impulse response of the optimal filter (subscript 'o') also known as the Wiener filter. Expanding and re-arranging the terms,

$$\sum_{v=0}^{\infty} h_o(v)\mathbb{E}\{x(k-i)x(k-v)\} = \mathbb{E}\{x(k-i)y(k)\} \tag{2.8}$$

The two expectation in eq. (2.8) are interpreted as follows:

1) The expectation $\mathbb{E}\{x(k-i)x(k-v)\}$ is equal to the autocorrelation function of the filter input for lag $v-i$. It can be expressed as,

$$r_{xx}(v-i) = \mathbb{E}\{x(k-i)x(k-v)\} \tag{2.9}$$

2) The expectation $\mathbb{E}\{x(k-i)y(k)\}$ is equal to the cross-correlation between the filter input

$x(k-i)$ and the desired response $y(k)$ for a lag of $-i$, thus

$$r_{xy}(-i) = \mathbb{E}\{x(k-i)y(k)\} \tag{2.10}$$

By using the definitions of eq. (2.8) and eq. (2.9) in eq. (2.10), an infinite large system of equations as the necessary and sufficient condition for the optimality of the filter:

$$\sum_{v=0}^{\infty} h_o(v) r_{xx}(v-i) = r_{xy}(-i) = r_{yx}(i) \tag{2.11}$$

The eq. (2.11) defines the optimal filter coefficients, in the most generalised setting. However, in practice there are certain constraints in using Wiener filters for many applications. Firstly, the statistics of $x(k)$ and $y(k)$ may not be known, thus exact knowledge of $r_{xx}(v-i)$ and $r_{xy}(i)$ is not guaranteed. In many applications the statistics change with time. Lastly, computing the inverse of $r_{xx}(v-i)$ may constrain the use of Wiener filters in real time application. These applications involve processing of signals that are generated by systems whose characteristics are not known *a priori*. Under this condition, a significant improvement in performance can be achieved by using adaptive rather than Wiener (or fixed) filters.

2.2 Adaptive filters

An adaptive filter is a self-designing filter that uses a recursive algorithm (known as adaptation algorithm or adaptive filtering algorithm) to design itself. The algorithm starts from an initial guess, chosen based on the *a priori* knowledge available to the system, then refines the guess in successive iterations, and converges, eventually, to the optimal Wiener solution in some statistical sense. The performance of an adaptive filtering algorithm is evaluated based on one or more of the following factors [16]:

Rate of convergence: This quantity describes the transient behaviour of the algorithm. This is defined as the number of iterations required for the algorithm, under stationary conditions, to converge close enough to the optimum Wiener solution in the mean square sense.

Misadjustment: This quantity describes steady-state behaviour of the algorithm. This is a quantitative measure of the amount by which the ensemble averaged final value of the mean-squared error exceeds the minimum mean-squared error produced by the optimal Wiener filter.

Computational Requirements: This is an important parameter from a practical point of view. The parameters of interest include the number of operations required for one complete iteration of the algorithm and the amount of memory needed to store the required data and also the program. These quantities influence the price of the computer needed to implement the adaptive filter.

Numerical Robustness: The implementation of adaptive filtering algorithms on a digital computer, which inevitably operates using finite word-lengths, results in quantisation errors. These errors sometimes can cause numerical instability of the adaptation algorithm. An adaptive filtering algorithm is said to be numerically robust when its digital implementation using finite-word-length operations is stable.

Another practical measure of performance is the number of computations needed for the adaptive filter to reach steady state. This measure combines the rate of convergence and computational requirements and is the product of the number of iterations needed for the algorithm to converge close enough to the optimum solution and the number of computations needed per iteration [19].

However with all the above considerations and practical constraints, the most common linear adaptive filters can be modelled as a tap-delay line filter. The aim is to find the solution to Wiener-Hopf equations; an iterative approach is used to obtain the solution. The most common method of linear optimisation of steepest-gradient [16] is used. The cost function, also referred to as the "index of performance"; defined as mean square error is used here. This method requires the use of gradient vector, the value of which depends on two parameters: the correlation matrix of the tap inputs in the linear filter and the cross-correlation vector between the desired response and the same tap inputs. When the instantaneous value for correlation described in eq. (2.8) is used, so as to drive an estimate of the gradient vector, making it assume a stochastic character, it is referred to as the LMS algorithm. In essence it can be represented as

$$\begin{pmatrix} \text{updated value} \\ \text{of tap} - \text{weight} \\ \text{vector} \end{pmatrix} = \begin{pmatrix} \text{old value} \\ \text{of tap} - \text{weight} \\ \text{vector} \end{pmatrix} + \begin{pmatrix} \text{learning} - \\ \text{rate} \\ \text{parameter} \end{pmatrix} \begin{pmatrix} \text{tap} - \\ \text{input} \\ \text{vector} \end{pmatrix} \begin{pmatrix} \text{error} \\ \text{signal} \end{pmatrix}$$

where the "learning rate parameter" defines the rate of adaptation. Too large a value of this

parameter this algorithm may never converge and too small a value could result in longer time to converge. Hence, this parameter value is application dependent and on the type of cost function used. It is usually defined over a range of values. There are various modification (and/or combinations) to LMS algorithm available such as leaky LMS, normalised LMS, frequency domain LMS, block LMS, signed LMS and variable step size LMS. Another approach to the development of linear adaptive filtering is based on the method of least squares. In this method, the cost function is defined as the sum of weighted error squares, where the error is itself defined as the difference between the derised and the actual filter output. One of the most popular methods that uses the method of least squares is recursive least squares (RLS) algorithm. Other forms than standard RLS include square-root RLS and Fast RLS [16].

2.3 Classes of application

The ability of adaptive filters to work satisfactorily in the unknown environment and tracking the variations in the system has made adaptive filters attractive for control and signal processing community. The application areas of the adaptive filters is thus enormous communications, radar, sonar, seismology, and biomedical engineering. The application areas are quite diverse, they usually have one basic thing in common: An input vector and a desired response, which are used to compute an estimation error, which in turn is used to control the values of a set of adjustable filter coefficients. The major difference between various applications of adaptive filters is the way the problem is defined, however four broad classifications for adaptive filters can be made [16].

The four basic classes of the adaptive filtering applications depicted in Figure 2.2-2.5 are as follows:

1) *System Identification*: The system identification as shown in Figure 2.2 uses an adaptive filter to provide a best fit to the unknown "Plant". The same input is fed to the plant and the adaptive filter, and respective outputs compared. Adaptive filter's coefficients are updated by some criterion, like LMS or RLS, using the estimation error.

2) *Inverse Modelling*: The adaptive filter in this class, finds the best fit to the inverse of the unknown Plant as shown in Figure 2.3. The requirement is to have "System output" the same as the delayed "System input", thereby requiring adaptive filter to compensate the effect of the plant.

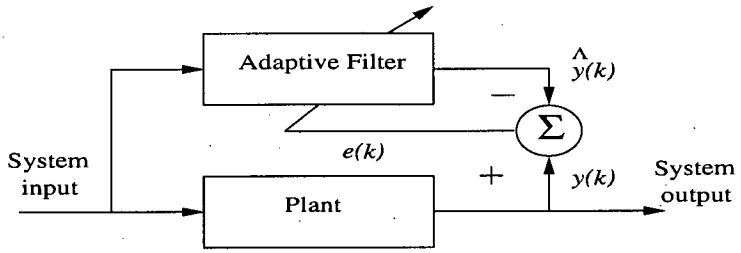


Figure 2.2: *Adaptive filter applied for system identification*

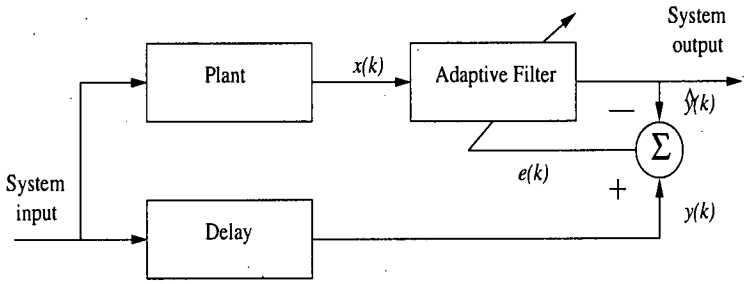


Figure 2.3: *Adaptive filter applied for inverse modelling*

3) *Prediction*: The function of the adaptive filter is to predict the current sample based on the past samples as in Figure 2.4. Thus the adaptive filter uses the past information to find the best fit to the current input (desired) signal. Depending on the application of interest, the adaptive filter output or the estimation (prediction) error may serve as the system output. In the first case, the system operates as a predictor; in the latter case, it operates as a prediction-error filter.

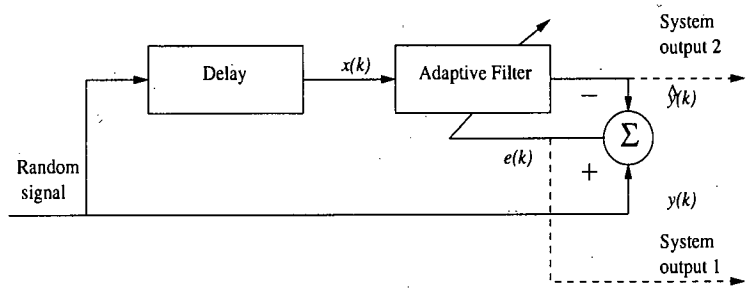


Figure 2.4: *Adaptive filter applied for prediction*

4) *Interference cancellation*: The unknown interfering signal is cancelled from the primary signal (information alongside interference signal), with cancellation being optimised in certain

sense. The primary signal serves as the desired response for the adaptive filter as in Figure 2.5. The reference signal is applied to the adaptive filter, where it is assumed that the information-bearing signal component is weak or essentially undetectable.

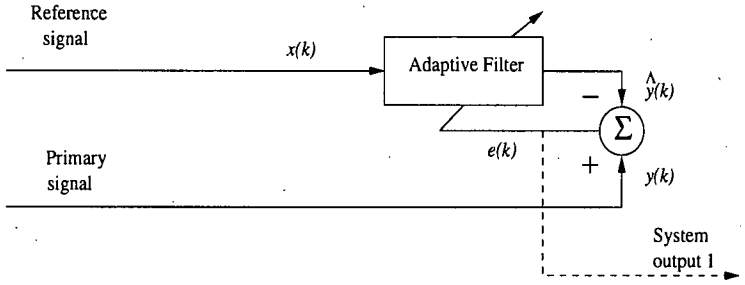


Figure 2.5: *Adaptive filter applied for interference cancellation*

2.4 Applications

The various classes of adaptive filter discussed above are applied in some-way or another in modern digital communication systems. A typical digital communication system is shown the Figure 2.6. The "data source" constitutes the signal generation system that generated the information to be transmitted. Some of the typical examples are speech coders, video coders and scanners. The raw data (information) is then coded by the "encoder". The "encoder" adds redundancy to the transmitted information to add reliability to the transmitted data. Some of the typical examples are "convolutional codes", "gray codes" and "block codes". The digital data transmission requires very large bandwidth. The efficient use of available bandwidth is achieved through the "transmitter filter", also called the modulating filter. The modulator on the other hand places the signal over a high frequency carrier for efficient transmission. Some of the typical schemes used in the digital communication are amplitude shift keying (ASK), frequency shift keying (FSK), pulse amplitude modulation (PAM), phase shift keying (PSK) modulation and quadrature phase shift keying (QPSK). The "channel" is the medium through which the information propagates from the transmitter to the receiver. The channel can be fixed, flat or multipath fading depending on the application area. At the receiver the signal is first "demodulated" to recover the baseband transmitted signal. This demodulated signal is processed by the "receiver filter", also termed as receiver demodulating filter, which should ideally match to the transmitted filter and channel. Normally, the propagation channel is not

known at the receiver, thus the receiver filter is matched to the transmitter filter only. The "equaliser" in the receiver removes the impairments, inter-symbol interference (ISI) and ICI introduced due to the communication channel. The "decision device" provides the estimate of the encoder transmitted signal. The "decoder" reverses the work of the encoder and removes the encoding effect revealing the transmitted information symbols.

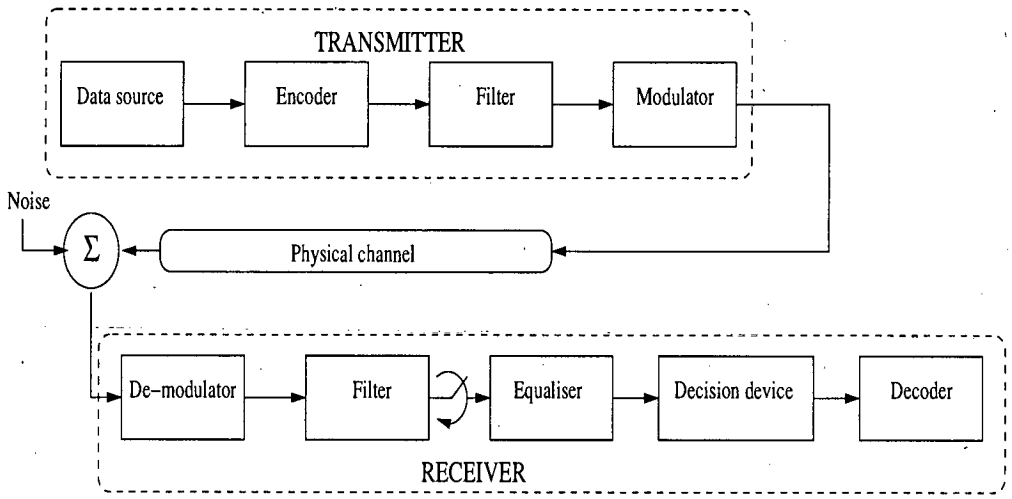


Figure 2.6: *A typical digital communication system*

2.4.1 Propagation channel

This section discusses the channel impairments that limit the performance of a digital communication system. The ideal transmission of the digital pulses over an analogue communication channel would require infinite bandwidth. A band-limited channel such as a telephone channel is characterised as a linear filter having an equivalent low-pass frequency-response characterstics, $H_C(f)$ [20], defined as:

$$H_C(f) = \mid H_C(f) \mid \exp(j\theta(f)) \tag{2.12}$$

where $H_C(f)$ represents the Fourier transform (FT) of the channel and $\theta(f)$ represents the phase response of the channel. The amplitude response of the channel $\mid H_C(f) \mid$ can be defined as,

$$\mid H_C(f) \mid = \begin{cases} \kappa_1, & \mid f \mid \leq f_c \\ 0, & \mid f \mid > f_c \end{cases} \tag{2.13}$$

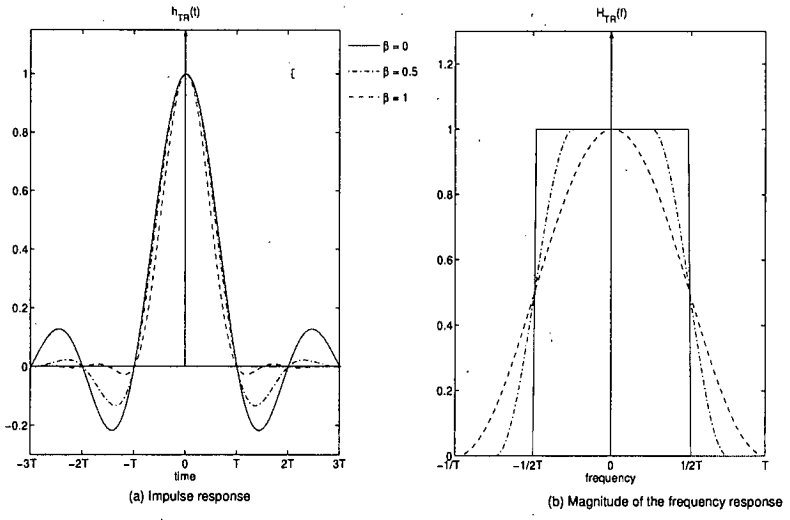


Figure 2.7: *Raised cosine pulse and its spectrum*

where κ_1 is a constant and f_c is the upper cut-off frequency. The channel group delay characteristic is given by,

$$\tau(f) = -\frac{1}{2\pi}\frac{d\theta(f)}{df} = \kappa_2 \tag{2.14}$$

where κ_2 is an arbitrary constant. The conditions described in eq. (2.13) and eq. (2.14) constitute fixed amplitude and linear phase characteristics of a channel. This channel can provide distortion free transmission of analogue signal band limited to f_c. Transmission of the infinite bandwidth digital signal over a band limited channel of f_c will obviously cause distortion. This demands for the infinite bandwidth digital signal be band limited to at least f_c, to guarantee distortion free transmission. This work is done with the aid of the transmitter and receiver filters shown in Figure 2.6. The combined frequency response of the physical channel, transmitter filter and the receiver filter can be represented as,

$$H(f) = H_T(f)H_C(f)H_R(f) \tag{2.15}$$

where, $H_T(f)$, $H_C(f)$, and $H_R(f)$ represent the FT of the transmitter filter, propagation channel, and the receiver filter respectively. When the receiver filter is matched to the combined response of the propagation channel and the transmitter filter, the system provides optimum signal to noise ratio (SNR) at the sampling instant [7] [20]. Since the channel response is not

known beforehand, the transmitter and receiver filter are matched, thus

$$H_R(f) = H_T^*(f) \tag{2.16}$$

$$h_R(t) = h_T(-t) \tag{2.17}$$

where * means complex conjugates, and $h_R(t)$, $h_T(t)$ denote time-domain representation of receiver and transmitter filter. For ideal channel presented in eq. (2.12), the design of the transmitter and receiver filters is critical for achieving distortion free transmission. One such filter capable of satisfying this criterion is the raised cosine filter given by,

$$H_{TR}(f) = \begin{cases} T, & 0 \leq f \leq \frac{1-\beta}{2T} \\ \frac{T}{2}\{1 + \cos[\frac{\pi T}{\beta}(\mid f \mid - \frac{1-\beta}{2T})]\}, & \frac{1-\beta}{2T} \leq \mid f \mid \leq \frac{1+\beta}{2T} \\ 0, & \mid f \mid > \frac{1+\beta}{2T} \end{cases}$$

$$H_{TR} = H_T(f)H_R(f) \tag{2.18}$$

T is the source symbol period and β, $0 \leq \beta \leq 1$, is the excess bandwidth and H_{TR} is the FT of the combined response of transmitter and receiver filter. The plot of this combined filter response is shown in Figure 2.7. Figures 2.7-(a) and 2.7-(b) represent the time and frequency domain responses of the combined filter respectively. A distortion free transmission can be achieved, if the receiver output is sampled at the correct time. A sampling timing error causes inter-symbol interference (ISI), which reduces with an increase in β. In the time-domain the impulse response in Figure 2.7-(a) can be represented as:

$$h_{TR}(t) = \frac{\sin(\frac{\pi t}{T})}{\frac{\pi t}{T}} \frac{\cos(\frac{\pi \beta t}{T})}{1 - (\frac{4\beta^2 t^2}{T^2})} \tag{2.19}$$

and for the special case of $\beta = 0$ provides a pulse satisfying the condition,

$$h_{TR}(t) = \frac{\sin(\frac{\pi t}{T})}{\frac{\pi t}{T}} \tag{2.20}$$

Under this condition the channel can provide the highest signalling rate, $T = \frac{1}{2f_c}$. At the other extrema, $\beta = 1$ provides a signalling rate equal to the reciprocal of the bandwidth, $T = \frac{1}{f_c}$. In this process selection of β provides a compromise between quality and signalling speed.

Here, it has been assumed that the physical channel is an ideal low pass filter eq. (2.12).

However, in reality all physical channels deviate from this behaviour. This introduces ISI even though the received signal is sampled at the correct time. The presence of this ISI requires an equaliser to provide proper detection.

In general all types of communication systems are effected by ISI. The combined channel due to the transmission filter, propagation channel and the receiver filter and the T-spaced sampler can be modelled by the digital finite impulsive response (FIR) filter represented in Figure 2.1. Here the channel observed output $y(k)$ is given by the sum of the noise free channel output $\bar{y}(k)$ and noise. $\bar{y}(k)$ in turn is formed by the convolution of the transmitted sequence $x(k)$ with the channel taps $h(i)$, $0 \leq i \leq N_T - 1$, where N_T is the number of taps. The z-domain transfer function of the impulse response can be represented by the equation

$$H(z) = \sum_{i=0}^{N_T-1} h(i)z^{-i} \tag{2.21}$$

the channel provides a dispersion of up to N_T samples. This discrete time model will be used throughout the thesis, with minor modification, which will be explained in that chapter in details.

2.4.2 Equalisation

As observed in the previous section, ISI in the communication channel is almost impossible to avoid, and therefore an equaliser forms a vital part of a modern day communications receiver. In general the family of adaptive equalisers can be divided into supervised equaliser and un-supervised equaliser. Supervised equalisation requires a set of training sequence to be transmitted. A replica of this training sequence is available at the receiver and the comparison of the two sequences with a certain rule form the supervised equaliser. After this initial training, the equaliser is then switched to decision directed mode, where the equaliser can update its parameters based on the past detected samples. In digital television and digital radio there is no scope for the use of training signal, hence the equalisers used in these applications are called un-supervised or blind-equalisers. The equaliser can also be categorised as linear and nonlinear equalisers based on their structure.

The linear equaliser often provides sufficient performance over typical telephone channels for data transmission. However, for a typical radio channel with multipath propagation and large delay spread, the linear equaliser forms a poor choice. The nonlinear equalisers are used in such

situations where distortion is too severe for a linear equaliser to handle. Two very effective nonlinear equalisation techniques which offer substantial improvements compared to linear equalisation techniques commonly used are: a) decision feedback equaliser and b) maximum likelihood sequence estimator.

The basic idea behind the DFE [21][22][23] is that when a decision has been made on an information (input) symbol, the ISI that it induces to the following symbols can be estimated and subtracted out before detection of the successive symbols. In principle the feedback part takes away the ISI caused by earlier detected symbols [24]. Several different cost functions are available to optimise the equaliser's performance, namely, the peak distortion criterion [25][1][26], the minimum mean square error (MMSE) criterion [25][20] and the MBER criterion [27][28][14]. The equaliser optimised for peak distortion criterion is called the zero-forcing (ZF) equaliser. In recent years however, the ZF equaliser has become less popular [20]. The current implementations of equaliser are normally based on the MMSE and MBER criterion. In developing training strategies for DFEs, it is convenient to adopt an MMSE cost function as this facilitates the use of standard adaptive "filter techniques" such as the LMS and recursive least squares (RLS) algorithms. However it has long been understood that the MMSE cost function is not optimal in this application; the MBER cost function being the more appropriate choice [27]. Further, the BER rate of a DFE optimised using an MMSE criterion can be distinctly inferior to the true optimum solution [14]. Over the recent years there has been considerable interest in developing the MBER based equalisers and their modifications [14][15] [28][29]. The MBER cost function has also shown performance improvements in various applications such as adaptive beamforming [30] [31], multiuser detection for direct sequence code division multiple access (DS-CDMA) systems [32] [33], multiuser detection for OFDM systems [34] and many more. In general, the relative performance of equalisers designed using MMSE and MBER criteria is very much dependent upon specific channel conditions. Further exposition on MBER criterion is provided in the next chapter.

The other popular nonlinear equaliser is the MLSE equaliser, which forms a robust equaliser for various channel conditions. The basic idea behind the MLSE is to test all possible data sequences and choose the data sequence with the maximum probability as the output [3]. This implies that a MLSE scheme has a large computational requirement compared to traditional methods where the decoding is carried out symbol-by-symbol. MLSE is optimal in the sense that it minimises the probability of sequence error. MLSE can be implemented by using the

Viterbi algorithm thereby reducing the computational complexity of verifying all the possible combinations [4]. The basic block diagram of a MLSE equaliser [35] is shown in Figure 2.8. The MLSE requires knowledge of the channel characteristics in order to compute the metrics for making decisions and also knowledge of the statistical distribution of the noise corrupting the signal. Thus, the probability distribution of the noise determines the form of the metric for optimum detection of the received signal. In case of Gaussian noise the metric can be reduced to the calculation of the Euclidean distance. From [8] we observe that in the presence of co-channel interference the noise deviates from Gaussianity and thus the Euclidean distance is not an appropriate metric. The probability distribution of noise (including interference) is used as the metric in [8]. The MLSE equaliser's performance is sensitive to the quality of the channel estimate, thus a good channel estimate can vastly improve the MLSE equaliser's performance.

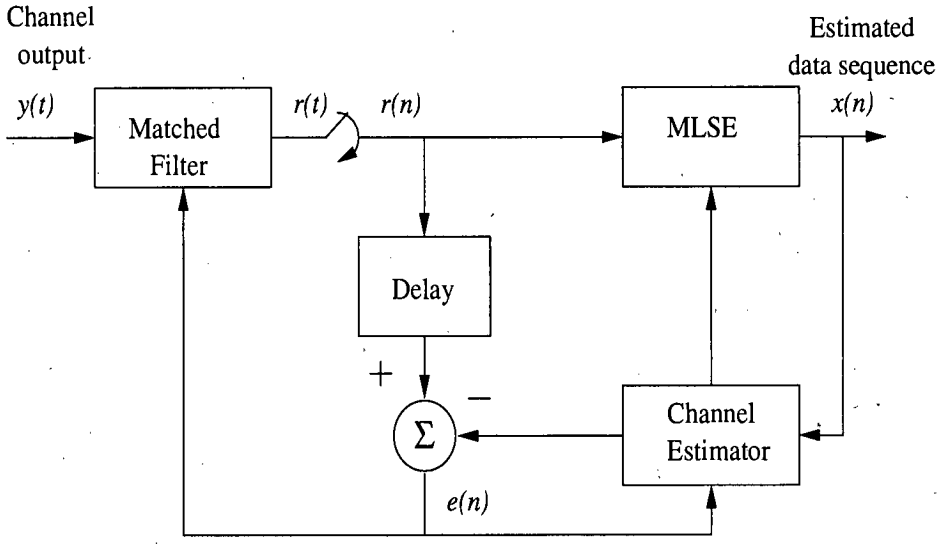


Figure 2.8: *Block diagram of MLSE equaliser*

2.4.3 Channel estimation

As discussed earlier a good channel estimate can vastly improve a communication receiver's performance. The channel estimation in communication systems can be done in three ways: 1) blind, 2) semi-blind, and 3) training based.

In blind channel estimation techniques, the channel is estimated without the exact knowledge of the transmitted sequence. This technique is attractive as the overall throughput is higher, as no bits are lost in training. However, blind estimation techniques require large amount of

data to be stored before channel estimation can begin, which invariably introduces delays. For those channel estimators based on a single-input single-output (SISO) channels, higher than second-order statistics or nonlinear optimisation are often required [36] [37] [38], which causes problems such as local and slow convergence [39] [40] for blind channel estimation.

Although non-data-aided or blind techniques for channel estimation have received considerable attention recently, many current digital communications systems employ a pilot sequence to probe the channel. The training based techniques estimate the channel by transmitting a known (at the receiver) training sequence (also known as pilots) along with the unknown data at the receiver. The receiver estimates the communication channel based on estimating the change in these training bits due to the channel. This technique provides a simple yet effective way of retrieving information about the channel, therefore facilitating all succeeding processing steps such as equalisation and symbol detection [41]. Training sequence also help in synchronisation and user identification.

There are various estimators/techniques such as the least squares (LS) estimator, maximum-likelihood (ML) estimator [42], expectation maximization (EM) [43], and methods of moments [44] are used for training based channel estimation in practice. However, the most common training sequence based channel estimator is the traditional LS estimator. The LS channel estimator forms an optimal estimator for channel estimation in Gaussian noise environments [42]. However when the communication channel is affected by an interfering user then the traditional LS based estimation technique does not suffice. Interference when considered together with the Gaussian noise can be represented as a Gaussian mixture [8]. The channel estimation problem can be viewed as a parameter estimation problem in Gaussian mixture. To find the ML parameter estimate of a Gaussian mixture the EM based techniques are widely used [45][46][47]. The EM algorithm basically breaks the Gaussian mixture into an assumption that each observation is from one of the mixture components which is Gaussian distributed. Thus the complex parameter estimation in Gaussian mixture is broken down into parameter estimation of many simpler estimation problems by the EM algorithm.

Semi-blind estimation algorithms have been proposed in [48] [49] [50] in anticipation of improved performance. These algorithms suggest the use of all the known information in the transmitted frame in contrast to either blind-only or training-only estimation algorithms. A deterministic and Gaussian ML approach is taken in [48] [49] and associated Cramér-Rao bounds are derived. In [50], a semi-blind cost function is proposed by combining a training and blind

cost function. In [51] a stochastic ML semi-blind channel estimation method is used. Fortunately, the hidden Markov model framework and associated estimation algorithms [52] [53] [54] [55] [56] provide a computationally efficient solution to the resulting optimisation problem. The semi-blind techniques try to reduce the size of the training sequence by exploiting both the known and the unknown (blind) portions of the data. There are various other semi-blind channel estimation techniques discussed in [57]. Recently, the semi-blind techniques have been applied for rapidly varying channels in [58], for 3rd-generation (3G) code division multiple access (CDMA) standard in [59] and for OFDM fading channels in [60].

With blind, semi-blind, and training based channel estimation methods discussed above, the training based methods still remains the most popular. One such successful example is the GSM communication systems receiver, where MLSE equaliser is used with a channel estimator as in Figure 2.8. Channel estimation of wireless channels becomes challenging with many criterion effecting the estimation. The wireless channel could be a fast or slow fading channel with flat or multipath fading. Thus considerable research has been undertaken in [43] [44] [61] [10] [62] where statistical methods are used to get a reliable channel estimate. With the recent proliferation of wireless communication devices, the communication channel is increasingly becoming interference limited rather than noise limited as considered in past. Thus alternative techniques based on joint estimation [63] and [64] have been proposed for communications receiver. Recently, with the advent and advantages of using multiple antennas, some techniques based on multiple antennas with whitening (linear prediction filter as in section-2.3) are discussed in [65] and [66].

2.5 Conclusions

In this chapter a brief overview of the adaptive filters and their various configurations were discussed in the beginning. A typical digital communication system was also presented. The constraints of a practical communication system and need for adaptive signal processing in a typical communications receiver was discussed in details. A brief review of the communication channel equaliser and estimator was provided at the end.

Chapter 3
Stochastic gradient algorithm for equalisers in alpha stable noise

Equalisers are used to combat ISI at the receiver in a communication channel as discussed in Chapter 1. Channel equalisation dates back to the early work of Lucky [1], Proakis and Miller [67], who established the theory of adaptive transversal (or tapped-delay-line) equalisers adjusted by the zero-forcing or the MSE criteria. The early work aimed almost entirely at the telephone channel, which can essentially be characterised as a linear time-invariant ISI channel. Later work was related to the line-of-sight microwave channel, which may be considered as a very slowly time-varying ionospheric and tropospheric channel. It was soon realised that linear equalisers were not able to provide the performance requirements for highly dispersive channels thus non-linear techniques were sought. DFE and trellis equalisers based on symbol-by-symbol MAP estimation or MLSE are the non-linear techniques used for equalisation [8]. Recently there has been renewed interests in the MBER equaliser [14], [15] which is discuss later in this chapter.

In some communication channels, the observation noise exhibits impulsive, as well as Gaussian characteristics. The sources of impulsive noise may be either natural (e.g. lightning, ice-cracking), or man-made. It may include atmospheric noise or ambient noise. It might come from relay contacts, electro-magnetic devices, electronic apparatus, or transportation systems, and switching transients [68] [69], as a result causing degradation in receiver's performance. Most of the present day systems are optimised under the Gaussian assumption and their performance is degraded by the occurrence of impulsive noise [9]. Impulsive noise is more likely to exhibit sharp spikes or occasional bursts of outlying observations than one would expect from Gaussian distributed signals. A variety of impulsive noise models have been proposed in [69] and [70]. However, a common model to represent impulsive phenomena is the family of α-stable random variables [71]. Stable distributions share defining characteristics with the Gaussian distribution, such as the stability property and generalised central limit theorem. Empirical data indicate that the probability density functions (pdf's) of the impulsive noise pro-

cesses exhibit a similarity to the Gaussian pdf, being bell shaped, smooth and symmetric, but at the same time having significantly heavier tails [68].

In [15] it was shown that adaptive linear equalisation based on probability of error performs better than that based on a least squared error cost function. Further, in [14] it was shown that the state-translated design with the MBER critera achieves a lower BER than conventional DFE structure. However the adaptive least BER (LBER) (with or without state translation) algorithm of [15] was derived on the assumption that the noise was drawn from a Gaussian distribution.

The general purpose adaptive algorithms for α-stable noise environments have been proposed in the literature [71] and [72], however they are based on the L_p norm (where $0 < p < \alpha$) of the error rather than BER. In this chapter, a class of adaptive equalisers (similar in complexity to the LMS algorithm) where the BER is minimised in an α-stable noise environment is developed. Generally, in adaptive equalisation, the Wiener solution is taken as a point of reference in measuring performance. However in α-stable noise the variance of the input signal to the equaliser is theoretically infinite and thus the Wiener solution is not defined. In practice, every receiver has a finite input dynamic range which *limits* the amplitude of the received samples and produces finite variances. As pointed out in [73] the limiter facilitates the use of standard correlation based algorithms in α-stable noise. Using this *limiter* the 'Wiener solution with limiter' (WSL) for α-stable noise environments is derived. Simulation results show that the LMS algorithm fails to converge to this WSL solution while the proposed α-stable noise LBER algorithm seeks the optimum BER solution for comparable computational complexity. Robustness of the Gaussian noise LBER algorithms of [15] in α-stable noise is also suggested through simulation.

The chapter is organised as follows: a brief overview of stable processes is provided in section 3.1; an overview of the state-translated DFE structure is presented in section 3.2; the WSL in α-stable noise is derived in section 3.3; the LBER adaptive algorithm for α-stable noise is derived in section 3.4; simulation techniques, assumptions and results are discussed in section 3.5; finally conclusions are drawn in section 3.6.

3.1 The class of stable random variables

The family of stable random variables is defined as a direct generalisation of the Gaussian law and in fact include the Gaussian density as a limiting case. The symmetric stable densities

have many features of the Gaussian distribution. They are smooth, unimodal, symmetric with respect to (w.r.t.) the median and bell-shape. However, the main characteristics of a non-Gaussian stable density function is that its tails are heavier than those of the normal density. This is one of the main reasons why the stable law is regarded suitable for modelling signals and noise of impulsive nature.

The univariate symmetric α-stable (SαS) pdf $f_\alpha(s)$ of a random variable (RV) S is defined by means of its characteristic function:

$$F_\alpha(\omega) = \exp(\delta i\omega - \gamma|\omega|^\alpha) \tag{3.1}$$

where $i = \sqrt{-1}$. The parameters α, γ and δ describe completely an SαS distribution.

The characteristics exponent, α

The characteristics exponent controls the heaviness of the tails of the stable density and hence the impulsiveness of the respective stable process. It can take values in the interval (0,2]; a smaller value of α implies heavier tails (i.e. severe impulsiveness), while a value of α close to 2 indicates a more Gaussian type behaviour. When $\alpha = 2$, the stable distribution is reduced to the Gaussian density.

The scale parameter, γ

The scale parameter, also called the *dispersion*, can be positive number. It plays an analogous role to variance and refers to the spread of the distribution. When $\alpha = 2$ the variance of the Gaussian distribution equals 2γ.

The location parameter, δ

This parameter is identical to the mean of the distribution. Throughout the thesis the SαS noise is assumed to be zero mean, hence the location parameter equals zero.

The α-stable distribution is the inverse Fourier transform of $F_\alpha(\omega)$, and can therefore be written as

$$f_\alpha(s) = \mathrm{FT}^{-1}\{F_\alpha(\omega)\} = \frac{1}{2\pi}\int_{-\infty}^{\infty} F_\alpha(\omega)\exp(j\omega s)d\omega \tag{3.2}$$

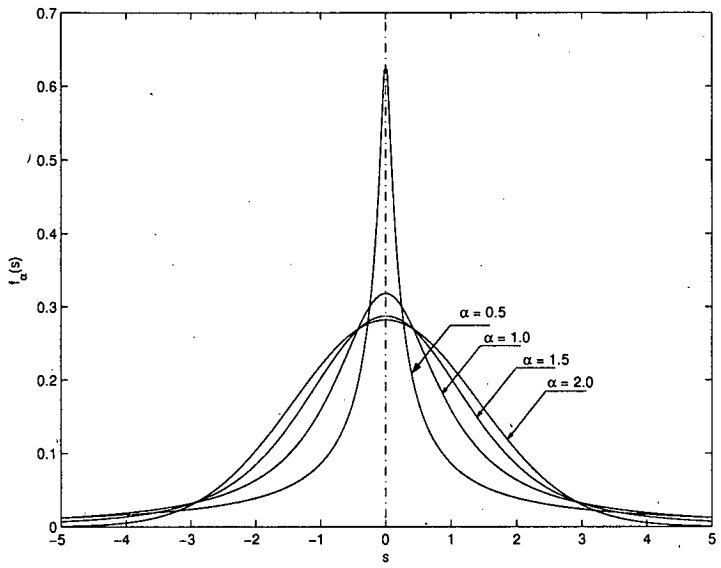


Figure 3.1: *The symmetric α-stable probability density function for four different values of the characteristic exponent α, including the Gaussian case ($\gamma = 1$ and $\delta = 0$)*

When the random variable is symmetric around zero ($\delta = 0$), then eq. (3.1) reduces to

$$F_\alpha(\omega) \mid_{\delta=0} = \exp(-\gamma|\omega|^\alpha) \tag{3.3}$$

in which case the characteristics function is real and even. That is, the density function can be written as

$$f_\alpha(s) = \frac{1}{\pi}\int_0^\infty \exp(-\gamma|\omega|^\alpha)\cos\omega s\, d\omega \tag{3.4}$$

Unfortunately, no closed-form expressions exist for the stable density, except for the Gaussian ($\alpha = 2$), Cauchy ($\alpha = 1$) and Pearson ($\alpha = \frac{1}{2}$) distributions [71] [74]. An important property of all non-Gaussian stable distributions is that only the lower moments are finite, so for example the Kurtosis is not defined for $\alpha < 2$. That is, if $\mathbf{s}$ is a non-Gaussian stable RV, then $\mathbb{E}_\mathbf{s}\{|s|^p\} < \infty$ iff $p < \alpha$. A well known consequence of this property is that all stable RV's with $\alpha < 2$ theoretically have infinite variance [9].

Figure 3.1 demonstrates the effects of α on the tails of a stable distribution. Four symmetric stable distributions are plotted, all with $\gamma = 1$ and $\delta = 0$ but with $\alpha = 0.5, 1.0, 1.5$ and 2.0. With $\alpha = 2$ corresponding to the Gaussian density with zero-mean and variance equal to 2, and

$\alpha = 1$ corresponding to the Cauchy density. The symmetric α-stable distributions are smooth, unimodal, symmetric with respect to the median and bell-shaped, just like the Gaussian density. A detailed comparison between the normal and the stable density functions shows that non-Gaussian stable distributions depart from the corresponding Gaussian density in the following ways. For small absolute values of s, the α-stable densities are more peaked than the normal. For some intermediate range of $\mid s \mid$, the α-stable distributions have lower densities than the normal. Most importantly, unlike the Gaussian distribution, the stable densities have algebraic tails which decay less rapidly. Further discussion on α-stable RV's and their properties can be found in [71].

3.2 Equaliser structures

The channel is modelled as a finite impulse response filter with an additive noise source, and the received signal at sample k (as in Figure 3.2) is

$$y(k) = \bar{y}(k) + w(k) = \sum_{i=0}^{N_T-1} h(i)x(k-i) + w(k) \tag{3.5}$$

where $\bar{y}(k)$ denotes the noiseless channel output; N_T is the channel length and $h(i)$ are the channel tap weights; the white noise $w(k)$ has zero mean and is drawn from an α-stable distribution with dispersion γ and characteristic exponent α; the symbol sequence $\{x(k)\}$ is independent and identically distributed (i.i.d.) and has an standard C-PAM constellation [20] defined by the set [14]

$$x_i = 2i - C - 1, 1 \leq i \leq C$$

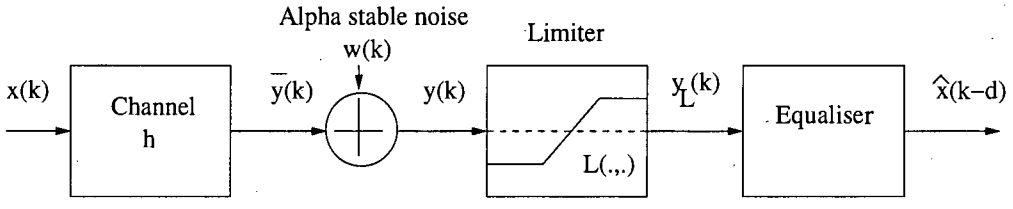


Figure 3.2: *Typical communication system*

Throughout this chapter $C = 2$ for 2-PAM is considered. For a conventional linear-combiner DFE (shown in Figure 3.3) the decision variable z at time k is a linear combination of received

samples and past decisions:

$$z(k) = \mathbf{a}^T \mathbf{y}(k) - \mathbf{b}^T \hat{\mathbf{x}}_b(k)$$

where $\mathbf{y}(k) = [y(k)\, y(k-1) \ldots y(k-m+1)]^T$ is the channel observation vector, $\hat{\mathbf{x}}_b(k) = [\hat{x}(k-d-1)\, \hat{x}(k-d-2) \ldots \hat{x}(k-d-n)]^T$ is the past detected symbol vector, $\mathbf{a} = [a(0)\, a(1) \ldots a(m-1)]^T$ is the feedforward coefficient vector and $\mathbf{b} = [b(1)\, b(2) \ldots b(n)]^T$ is the feedback coefficient vector. The integers d, m and n will be referred to as the decision delay, the feedforward delay and feedback taps respectively. Without loss of generality, $d = N_T - 1$, $m = N_T$ and $n = N_T - 1$ will be used as this choice of DFE structure parameters which is sufficient to guarantee the linear separability of the subsets of the channel states related to the different decisions [14]. Alternatively, the linear-combiner DFE can be expressed in state translated form [75]:

$$z(k) = \mathbf{a}^T \left(\mathbf{y}(k) - \mathbf{F}_2 \hat{\mathbf{x}}_b(k)\right) = \mathbf{a}^T \mathbf{y}'(k) \tag{3.6}$$

The translation of the original input (to the equaliser i.e. received) vector $\mathbf{y}(k)$ to transformed new input (to the equaliser) vector $\mathbf{y}'(k)$ is done as $\mathbf{y}'(k) = \mathbf{y}(k) - \mathbf{b}^T \hat{\mathbf{x}}_b(k)$, assuming that the feedback vector $\hat{\mathbf{x}}_b(k)$ is correct (shown in Figure 3.4). The matrix $\mathbf{F}_2$ is constructed by partitioning the channel impulse response matrix $\mathbf{F} = [\mathbf{F}_1\, \mathbf{F}_2]$, where:

$$\mathbf{F}_1 = \begin{bmatrix} h(0) & h(1) & \cdots & h(N_T-1) \\ 0 & h(0) & \ddots & \vdots \\ \vdots & \ddots & \ddots & h(1) \\ 0 & \cdots & 0 & h(0) \end{bmatrix}$$

$$\mathbf{F}_2 = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ h(N_T-1) & 0 & \ddots & \vdots \\ h(N_T-2) & h(N_T-1) & \ddots & 0 \\ \vdots & \ddots & \ddots & 0 \\ h(1) & \cdots & h(N_T-2) & h(N_T-1) \end{bmatrix}$$

The above translation performed on $\mathbf{y}(k)$ removes the contribution of past detected symbols $\hat{\mathbf{x}}_b(k)$, which essentially reduces the requirement of having multiple conditional decision function for each feedback pattern [76]. By performing the translation of eq. (3.6), the linear

combiner DFE is reduced to an equivalent linear equaliser 'without decision feedback':

$$f'(\mathbf{y}'(k)) = \mathbf{a}^T\mathbf{y}'(k) \tag{3.7}$$

The decision boundary of this equivalent linear equaliser consists of $C-1$ hyperplanes defined by: $\mathbf{y}' : \mathbf{a}^T\mathbf{y}' = 2i - C$, $1 \leq i \leq C-1$. In particular, for $C = 2$, the decision boundary, $\mathbf{y}' : \mathbf{a}^T\mathbf{y}' = 0$, is a hyperplane passing through the origin of the $\mathbf{y}'(k)$-space. It is shown, in [14], that in the state translation the channel states remain separable despite translation. The states can be made separable by applying a simple initial condition [14]. The performance of the state translated linear combiner DFE is shown to be better than conventional MMSE DFE, however performance depends on the accuracy of the built-in channel estimator [14].

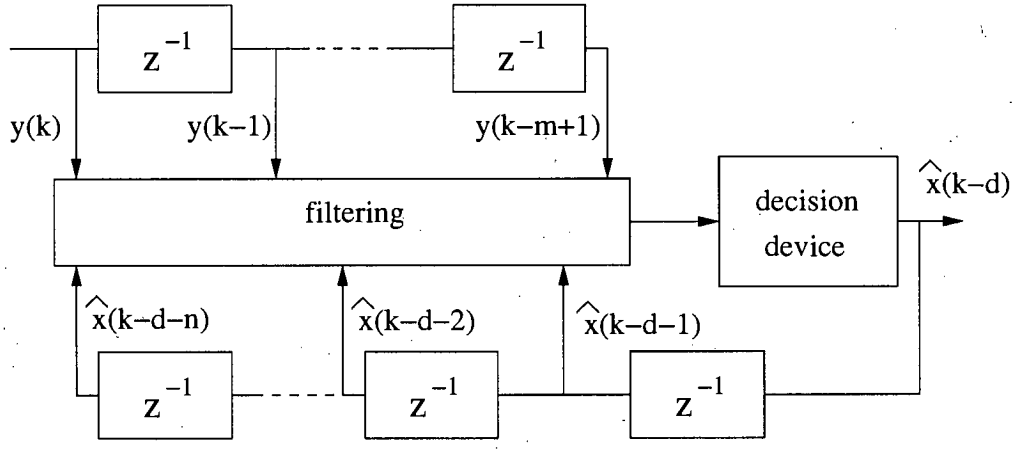


Figure 3.3: *Generic decision feedback equaliser*

The Wiener or MMSE solution is often said to provide the optimal $\mathbf{a}$ and $\mathbf{b}$. It is however optimal only with respect to the mean square error criterion. Obviously, there must exist a solution $\mathbf{a}_{opt}$ which achieves the best equalisation performance for the structure of eq. (3.7). The $\mathbf{a}_{opt}$ is refered to as the MBER solution of the linear-combiner DFE. The MMSE linear-combiner DFE is generally not this MBER solution. A natural question is how different the MMSE and MBER solutions can be. The difference in performance of MMSE and MBER solutions for Gaussian distributed noise is demonstrated in [14].

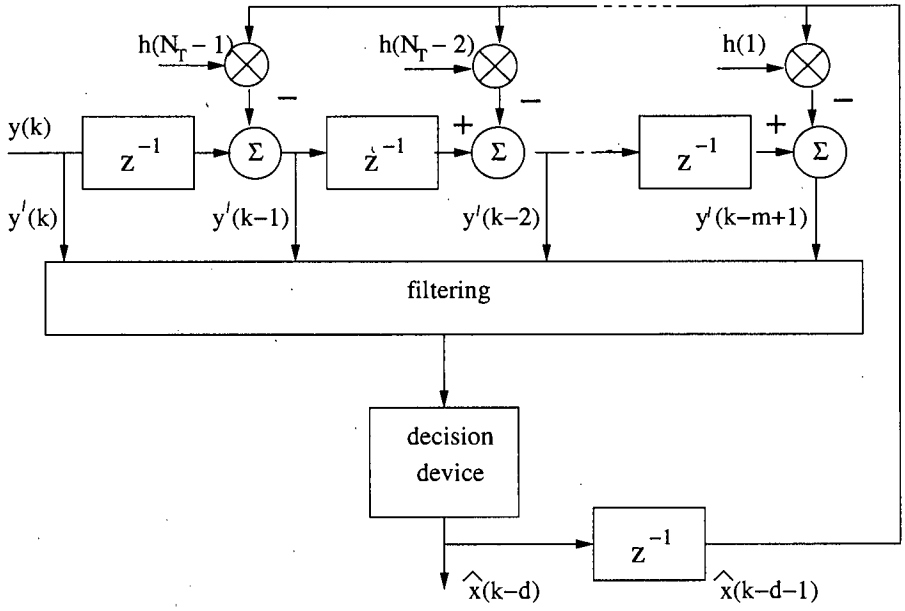


Figure 3.4: *Translated decision feedback equaliser*

3.3 Minimum bit error rate equalisation

It is obvious that the MBER and MMSE solutions are different as discussed in [77]. In this section the MBER criterion for a general DFE structure is described first. The calculation of MMSE solution is not possible for α-stable noise because of infinite variance. However, by introducing a practical design constraint of a limited dynamic range the Wiener solution (the conventional way) is estimated. For clarity we describe it as the WSL.

3.3.1 MBER criterion

The bit error rate (BER) observed at the output of the equaliser is dependent on the distribution of the decision variable $z(k)$ which in turn is a function of the equaliser tap weights. To be more specific, the probability of error, P_E, is:

$$P_E = P(\text{sgn}(x(k-d))z(k) < 0)$$

The sign adjusted decision variable $z_s(k) = \text{sgn}(x(k-d))z(k)$ is drawn from a mixture process. From the definition of $z(k)$,

$$\begin{aligned} z_s(k) &= \text{sgn}(x(k-d))(\mathbf{a}^T\mathbf{F}\mathbf{x}(k) - \mathbf{b}^T\hat{\mathbf{x}}_b(k)) + \text{sgn}(x(k-d))\mathbf{a}^T\mathbf{w}(k) \\ &= \text{sgn}(x(k-d))z'(k) + w'(k) \end{aligned} \tag{3.8}$$

where $\mathbf{w}(k) = [w(k)\, w(k-1) \ldots w(k-d-n)]^T$ is the vector of noise samples; $\mathbf{x}(k) = [x(k)\, x(k-1) \ldots x(k-d-N_T)]^T$ is the vector of transmitted symbols. The first term on the right hand side of eq. (3.8), $\text{sgn}(x(k-d))z'(k)$, is the noise-free sign-adjusted equaliser output and is a member of a finite set with N_z elements. These N_z elements forms the noiseless channel states i.e. the local means, of the mixture $z_s(k)$. Without noise the combination of the channel and DFE is a finite state machine whose state is completely defined by the vector $\mathbf{x}(k)$. Thus if $\mathbf{x}(k) \in \{\mathbf{x}_1 \cdots \mathbf{x}_i \cdots \mathbf{x}_{N_z}\}$, then $\mathbf{x}_i$ represent one such state of possible N_z states. Since $z'(k)$ (from eq. (3.8)) and $\mathbf{y}(k)$ (from eq. (3.5)) are functions of input signal, the vector $\mathbf{x}_i$ uniquely defines the i^{th} state of $z'(k)$, $\mathbf{y}(k)$, $x(k-d)$ and $\hat{\mathbf{x}}_b(k)$ - label these z_i, $\mathbf{y}_i$, x_i and $\hat{\mathbf{x}}_{bi}$ respectively. Note that while $\mathbf{x}(k)$ has N_z states, $x(k-d)$ has 2 possible values (2-PAM). However since $x(k-d)$ is a component of the vector $\mathbf{x}(k)$, the state of $\mathbf{x}(k)$ uniquely defines the value of $x(k-d)$. The second term $w'(k)$ is a zero mean α-stable white noise process with dispersion $\gamma(\sum_{j=1}^{m} |a_j|^\alpha)^{\frac{1}{\alpha}}$ and characteristic exponent α - defining the distribution about the local means.

3.3.2 Wiener solution with limiter

In an α-stable noise environment with $\alpha < 2$ the variance of the noise is infinite [71] making the use of the traditional Wiener solution theoretically meaningless. Nevertheless, all receivers in practice have a finite input dynamic range. This is achieved by using the structure as shown in Figure 3.2. The limiter at the front end of the receiver is assumed to be an ideal saturation device, with transfer function

$$L(x,G) = \begin{cases} x & : \ |x| \leq G \\ \text{sgn}(x)G & : \ \text{elsewhere} \end{cases}$$

G being the saturation point of the limiter. The saturation limit level G is kept at a reasonable distance from noiseless channel states to preserve the noise structure and not limit (clip) the noiseless channel state. Provided $G > \max(\bar{y}(k))$, the received signal at the output of the

limiter, $y_L(k)$, is the sum of the noise-free channel output $\bar{y}(k)$ and what is termed a *truncated* $\alpha-$stable noise process, $w_L(k)$: $\forall k$. The pdf of this *truncated* $\alpha-$stable noise process is given by:

$$f_\alpha(s, G_1, G_2) = f_\alpha(s)\mathbb{I}_{[G_1,G_2]}(s, G_1, G_2) + I_l(-G_1)\delta(s + G_1) + I_r(G_2)\delta(s - G_2) \quad (3.9)$$

where

$$\mathbb{I}_{[G_1,G_2]}(s) = \begin{cases} 1 & : \quad -G_1 \leq s \leq G_2 \\ 0 & : \quad \text{elsewhere} \end{cases}$$

$$I_l(-G_1) = \int_{-\infty}^{G_1} f_\alpha(s)ds, I_r(G_2) = \int_{G_2}^{\infty} f_\alpha(s)ds$$

where $f_\alpha(s)$ represents the α-stable distribution. Since the limiter is assumed to be symmetrical $G_1 = G_2 = G$ is considered. The pdf of the channel states (assuming equi-probable) is a delta function at the channel centres.

$$f_{\overline{c_i}}(s) = \frac{1}{N_{sc}} \sum_{i=1}^{N_{sc}} \delta(s - \overline{c_i}) \quad (3.10)$$

Since the truncated α-stable noise process of eq. (3.9) and the noise-free scalar channel states of eq. (3.10) are independent, the combined pdf is given by:

$$f_{y_L}(s) = \frac{1}{N_{sc}} \sum_{i=1}^{N_{sc}} f_{y_L|\overline{c_i}}(s) = \frac{1}{N_{sc}} \sum_{i=1}^{N_{sc}} f_\alpha(s - \overline{c_i}, -G - \overline{c_i}, G - \overline{c_i}) \quad (3.11)$$

where $N_{sc} = 2^{N_T}$ is the number of the scalar centres $\overline{c_i}$ of the channel, i.e., $\overline{c_i} = \mathbf{h}^{\mathbf{T}}.\mathbf{x}_{\mathbf{ch_i}} (i = 1, 2, ..., N_{sc})$, where $\mathbf{h} = [h(0)...h(N_T - 1)]^T$ and $\mathbf{x}_{\mathbf{ch_i}} = [x(k)...x(k - N_T + 1)]^T$ are all the possible combinations for the channel input vector. This pdf is the same as that observed at the output of the receiver, which confirms independence. The limiter "$L(x, G)$" truncates the pdf of the received signal and its tails are concentrated at the points $\pm G$, where they appear as *Dirac* impulses $\delta(s)$. The noise variance can be calculated theoretically from [9], with knowledge of α, limiting level G and noiseless channel states $\overline{c_i}$.

From classical Wiener filter theory [16], the WSL is $\mathbf{a}_o = \mathbf{R}^{-1}\mathbf{p}$, where $\mathbf{a}_o$ is the optimum tap-weight vector, $\mathbf{R} = \mathbb{E}\{\mathbf{y_L}\mathbf{y_L}^T\}$ is the input autocorrelation matrix, $\mathbf{p} = \mathbb{E}\{\mathbf{y_L}x\}$ is the cross-correlation vector and $\mathbf{y_L} = [y_L(k)\, y_L(k-1) \ldots y_L(k-m+1)]^T$. The autocorrelation matrix is simply the sum of two autocorrelation matrices: (i) the autocorrelation matrix associated with

the noise free channel output; (ii) a scaled identity matrix. The scale factor is the variance of the truncated α-stable process and thus the scale factor is $\int s^2 f_\alpha(s, G)ds$. The cross-correlation matrix is simply the cross-correlation of the noise free channel output with the target symbol. Because the variance of the truncated α-stable noise process is a function of both the parameter α and the limiter value G, the WSL will be as well.

Thus the WSL is formulated after the limiter using the independence property, which was not obvious from [9].

3.4 Stochastic gradient adaptive equalisers

In this section the problem of minimising BER in an α-stable noise environment is addressed directly and a stochastic gradient algorithm for the task is derived. As the development is in terms of probability of error rather than mean squared error the requirement for a limiter is removed.

Consider the noise density function $f(x)$ associated with the zero mean random variable x. The density function is symmetrical and normalised such that the variance or dispersion is unity. The associated distribution function is $P(x)$. The "generalised" error function is $Q(x) = 1 - P(x)$ and its derivative is $Q'(x) = -f(x)$. The probability of error at the output of a linear or state translation equaliser with N noise free states as a function of the weight m-vector $\mathbf{a}$ is:

$$P_E(\mathbf{a}) = \frac{1}{N}\sum_{i=1}^{N} Q(g_i(\mathbf{a}))$$

where $g_i(\mathbf{a})$ is the signed decision variable associated to the i^{th} state, normalised by the "strength" of the noise. In the Gaussian case [15]:

$$g_i(\mathbf{a}) = \frac{\mathbf{a}^T \mathbf{y}_i x_i}{||\mathbf{a}||\sigma} \tag{3.12}$$

where $\mathbf{y}_i$ is the i^{th} noise free received vector; the Euclidean norm is $||\mathbf{a}|| = (\sum_{j=1}^{m} |a_j|^2)^{\frac{1}{2}}$; x_i is the transmitted symbol associated with that vector; σ^2 is the noise variance. In the α-stable case:

$$g_i(\mathbf{a}) = \frac{\mathbf{a}^T \mathbf{y}_i x_i}{||\mathbf{a}||_\alpha \gamma^{\frac{1}{\alpha}}} \tag{3.13}$$

where the "α-norm" is defined as: $||\mathbf{a}||_\alpha = (\sum_{j=1}^{m} |a_j|^\alpha)^{\frac{1}{\alpha}}$ For adaptive filters, derivatives of the form $\partial P_E / \partial a_j : \forall j$ are required.

$$\begin{aligned}\frac{\partial P_E}{\partial a_j} &= \frac{1}{N}\sum_{i=1}^{N} Q'(g_i(\mathbf{a}))\frac{\partial g_i(\mathbf{a})}{\partial a_j} \\ &= -\frac{1}{N}\sum_{i=1}^{N} f(g_i(\mathbf{a}))\frac{\partial g_i(\mathbf{a})}{\partial a_j}\end{aligned}$$

In the Gaussian case the derivative of eq. (3.12) is given by:

$$\begin{aligned}\frac{\partial g_i(\mathbf{a})}{\partial a_j} &= \frac{\partial}{\partial a_j}\left(\frac{\mathbf{a}^T}{||\mathbf{a}||}\right)\frac{\mathbf{y}_i x_i}{\sigma} \\ &= \frac{1}{||\mathbf{a}||}\left(\mathbf{1}_j^T - \frac{\mathbf{a}^T a_j}{||\mathbf{a}||^2}\right)\frac{\mathbf{y}_i x_i}{\sigma}\end{aligned} \tag{3.14}$$

where $\mathbf{1}_j$ is an m-vector with all zero elements apart from the j^{th} entry which is unity. In the α-stable case the derivative of eq. (3.13) is taken:

$$\begin{aligned}\frac{\partial g_i(\mathbf{a})}{\partial a_j} &= \frac{\partial}{\partial a_j}\left(\frac{\mathbf{a}^T}{||\mathbf{a}||_\alpha}\right)\frac{\mathbf{y}_i x_i}{\gamma^{\frac{1}{\alpha}}} \\ &= \frac{1}{||\mathbf{a}||_\alpha}\left(\mathbf{1}_j^T - \frac{\mathbf{a}^T|a_j|^{\alpha-1}\mathrm{sgn}(a_j)}{||\mathbf{a}||_\alpha}\right)\frac{\mathbf{y}_i x_i}{\gamma^{\frac{1}{\alpha}}}\end{aligned} \tag{3.15}$$

The α-stable case being more general is used in the derivations from now on. Thus multipling out gives:

$$\frac{\partial g_i(\mathbf{a})}{\partial a_j} = \frac{1}{||\mathbf{a}||_\alpha}\left(y_{ij} - \frac{z_i|a_j|^{\alpha-1}\mathrm{sgn}(a_j)}{||\mathbf{a}||_\alpha}\right)\frac{x_i}{\gamma^{\frac{1}{\alpha}}}$$

where y_{ij} is the j^{th} element of $\mathbf{y}_i$ and $z_i = \mathbf{a}^T\mathbf{y}_i$ i.e. the equaliser output associated with the i^{th} noise free state. Collecting partial derivatives together to form a gradient vector we have:

$$\nabla P_E(\mathbf{a}) = -\frac{1}{N}\sum_{i=1}^{N} f(g_i(\mathbf{a}))\frac{1}{||\mathbf{a}||_\alpha}\left(\mathbf{y}_i - \frac{z_i\langle\mathbf{a}\rangle_\alpha}{||\mathbf{a}||_\alpha}\right)\frac{x_i}{\gamma^{\frac{1}{\alpha}}}$$

where $\langle\mathbf{a}\rangle_\alpha$ is an m-vector with j^{th} element equal to $|a_j|^{\alpha-1}\mathrm{sgn}(a_j)$. Since the norm of the weight vector does not affect P_E in the binary signalling case it can be set to unity at each iteration thus:

$$\nabla P_E(\mathbf{a}) = -\frac{1}{N\gamma^{\frac{1}{\alpha}}}\sum_{i=1}^{N} f\left(\frac{z_i x_i}{\gamma^{\frac{1}{\alpha}}}\right)\left(\mathbf{y}_i - \langle\mathbf{a}\rangle_\alpha z_i\right)x_i$$

The key to developing the LMS algorithm from its related steepest descent algorithm is to

replace the ensemble average of the gradient with a single point estimate of the gradient. The same concept can also be employed to develop an LMS-style update algorithm to train a MBER DFE [15]. Thus the final update equation is modified as an LMS-style update:

$$\mathbf{a}(k+1) = \mathbf{a}(k) + \mu f\left(\frac{z(k)x(k-d)}{\gamma^{\frac{1}{\alpha}}}\right)(\mathbf{y}(k) - \langle \mathbf{a}(k)\rangle_{\alpha} z(k))\frac{x(k-d)}{\gamma^{\frac{1}{\alpha}}} \tag{3.16}$$

$$\mathbf{b}(k+1) = \mathbf{b}(k) - \mu f\left(\frac{z(k)x(k-d)}{\gamma^{\frac{1}{\alpha}}}\right)\frac{x(k-d)}{\gamma^{\frac{1}{\alpha}}}\hat{\mathbf{x}}_b(k) \tag{3.17}$$

For a state-translated DFE, eq. (3.16) is modified by replacing $\mathbf{y}(k)$ with $\mathbf{y}'(k)$

$$\mathbf{a}(k+1) = \mathbf{a}(k) + \mu f\left(\frac{z(k)x(k-d)}{\gamma^{\frac{1}{\alpha}}}\right)(\mathbf{y}'(k) - \langle \mathbf{a}(k)\rangle_{\alpha} z(k))\frac{x(k-d)}{\gamma^{\frac{1}{\alpha}}} \tag{3.18}$$

where

$$\mathbf{y}'(k) = \mathbf{y}(k) - \mathbf{F}_2(k)\hat{\mathbf{x}}(k) \tag{3.19}$$

Eq. (3.17) is not used here, rather an L_p-norm (equivalent to α-norm discussed earlier) is considered to form an estimate of the impulse response vector $\mathbf{h} = [h(0)h(1)...h(N_T-1)]^T$:

$$\mathbf{h}_{k+1} = \mathbf{h}_k + \mu \left[\begin{array}{c} x(k-d) \\ \hat{\mathbf{x}_b} \end{array}\right] \left\| y(k-d) - \mathbf{h}^T(k) \left[\begin{array}{c} x(k-d) \\ \hat{\mathbf{x}_b} \end{array}\right] \right\|_{\alpha} \tag{3.20}$$

The equaliser tap weights are normalised after each update. The final decision, $\hat{x}(k-d)$, is made on the filter output $\mathbf{a}^T(k)\mathbf{y}'(k)$.

3.5 Simulation study

In this chapter, the SNR of the limited received signal $y_L(k)$ is used for performance evaluation in environments where the noise variance is infinite. By using the limiter the SNR is always finite and hence measurable. This is referred as the SNR *at the receiver*. Simulations were performed for anti-podal signalling ($C = 2$), assuming that the noise is Cauchy distributed i.e. $\alpha = 1$ and the limiter, at DFE front-end, is at ± 4 [9] to avoid being close to noiseless channel states at the transmitter output. Method developed by Chambers, Mallows and Stuck [78] is used in this thesis to generate random noise. The same procedure was used in [79] and [13].

The variance of the truncated α-stable process $w_L(k)$ is calculated as discussed in [9]. Figure 3.2 represents the receiver architecture considered in simulations.

As the performance of equalisers are highly dependent on the nature of the channel considered, two channels which have been well studied in the literature [13] were chosen to characterise the performance. These channels have impulse responses [0.3482 0.8704 0.3482] and [1.0 0.50 0.25]. The location of poles and zeros for both the channels is shown in Figure 3.5 and the frequency respose in Figure 3.6. As observed from the figures, the channel response of Figure 3.5-a) is more difficult to equalise than the channel of Figure 3.5-b). The presence of a zero outside the unit circle, as in Figure 3.5-a), results in the channel being non-invertible leading to difficulty for linear equaliser. As observed from the frequency response a high noise enhancement will result if an linear transversal equaliser was used since the gain required to equalise the channel [0.3482 0.8704 0.3482] at higher frequency tends to be much larger than for the second channel [1.0 0.5 0.25] as also seen from Figure 3.6. The DFE provides better equalisation than linear equaliser in such deep fade communication channels. The performance of the DFE equaliser can be improved by increasing the order of feed-forward and feed-back taps. However to have limited computational complexity, the DFE structure is chosen to be $d = 2$, $m = 3$ and $n = 2$.

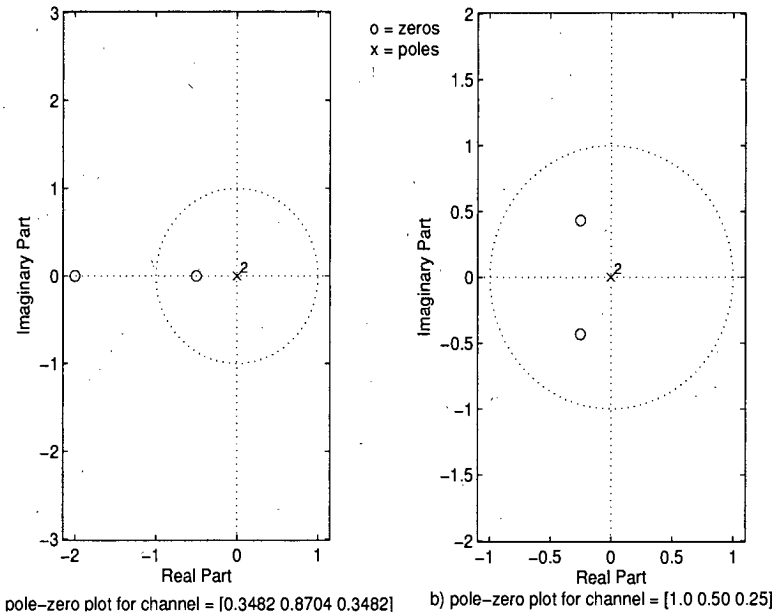


Figure 3.5: *Pole zero plots for the communication channels to be equalised*

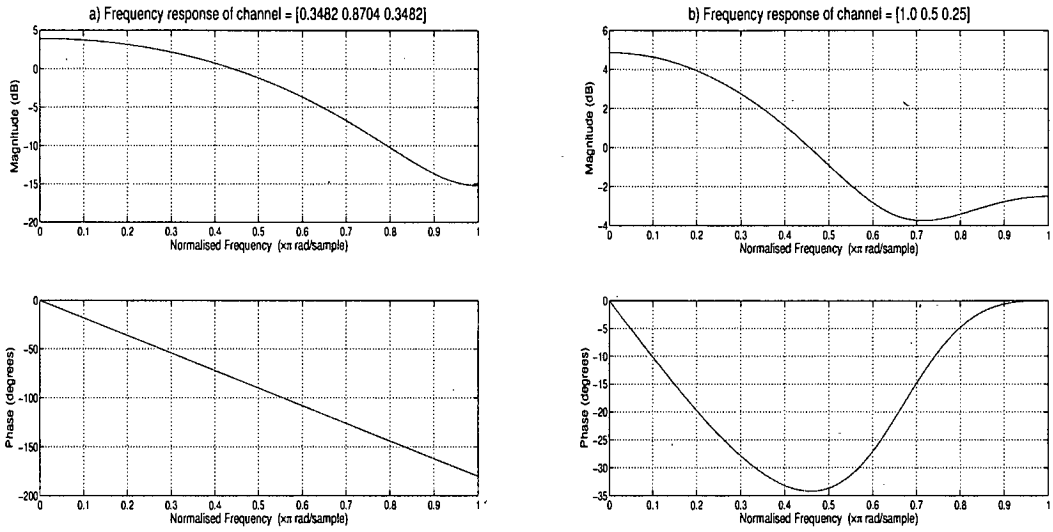


Figure 3.6: *Frequency response of the communication channels to be equalised*

The legends in Figure 3.7, Figure 3.8, Figure 3.9 and Figure 3.10 depict: a) 'LMS' refers to a conventional LMS algorithm for both the feedforward and feedback taps of a conventional DFE, b) 'LBER-Gaussian' refers to a LBER algorithm for adapting both feedforward and feedback equaliser taps of a conventional DFE assuming that the noise is Gaussian [15] (using eq. (3.16)-(3.17) with density $f(.)$ being Gaussian distributed) , c) 'LBER-Cauchy' refers to adapting both the feedforward and feedback taps of a conventional DFE assuming Cauchy distributed noise using eq. (3.16)-(3.17) with density $f(.)$ being Cauchy distributed, d) 'state trans-Gaussian' refers to the same adaptive algorithm as (b) but with state translated design [14] (update eq. (3.18)-(3.20)), e) 'state trans-Cauchy' refers to the same adaptive algorithm as (c) but with a state translated design updated as in eq. (3.18)-(3.20), f) 'modified Wiener' represents WSL calculated after the limiter using $y_L(k)$ as discussed in section-3.3.2. A total of 10^5 samples was used to generate the convergence and performance plots using Matlab. In order to make a fair comparison of the relative performance of the algorithms the adaptation constant μ is fixed as $\frac{1}{6(m+n)}$ [15] for all the adaptive algorithms compared in this chapter. A large sample size and ensemble for simulations was taken to reach conclusions because of the impulsive (high variations in input signal amplitude) nature of α-stable noise.

An ensemble of 100-runs was taken to generate convergence plots as shown in Figure 3.7 and Figure 3.9 at a SNR of 7.9 dB's. As can be observed the convergence behaviour of the LMS is

unstable. This can be attributed to the fact that the LMS is dependent on the magnitude of the instantaneous error, which varies a lot in an impulsive noise environments. Algorithms designed to minimise BER in a Gaussian noise environment converge more slowly than those specifically designed for the Cauchy noise environment. It is safe to conclude that the state translated design for Cauchy noise has faster and more stable convergence than the other algorithms.

To observe the BER performance of these algorithms an ensemble of 1000-runs was taken. The equalisers were trained using the first 1000-samples of a particular run after which training was inhibited and the BER for that run measured. The final BER estimate was obtained by averaging over 1000 such runs in the ensemble. Figure 3.8 and Figure 3.10 summarise the results for the two channels used. It is observed at a BER of 5×10^{-3} approximately 5 dB's is gained by using a minimum-BER criterion instead of an LMS algorithm. Again the Gaussian noise based LBER algorithms perform well with respect to Cauchy noise based LBER algorithms which are tailored to the particular environment. The state translated Cauchy noise based LBER DFE performs better than the other algorithms as is apparent from both Figure 3.8 and Figure 3.10. It is also interesting to observe that this MBER algorithm performs better than the WSL.

While the WSL provides an optimal solution in the MSE sense, however it does not minimise the BER. However the LMS algorithm, which would normally find the MSE solution, fails to converge to this solution in this environment. The LBER algorithms, by their nature, seek the desired optimum MBER solution. LBER algorithms have been demonstrated to find the optimum BER solution with a computational complexity similar to that of the LMS, as obvious from eq. (3.16)-(3.17) [15]. From the simulations it is observed that the state-translated DFE for Cauchy distributed noise has better convergence and BER performance than the other algorithms considered. LBER algorithms based on Gaussian noise [15] assumptions have also been demonstrated to perform well in α-stable noise environments. The close performance of Gaussian noise based algorithms [15] to the Cauchy noise based ones can be attributed to the fact that the Gaussian mixtures may model α-stable distributions [80] [81] and to the presence of limiter at the DFE front-end which essentially limits the heavy tails of α-stable noise distribution.

3.6 Conclusions

A minimum bit error rate adaptive algorithm for impulsive noise modelled as α-stable noise has been proposed in this chapter. By introducing a limiter at the receiver front-end both SNR and

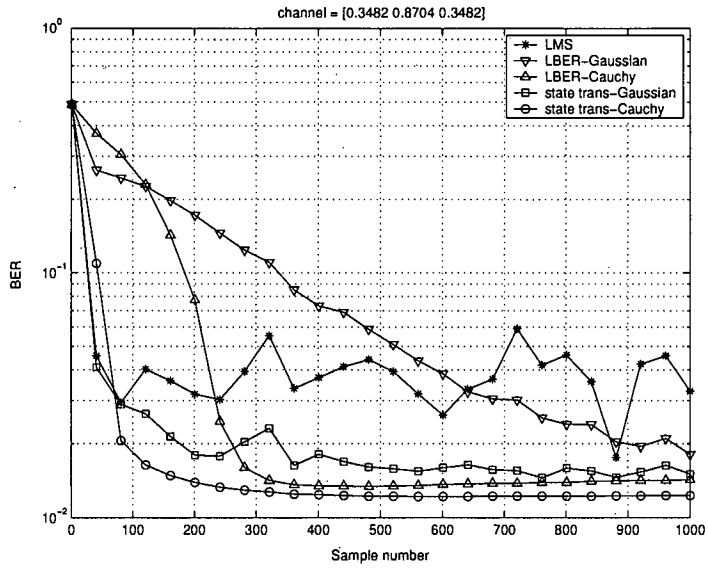


Figure 3.7: *Convergence plot for Cauchy (α = 1) distributed noise for channel = [0.3482 0.8704 0.3482]*

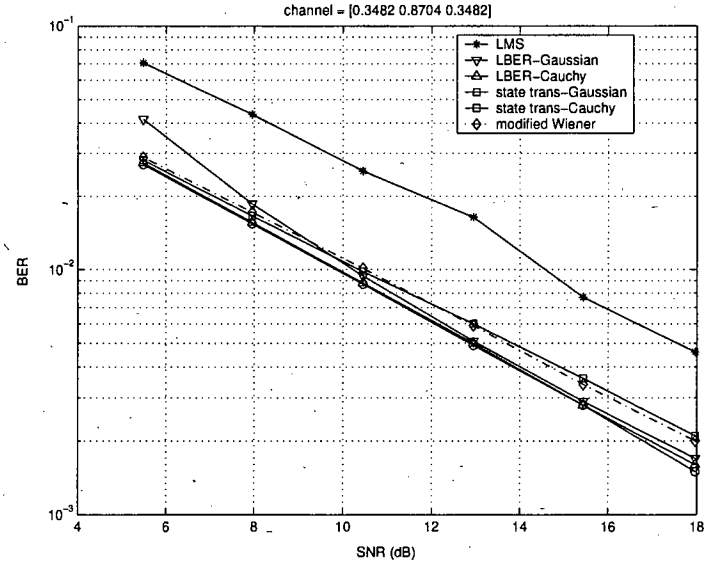


Figure 3.8: *Performance plot for Cauchy (α = 1) distributed noise for channel = [0.3482 0.8704 0.3482] with SNR calculated after the limiter*

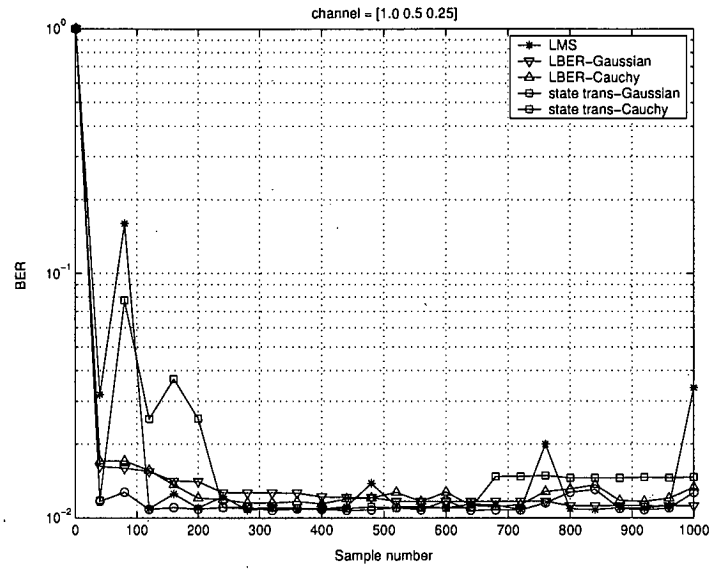


Figure 3.9: *Convergence plot for Cauchy (α = 1) distributed noise for channel = [1.0 0.5 0.25]*

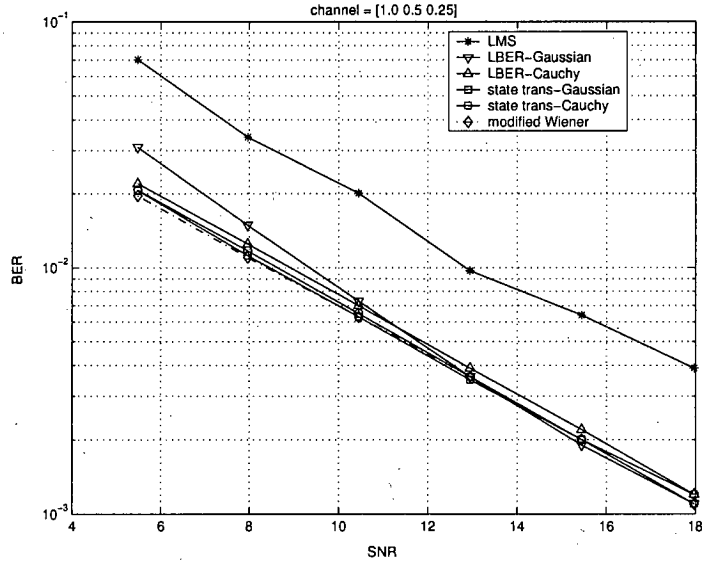


Figure 3.10: *Performance plot for Cauchy (α = 1) distributed noise for channel = [1.0 0.5 0.25] with SNR calculated after the limiter*

Wiener solution can be calculated theoretically and by simulations. It is shown that for minimum bit error design, the adaptation is a function of the noise density function. The comparison between various adaptive algorithms working in identical channel, noise and DFE structure has been drawn. The LBER-Cauchy and the state trans-Cauchy has faster convergence than the other adaptive algorithms in Cauchy noise environments, which is a special form of α-stable noise. Extensive simulations strongly suggest that the state-translated design for the α-stable noise has better convergence and BER performance than the other algorithms, as the translation in space leads to linear seperability and reduction in states [14]. It is also interesting to observe that the adaptive algorithms based on a Gaussian noise assumption despite slow convergence in impulsive noise environments perform closer to those designed with Cauchy noise assumption. Lastly, as expected, the LMS algorithm performs poorer that the other algorithms in α-stable noise environments. Observations from Figure 3.8 and Figure 3.10 suggests MBER algorithms' superior performance with respect to the WSL solution. It is worth mentioning that the overall performance of the different equalisers used in this chapter can be improved by increasing the equaliser taps order.

Chapter 4
Non-parametric maximum likelihood channel estimator in the presence of uncorrelated non-Gaussian noise

The Gaussian random process has always been the dominant noise model in communications and signal processing, mainly because of the central limit theorem and the relative ease of analytic manipulation. Unfortunately, in some communication channels, the observation noise exhibits non-Gaussian characteristics either due to impulsive noise [70] or co-channel interference. Impulsive noise is more likely to exhibit sharp spikes or occasional bursts of outlying observations than one would expect from normally distributed signals as discussed in details in the previous chapter. Co-channel interference is a dominant feature of modern radio communications systems in that virtually no radio link or system is alone in its allocated frequency band. Other radio transmitters, near and far, constantly cause interference [24]. The combination of CCI and thermal noise leads to observation noise that is drawn from a Gaussian mixture [82]. Thus the non-Gaussianity of the observation noise may be due to the presence of impulsive noise or CCI or unknown mixtures of both and it would be desirable to have channel estimations techniques that address this non-Gaussian nature without requiring explicit knowledge of which form is present or their relative intensities.

Channel estimation forms an integral part of a communication receiver [82]. The channel estimation in communication systems can be done in three ways: 1) blind, 2) semi-blind, and 3) training based. In this chapter a training based channel estimator is addressed, where the channel is estimated over a block of data (similar to in GSM) [82]. As discussed above the observed noise at the receiver may not be Gaussian. This degrades the performance of a LS based channel estimators. Various statistical techniques like EM and method of moments based channel estimators have been proposed for communication systems in [83] [44]. However they usually limit (approximate) the interference as Gaussian distributed [84] [44], which may not be the case if there are only a few strong interferers. Other techniques based on joint detection/estimation like in [63] and [64] which work on interference cancellation make certain

assumptions about the interferer, which cannot be generalised in practice. Such joint detection and estimation is out of scope of this chapter. In this chapter no assumption on the distribution (and number) of the interference is made, which makes the proposed technique robust to the nature of the observed mixture distribution.

From [82] [80] [10] [85] it is clear that various types of observation noise encountered in communications systems can be modelled as a Gaussian mixture. In this chapter two main classes of finite Gaussian mixture distribution are considered: a) uni-modal distribution where zero-mean processes of differing variances are mixed; b) multi-modal distribution where non-zero-mean processes with the same variance are mixed. The uni-modal form lends itself naturally to impulsive noise [10] [86] [85] while the multi-modal form is appropriate for CCI [82] [87]. The most popular choice for ML system identification in a Gaussian mixture observation noise environment is to use the EM algorithm [47] [45] [88]. The EM and its variant space-alternating generalised EM algorithm (SAGE) [89] is used in [90] [10] [86] [43] and many more for parameter estimation and detection. Application of the EM algorithm to the problem usually requires *a priori* assumptions about the specific functional form of the pdf of the observation noise. The unknown parameters of the mixture (e.g. the means and variances of the mixtures) are grouped with the channel impulse response to form the vector to be estimated. For example in [10] [86] the specific case of impulsive noise modelled as a zero-mean Gaussian mixture is considered. In [43] it is shown that for deterministic channel estimation the problem reduces to RLS.

In this chapter a generic non-parametric approach to ML channel estimation that is capable of addressing both the impulsive noise environment where the uni-modal form is appropriate and the CCI environment where the multi-modal form is appropriate is presented. The technique can be applied without making any *a priori* assumptions about the number and nature of the means and variances of Gaussian mixture. In the impulsive noise environment this technique is shown to have similar performance to existing EM based algorithms.

The key to this non-parametric approach is the use of kernel density estimation to characterise the observation noise directly from the data. The theory developed in [91] and [92] to estimate the communication channel impulse response in a non-Gaussian noise environment is estimated. The relationship between the minimum error entropy (MEE) algorithm and the non-parametric ML (NPML) is also highlighted here. Improvements in the mean square error (MSE) performance with respect to the traditional LS channel estimate are examined and a comparison with EM is also considered.

The Cramér Rao bound (CRB) defines the lower bound on the channel estimator's performance. There is rich literature available on CRB formulations for both Gaussian and non-Gaussian noise scenarios in [42] [85] [93] [94] to name a few. The CRB for channel estimation in the two (uni-modal and multi-modal) Gaussian mixture case is formulated in this chapter. First an analytical expression for the CRB is developed and then from simulations it is observed that NPML algorithm is closer to achieving the CRB than the LS technique.

This chapter is organised as follows. First, the problem statement is formulated in section-4.1 for a general communication system. A short discussion on kernel density estimators is provided in section-4.2. The non-parametric maximum likelihood algorithm using kernel density estimators is discussed in section-4.3. The iterative MEE channel estimator is derived and compared with NPML in section-4.4. The theoretical lower bounds on the channel estimator in non-Gaussian noise modelled as Gaussian mixtures is formulated in section-4.5. In section-4.6 simulation results are presented. Conclusions based on analysis and simulation are drawn at the end.

4.1 Formulation of the problem

The discrete-time model in the low-pass equivalent form of the communication system channel estimator is shown in Figure 4.1. Without loss of generality, the input signal is assumed to be a randomly generated binary anti-podal PAM signal, so that the transmitted symbols are $\mathbf{x} \epsilon \{\pm 1\}$. Here $y(k)$ represents the received signal and $w(k)$ is the additive observation noise. The model is simplified by assuming that the channel is of order $N_T - 1$ i.e.: $\mathbf{h} = [h(0), h(1), ..., h(N_T - 1)]$.

More precisely, the received signal $y(k)$ sampled once per symbol can be written as:

$$y(k) = \sum_{n=0}^{N_T-1} h(n)x(k-n) + w(k) \tag{4.1}$$

The problem is to estimate the channel coefficients from the received signal, assuming that the input signal (in a supervised training mode) and the channel (tap) length is known at the receiver. Thus the problem reduces to the well known problem of system identification. In this chapter a block based channel estimator with training symbols, similar to that of the GSM system [82] is considered. As usually assumed for slow fading channels in GSM environments,

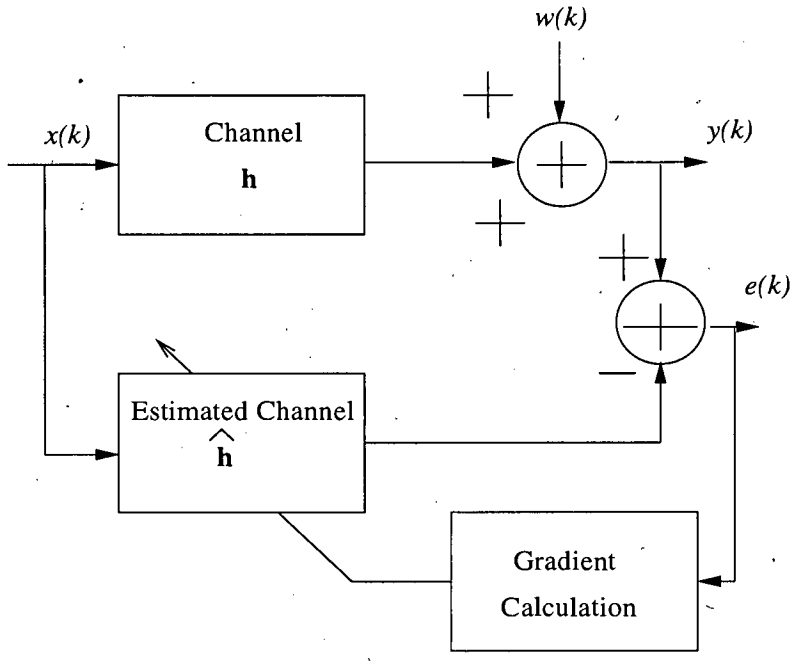


Figure 4.1: *A typical communication systems channel estimator*

the channel is assumed to be fixed for the burst (block) duration.

There are various algorithms based on different criteria to estimate the channel taps. The LS solution is the optimum solution for the Gaussian noise environments where it is equivalent to an ML estimate [39]. However, here it is assumed that the observation noise is non-Gaussian and thus LS does not provide the ML solution. The pdf of the observation noise is assumed to be unknown at the receiver. Thus the channel estimator in this chapter performs two tasks: (i) estimation of the impulse response itself; (ii) estimating the pdf of the observation noise at the receiver to construct the likelihood.

The pdf of the additive noise is modelled by a mixture of finite Gaussian distributions. The three justifications for using a Gaussian mixture model are (1) the set of Gaussian mixture distributions include as approximation to Middleton's canonical class A model [95], (2) Fan's theorem [96] indicates that Gaussian mixture distributions can approximate a large class of pdfs, and (3) the Gaussian mixtures distribution naturally includes the Gaussian thermal noise that is present in communication systems [81]. In addition to these justifications, a generalised

Gaussian mixture pdf model is used:

$$f_W(w) = \sum_{l=1}^{N_M} \frac{\lambda_l}{\sqrt{2\pi\sigma_l^2}} \exp\left(-\frac{(w-\hat{w}_l)^2}{2\sigma_l^2}\right) \tag{4.2}$$

where w is the noise sequence, λ_l represents the probability that w is chosen from the l^{th} term in the mixture pdf, with $\sum_{l=1}^{N_M} \lambda_l = 1$ for N_M number of mixtures. We will verify the robustness of the algorithm by assuming $\forall\, l; \hat{w}_l = 0$ for uni-modal noise and non-zero for multi-modal noise distribution. It is interesting to note from [82] that the co-channel interference can be modelled as a Gaussian mixture of non-zero mean processes with identical variance ($\sigma_1^2 = \sigma_2^2 ... = \sigma_{N_M}^2$). This will be discussed in detail later in this chapter.

4.2 Kernel density estimation

Non-paramertric density estimaton is a classical topic in statitics, where the two most common techniques are histogram and kernel density methods. A histogram is the simplest non-parametric density estimator and the one that is most frequently encountered. To construct a histogram, the sampled data is divided into the intervals covered by the data values and then into equal sub-intervals, known as 'bins'. Every time, a data value falls into a particular sub-interval, then a block of size equal to the binwidth is placed on the top of it. The disadvantages with histogram plot are that, it does not provide a smooth estimate of the density, it is sensitivity to the end points of bins, and width of the bins. The problem of smoothing and end points in histogram can be overcome by using a smooth kernel placed on the observed data point. The Figures 4.2 and 4.3 show estimated density plot for hypothetical sampled data $[-0.90\ -0.70\ -0.50\ -0.450\ -0.35\ 0.25\ 0.35\ 0.50\ 0.70]$ from histogram and kernel method respectively, where a Gaussian kernel is used to estimate the density. Thus to estimate the pdf at the receiver the kernel density estimator technique is used in the thesis. Parzen's window technique or kernel density estimation assumes that the probability density is a smoothed version of the empirical samples. Thus, for a collection of M measured data samples $\{y(j)\}_{j=1}^{M}$ the estimate $\hat{f}(y)$ (where $\hat{f}(y)$ is a random variable) of the underlying pdf $f(y)$ is the average of radial kernel functions centered on the M measured data samples:

$$\hat{f}(y) = \frac{1}{M}\sum_{j=1}^{M} K(y - y(j)) \tag{4.3}$$

$K(.)$ is the Gaussian kernel (Parzen kernel) [82] defined as:

$$K(y) = \mathcal{N}(0,\sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(\frac{-y^2}{2\sigma^2}\right) \tag{4.4}$$

with variance (or kernel width) defined as σ. Other choices of kernel like *Epanechnikov kernel* are possible. The *Epanechnikov kernel* is defined as [8]:

$$K_E(y) = \begin{cases} \frac{2}{\pi\sigma^2}(1 - \frac{y^2}{\sigma^2}) & : \quad \frac{y^2}{\sigma^2} < 1 \\ 0 & : \quad \text{elsewhere} \end{cases}$$

It can be shown that under the right choice of kernel function $\hat{f}(y)$ will converge to the true density $f(y)$ as $M \to \infty$ [97].

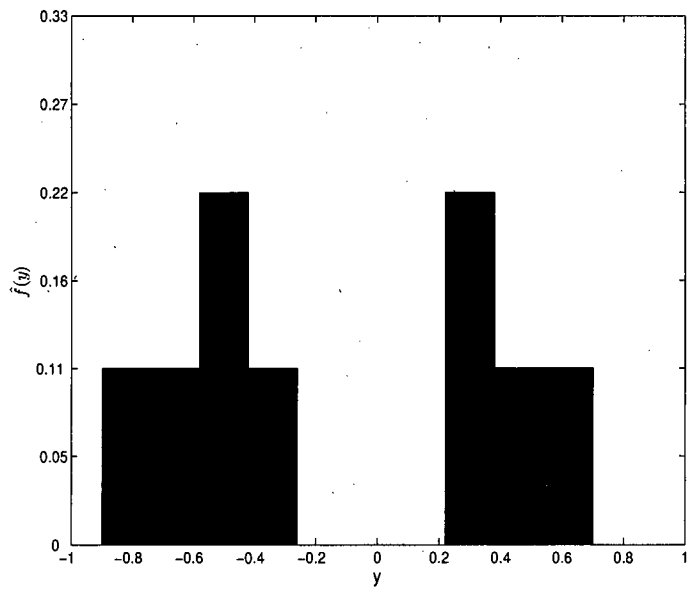


Figure 4.2: *Histogram density estimate for [-0.90,-0.70,-0.50,-0.450,-0.35,0.25,0.35,0.50,0.70]*

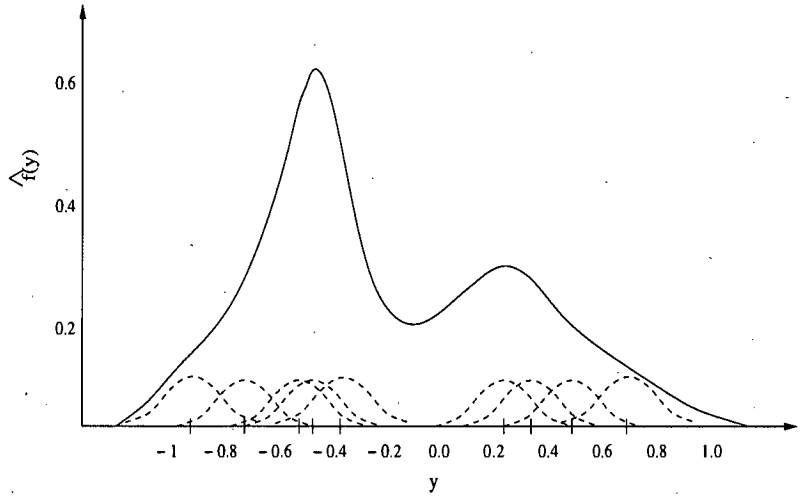


Figure 4.3: *Kernel density estimate for [-0.90,-0.70,-0.50,-0.450,-0.35,0.25,0.35,0.50,0.70]*

4.3 Non-parametric maximum-likelihood (NPML) channel estimation

For the communication system represented by eq. (4.1) the ML estimate forms the optimal estimator for the channel. The log-likelihood function can be represented as:

$$\mathcal{L}(\mathbf{h} \mid \mathbf{y}) = \log f(\mathbf{y} \mid \mathbf{h}) = \sum_{i=1}^{M} \log f_w(e(i)) \tag{4.5}$$

where $f_w(.)$ is the pdf of the independently identically distributed (i.i.d.) observation noise w (eq. (4.1)), $e(j) = y(j) - \sum_{n=0}^{N_T-1} x(j-n)h(n)$ is the estimation error and M is the number of data points in a block. The optimal channel estimate is the solution to:

$$\nabla_{\mathbf{h}} \mathcal{L}(\mathbf{h} \mid \mathbf{y}) = \frac{\partial \mathcal{L}(\mathbf{h} \mid \mathbf{y})}{\partial \mathbf{h}} = 0 \tag{4.6}$$

If the noise was Gaussian then the solution to the above equation leads to the LS estimate. However, in communication systems where the noise is non-Gaussian closed form equation for $f_w(.)$ do not exist in general. Even if $f_w(.)$ was known and was genuinely a Gaussian mixture, no closed form analytic solution for eq. (4.6) exists. Hence to find the solution, an iterative gradient ascent based algorithm is proposed as follows. The gradient here is the first derivative

of the log-likelihood function with a constant multiplier. The update equation is:

$$\hat{\mathbf{h}}_k = \hat{\mathbf{h}}_{k-1} + \mu(k)\,\nabla_{\mathbf{h}}\mathcal{L}(\mathbf{h} \mid \mathbf{y})\,|_{\mathbf{h}=\hat{\mathbf{h}}_{k-1}} \tag{4.7}$$

$$= \hat{\mathbf{h}}_{k-1} + \mu(k)\,\frac{\partial\{\sum_{i=1}^{M}\log f_w(e(i))\}}{\partial \mathbf{h}}\bigg|_{\mathbf{h}=\hat{\mathbf{h}}_{k-1}} \tag{4.8}$$

where $\mu(k)$ is the adaptation step-size. Since the channel estimator is assumed to have no *a priori* knowledge of the pdf $f_w(.)$, the unknown pdf is estimated using the kernel density estimator eq. (4.3) with Gaussian kernels as shown below. As the kernel estimators are known to be effective in density estimation over short data record, this technique is used over the available data (error) record to estimate the unknown density. Using the kernel density estimator [92] the estimated pdf is written as:

$$\hat{f}_w(e) = \frac{1}{M}\sum_{j=1}^{M} K(e - e(j)) \tag{4.9}$$

Thus the estimated log-likelihood function becomes:

$$\hat{\mathcal{L}}(\mathbf{h} \mid \mathbf{y})\,|_{\mathbf{h}=\hat{\mathbf{h}}_{k-1}} = \sum_{i=1}^{M}\log\left(\frac{1}{M}\sum_{j=1}^{M}K(e(i)-e(j))\right)$$

$$= \sum_{i=1}^{M}\log\sum_{j=1}^{M}K(e(i)-e(j)) - \log M \tag{4.10}$$

The gradient of the log-likelihood can be formulated as:

$$\nabla_{\mathbf{h}}\hat{\mathcal{L}}(\mathbf{h} \mid \mathbf{y})\,|_{\mathbf{h}=\hat{\mathbf{h}}_{k-1}} = \frac{\partial}{\partial \mathbf{h}}\hat{\mathcal{L}}(\mathbf{h} \mid \mathbf{y})\bigg|_{\mathbf{h}=\hat{\mathbf{h}}_{k-1}}$$

$$= \sum_{i=1}^{M}\frac{\partial}{\partial \mathbf{h}}\log\sum_{j=1}^{M}K(e(i)-e(j))\bigg|_{\mathbf{h}=\hat{\mathbf{h}}_{k-1}}$$

$$= \sum_{i=1}^{M}\frac{\sum_{j=1}^{M}\frac{\partial}{\partial \mathbf{h}}K(e(i)-e(j))}{\sum_{k=1}^{M}K(e(i)-e(k))}\bigg|_{\mathbf{h}=\hat{\mathbf{h}}_{k-1}}$$

$$= \frac{1}{\sigma^2}\sum_{i=1}^{M}\frac{\sum_{j=1}^{M}(e(i)-e(j))(\mathbf{x}(i)-\mathbf{x}(j))K(e(i)-e(j))}{\sum_{k=1}^{M}K(e(i)-e(k))}\bigg|_{\mathbf{h}=\hat{\mathbf{h}}_{k-1}} \tag{4.11}$$

Thereby substituting this estimated log-likelihood into eq. (4.7) and iterating till $\hat{\mathbf{h}}_k$ converges to a unique ML channel estimate. The selection of the step-size $\mu(k)$ is considered in the Appendix-A, this setp-size selection makes the channel estimator's update independent of the kernel width. Choosing the appropriate value of σ, the kernel width or smoothing parameter, depends on the type of density to be estimated. If the exact noise pdf was known at the receiver then an appropriate smoothing parameter could be chosen, which minimises the mean integrated square error between the actual and the estimated pdf. However since in this chapter no *a priori* knowledge of the noise pdf is assumed, thus a dynamic estimate for kernel width is used [98]:

$$\sigma = 0.9 \min(\text{standard deviation}, \text{interquartile range}/1.34)\, M^{-1/5} \tag{4.12}$$

The interquartile range is a measure of spread or dispersion. It is the difference between the 75^{th} percentile (often called $Q3$) and the 25^{th} percentile ($Q1$). The formula for interquartile range is therefore: $Q3 - Q1$. This choice of width parameter works well as, when the samples are closer to Gaussian pdf then the 'standard deviation' is less than the 'interquartile range', thereby making the kernel width approximately equivalent to optimal (in trems of mean integrated squares error) kernel width for. On the other hand when the multi-modality is clear, then the parameter 'interqaurtile range' govern the kernel width.

For sample size of 100 the skewness and the multi-modality of the density will be clear by choosing the σ by the above technique. It is also noted from [98] that this smoothing parameter will do very well for a wide range of densities and it is trivial to evaluate.

The NMPL channel estimator is initialised by the LS estimate and iterates on the per received block basis i.e. the eq. (4.7) is iterated over the received block till the channel estimates converge. The algorithm is depicted in Table-4.1.

The sample (training) size is taken as 100-symbols for the proposed algorithm. The effect of training block length for correlated multi-modal Gaussian distribution (i.e. CCI) is discussed in section-4.6.2.

Being a gradient based solution to a non-linear optimisation problem the proposed algorithm may suffer from the usual problems of possible convergence to local maxima (minima). However it is found during simulations that the combination of LS initialisation and selection of

1)	Initialise with LS estimate, $\hat{\mathbf{h}}_k = \hat{\mathbf{h}}_{LS}$
2)	Calculate estimation error, $e(j) = y(j) - \sum_{n=0}^{N_T-1} x(j-n)h(n)$
3)	Estimate the gradient from estimated density, eq. (4.11)
4)	Update channel taps, eq. (4.7)
5)	Follow step 2) to step 4) till converged

Table 4.1: *NPML Channel Estimator*

step size as in the Appendix-A always provided solutions that were superior to the LS one.

In other channel estimation techniques that address the non-Gaussian environment [62], [10], [43], the number of mixtures is assumed known at the receiver, and then the variance and other parameters are estimated iteratively. Here, in order to circumvent the difficulty of estimating the individual parameters of the likelihood (in this case estimation error) pdf, the kernel density estimator is used to estimate the likelihood function directly. By directly estimating the likelihood, the receiver does not need to know (or estimate) the number, the relative probability or variance of different components of the mixture [81]. For the channel estimation problem the MEE algorithm is discussed and the similarity with the NPML algorithm is presented in the next section.

4.4 Minimum error entropy algorithm

In this section a channel estimator based on the MEE criterion is presented and the similarity with the ML channel estimator is observed. Entropy, introduced by Shannon [99], is a scalar quantity that provides a measure for the average information contained in the given probability distribution function. By definition, information is a function of the pdf, hence the entropy as an optimality criterion is more general than MSE. The MSE criterion minimises the energy between the desired and the system output. The minimum mean square error criteria is totally dependent on the second order statistics of the system, which is normally not optimal for non-Gaussian non-linear environment. When the entropy is minimised all the moments of the error pdf, not only the second moments are constrained. The entropy criterion can generally be utilised as an alternative for MSE in supervised adaptation [100]. The MEE principle for minimisation of the distance between the two probabilities is employed here. In the following it will be clear that by minimising the error entropy is equivalent to minimising the distance

between the probability distributions of the desired and the system outputs.

The estimation error-entropy is defined as:

$$H_E = \mathbb{E}\{-\log f(e)\} \tag{4.13}$$
$$= -\int_{-\infty}^{\infty} f(e)\log f(e)\,de \tag{4.14}$$

where $f(e)$ is the pdf of estimation error for the block of length M. This measure is also known as the integral estimate of entropy [101]. As discussed in the previous section, the density is estimated by the kernel density estimator. Thus the non-parametric entropy for large block length and assuming ergodicity (or by re-substitution estimate) becomes [101]:

$$\hat{H}_E = \frac{-1}{M}\sum_{i=1}^{M}\log\hat{f}(e(i))$$
$$= \frac{-1}{M}\sum_{i=1}^{M}\log\sum_{j=1}^{M}K(e(i)-e(j)) + \log M \tag{4.15}$$

Comparing the log-likelihood function of eq. (4.10) with the above equation, they differ only in the scaling term and sign. Since this cost function minimises the entropy, the update equation in eq. (4.7) is modified as:

$$\hat{\mathbf{h}}_k = \hat{\mathbf{h}}_{k-1} - \mu_H(k)\,\nabla_{\mathbf{h}}\hat{H}_E\,|_{\mathbf{h}=\mathbf{h}_{k-1}} \tag{4.16}$$

Substituting the gradient of eq. (4.15) in eq. (4.16) formulates an iterative solution. By substituting $\mu_H(k) = \sigma^2$ (refer Appendix-A) in eq. (4.16) the estimated channel taps always converge to the MEE estimate solution after a few iterations. Use of $\mu_H(k) = \sigma^2$ as an effect makes the adaptation independent of the noise variance.

4.5 Cramér-Rao bound for Gaussian mixture

Earlier in this chapter an ML based channel estimation technique was discussed. To assess the performance of the proposed estimator the fundamental theoretical lower bound is found. The Cramér Rao lower bound defines a lower bound on the variance of an unbiased channel estimator. The lower bound placed on the estimator proves to be extremely useful in practice. At best, it allows to asses that an estimator is a minimum variance estimator. This is the case

when the estimator attains the bound for all the estimated parameters. At worst, it provides a benchmark against which the performance of the unbiased channel estimator is compared. The CRB can generally be easily found in Gaussian noise environments. However it is generally impossible to find closed form lower bounds for non-Gaussian noise environments. Since it is known that the ML estimator is asymptotically unbiased and achieves CRB [42], the lower bounds for the channel estimator in non-Gaussian noise environment are formulated next.

The CRBs for non-Gaussian noise have been studied explicitly in details in [85] [93] [94]. In [85] and later on in [93] the CRB for an autoregressive model was found. In the later work by Swami [94] the case of additive and multiplicative noise was dealt with. In [62] the CRB for estimation of the angle of arrival for complex impulsive noise was formulated. However cited earlier work on the CRB in presence of non-Gaussian noise has been done primarily for the impulsive noise modelled as a two mixture Gaussian process with zero mean. Inspired by [85] and [62] the CRB for the impulsive noise is formulated in this section. In addition to the above, we also consider the case of multi-modal Gaussian mixture noise, which is not considered explicitly in the earlier works. It is observed that the CRB for multi-modal Gaussian mixture depends on the noiseless channel states of the interferer (or means $\hat{w}_l$ in eq. (4.2)) and the additive Gaussian noise variance σ_w^2.

First the CRB theory based on Fisher information is revisited and then the two specific non-Gaussian noise scenarios are considered. Let us define a communication channel:

$$y(k) = b(\mathbf{h}; k) + w(k) \tag{4.17}$$

where $y(k)$ is the received symbol, $b(\mathbf{h}; k) = \sum_{n=0}^{N_T-1} h(n)x(k-n)$ represents the noiseless channel states and $w(k)$ is the additive noise.

The noise is assumed to be i.i.d. and symmetric with pdf $f_w(w)$ which satisfies the regularity conditions (condition for which the expectation of the first derivative of the log-likelihood function w.r.t. the parameter is zero) so that the CRB exists [42]. The CRB for any unbiased estimator $\hat{h}_i$ of a component h_i of $\mathbf{h}$, is given by $var(\hat{h}_i) \geq [\mathbf{J_h}^{-1}]_{ii}$, where $\mathbf{J_h}$ is the Fisher information matrix (FIM) for $\mathbf{h}$ with elements

$$J_{ij} = \mathbb{E}\left\{ \left(\frac{\partial \log f_{\mathbf{y}}(\mathbf{y})}{\partial h_i} \right) \left(\frac{\partial \log f_{\mathbf{y}}(\mathbf{y})}{\partial h_j} \right) \right\} \tag{4.18}$$

where column vector $\mathbf{y} = [y(1), ..., y(M)]$ is the set of observation with pdf $f_{\mathbf{y}}(\mathbf{y}) = \prod_{k=1}^{M} f_w(y(k)-$

$b(\mathbf{h}, k)$). By the chain rule:

$$\frac{\partial}{\partial h_i} f_w(y(k) - b(\mathbf{h}, k)) = \frac{\partial}{\partial w} f_w(w) \left[-\frac{\partial b(\mathbf{h}, k)}{\partial h_i} \right] \tag{4.19}$$

Thus element (i, j) of the FIM has the form [94]:

$$J_{ij} = \mathcal{I} \sum_{k=1}^{M} \frac{\partial b(\mathbf{h}, k)}{\partial h_i} \frac{\partial b(\mathbf{h}, k)}{\partial h_j} \tag{4.20}$$

where $\mathcal{I}$ depends on the pdf of the noise. For identically distributed observations $\mathcal{I} = M\tau$ [42] where τ is defined as:

$$\tau = \mathbb{E}\left\{ \left(\frac{\partial \log f_w(w)}{\partial w} \right)^2 \right\} = \int_{-\infty}^{\infty} \left(\frac{(f'_w(w))^2}{f_w(w)} \right) \partial w = 2 \int_{0}^{\infty} \left(\frac{(f'_w(w))^2}{f_w(w)} \right) \partial w \tag{4.21}$$

The FIM reduces to a diagonal matrix (as the input sequence is independent and zero mean). Calculating the inverse is now trivial. Now eq. (4.20) and eq. (4.21) are applied to the two special cases where the observation noise can be described as a Gaussian mixture.

4.5.1 Uni-modal mixture

Impulsive noise is often modelled as a finite mixture of zero-mean Gaussian processes [10] [80]. The pdf has the form:

$$f_w(w) = \frac{1}{\sqrt{2\pi}} \sum_{l=1}^{N_M} \frac{\lambda_l}{\sigma_l} \exp\left(\frac{-w^2}{2\sigma_l^2} \right) \tag{4.22}$$

Thus:

$$\left(\frac{\partial}{\partial w} f_w(w) \right)^2 = \frac{w^2}{\sqrt{2\pi}\sqrt{2\pi}} \sum_{l=1}^{N_M} \sum_{q=1}^{N_M} \frac{\lambda_l \lambda_q}{\sigma_l^3 \sigma_q^3} \exp\left(\frac{-w^2}{2} \left(\frac{1}{\sigma_l^2} + \frac{1}{\sigma_q^2} \right) \right) \tag{4.23}$$

and:

$$\tau = \int_{-\infty}^{\infty} \frac{1}{f_w(w)} \left(\frac{\partial}{\partial w} f_w(w) \right)^2 \partial w = \frac{2}{\sqrt{2\pi}} \int_{0}^{\infty} \frac{\sum_{l=1}^{N_M} \sum_{q=1}^{N_M} \frac{\lambda_l \lambda_q}{\sigma_l^3 \sigma_q^3} \exp\left(\frac{-w^2}{2} \left(\frac{1}{\sigma_l^2} + \frac{1}{\sigma_q^2} \right) \right)}{\sum_{r=1}^{N_M} \frac{\lambda_r}{\sigma_r} \exp(\frac{-w^2}{2\sigma_r^2})} w^2 \partial w \tag{4.24}$$

Thus the CRB in this case is similar to the one formulated for angle of arrival estimation in [62]:

$$var(\hat{h}) \geq \frac{1}{\tau M \sigma_x^2} \tag{4.25}$$

where σ_x^2 is variance of the input symbols x (eq. (4.17)).

4.5.2 Multi-modal mixture

In deriving a CRB for the multi-modal case it is assumed that the process is i.i.d. and the observed pdf can be written as:

$$f_w(w) = \frac{1}{\sqrt{2\pi\sigma_w^2}} \sum_{l=1}^{N_M} \lambda_l \exp\left(\frac{-(w-w_l)^2}{2\sigma_w^2}\right) \tag{4.26}$$

where σ_w^2 is the variance of the Gaussian noise (without loss of generality the Gaussian noise variance is assumed the same for all the mixture components). w_l represents the different modes (or channel states of the interferer) for the distribution. Hence:

$$\left(\frac{\partial f_w(w)}{\partial w}\right)^2 = \left(\frac{-1}{\sqrt{2\pi\sigma_w^2}}\frac{1}{\sigma_w^2}\right)^2 \sum_{l=1}^{N_M}\sum_{q=1}^{N_M} \lambda_l\lambda_q(w-w_l)(w-w_q) \exp\left(\frac{-1}{2\sigma_w^2}\left\{(w-w_l)^2+(w-w_q)^2\right\}\right) \tag{4.27}$$

Assuming that the interferer's channel states are equi-probable ($\lambda_1 = ... = \lambda_{N_M}$):

$$\begin{aligned}\tau &= \int_{-\infty}^{\infty} \frac{1}{f_w(w)}\left(\frac{\partial}{\partial w} f_w(w)\right)^2 \partial w \\ &= \frac{2}{N_M\sqrt{2\pi\sigma_w^2}\sigma_w^4}\int_0^{\infty} \frac{\sum_{l=1}^{N_M}\sum_{q=1}^{N_M}(w-w_l)(w-w_q)\exp\left(\frac{-1}{2\sigma_w^2}\left\{(w-w_l)^2+(w-w_q)^2\right\}\right)}{\sum_{l=1}^{N_M}\exp\left(\frac{-1}{2\sigma_w^2}(w-w_l)^2\right)}\partial w\end{aligned} \tag{4.28}$$

Thus:

$$var(\hat{h}) \geq \frac{1}{\tau M \sigma_x^2} \tag{4.29}$$

4.6 Simulation results

The robustness and performance of the proposed adaptive algorithm (in section-4.3) is verified for two cases where the CRB has been derived. In the first case, the noise is simulated as

mixture of two zero mean Gaussian noises with different variances, to simulate impulsive noise. In the next set of simulations a communication channel model, like GSM, considering CCI with Gaussian noise as a multi-modal, i.i.d., Gaussian mixture interference as discussed in [82] is assumed. The performance loss by estimating the pdf by kernel density estimator for NPML is also compared with the case where it is assumed that the exact mixture pdf is known at the receiver. The performance of the channel estimator is calculated by normalised mean square error (NMSE), as shown in eq. (4.30).

$$NMSE = \frac{\mathbb{E}\{(h - \hat{h})^2\}}{\mathbb{E}\{h^2\}} \tag{4.30}$$

where h is the actual channel and $\hat{h}$ is the estimated channel. For all simulation results, the input symbols of length 100 and ensemble of 1000-runs is considered.

4.6.1 Uni-modal Gaussian mixture

As seen earlier, the uni-modal Gaussian mixture model has been used in many applications to model impulsive noise [85] [62]. This is achieved if $\forall\, l; \hat{w}_l = 0$ in eq. (4.2) to obtain eq. (4.22) in this chapter. For a popular impulsive noise model from eq. (4.22), $N_M = 2$ as in [85] [62] is considered. The variance $\sigma_2^2 \gg \sigma_1^2$ with relative probabilities $\lambda_2 < \lambda_1$, so that large noise samples with variance σ_2^2 occur with frequency λ_2 in a background of Gaussian noise with variance σ_1^2. The performance of NPML estimator with other algorithms for a single-tap channel (used to model flat-fading) in impulsive noise is compared. Then we take a more realistic 5-tap channel (usually used to simulate a GSM communication channel).

In Figure 4.4 and Figure 4.5, the legends LS, $\text{NPML}_{\text{estimated pdf}}$, $\text{NPML}_{\text{known pdf}}$ and $\text{CRB}_{\text{impulsive}}$ stand for least squares estimate, NPML estimate when the estimated log-likelihood (eq. (4.10)) is used in eq. (4.7), NPML estimate (discussed in section-4.3) when the pdf in likelihood function in eq. (4.7) is known and Cramér Rao bound as estimated by eq. (4.25) respectively. The legend EM(2-mix) is where the EM algorithm of [81] was used, where EM assumed two mixtures. From Figure 4.4, it is observed that EM(2-mix) and $\text{NPML}_{\text{known pdf}}$ have similar performance. Performance loss by density estimation can also be observed. The EM algorithm as suggested in [81] performs closer to CRB and with less computation than $\text{NPML}_{\text{estimated pdf}}$.

However, when the EM algorithm assumes 4-mixtures for channel estimation with the simulation environment being the same, the over-parametrisation (since the impulsive noise is mod-

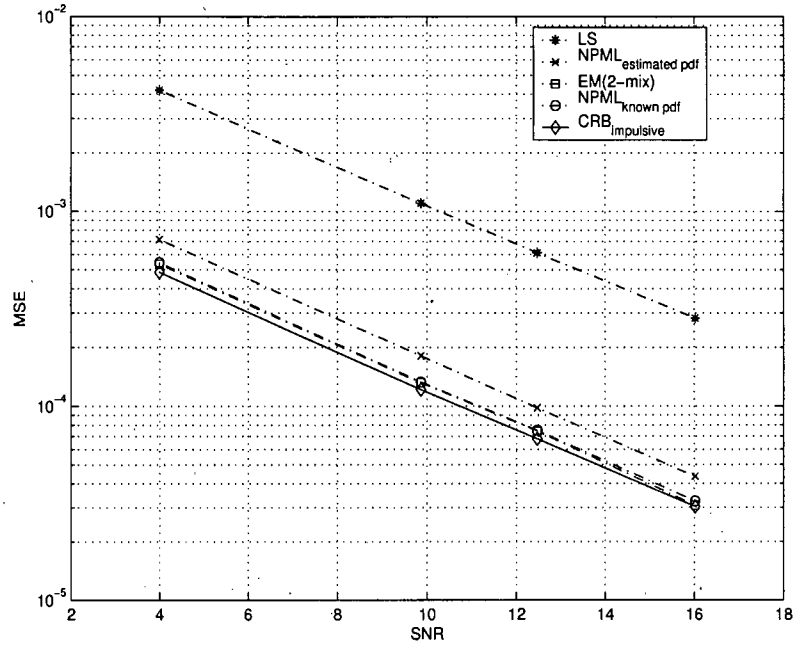


Figure 4.4: *NPML comparison with 2-mixture EM for impulsive noise (EM(2-mix) and* $NPML_{\text{known pdf}}$ *are overlaid)*

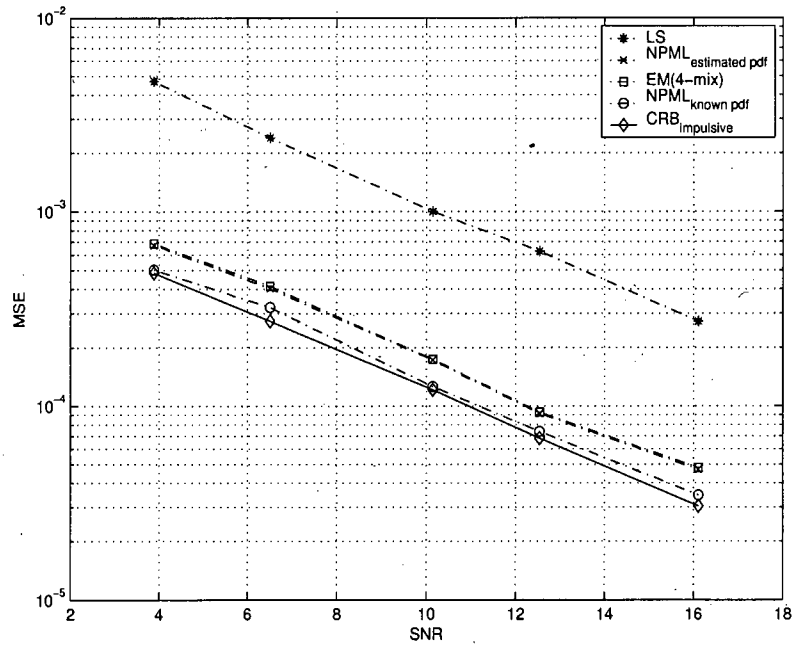


Figure 4.5: *NPML comparison with 4-mixture EM for impulsive noise (EM(4-mix) and* $NPML_{\text{estimated pdf}}$ *are overlaid)*

elled as mixture of two Gaussian) by the EM in Figure 4.5 causes degradation in the EM-based channel estimator's performance. It is observed from Figure 4.5 that both the NPML and the EM(4-mix) algorithms performance coincides whereas the $\text{NPML}_{\text{known pdf}}$ outperforms both the EM(4-mix) and $\text{NPML}_{\text{estimated pdf}}$. Thus EM-based algorithm looses some of its performance when over-parametrised, thus *a priori* knowledge of number of mixtures becomes an important factor for the EM-based estimator. Moreover it is observed that the algorithm in [81] cannot be used for ISI affected channels in the presence of impulsive noise. For the second case, an ISI affected communication channel modelled as h = [-0.227 0.460 0.688 0.460 -0.227] with the noise modelled as $\frac{\sigma_1^2}{\sigma_2^2} = 411$ for $\lambda_1 = 0.9$ and $\lambda_2 = 0.1$ is considered. The proposed NPML algorithm has a fast and smooth convergence, as shown in Figure 4.13. It is also verified from the plots in Figure 4.6 that the noise pdf and the error pdf are quite closely matched for SNR = 4.5dB, which confirms that the NPML algorithm estimates the likelihood function very closely. Impulsive noise is characterised by heavier tails than a normal distribution. The performance in the tails is better illustrated with Figure 4.7 which shows the 'log' of the actual noise and estimated (error) pdf. The MSE performance of the proposed algorithm is shown in Figure 4.8, the $\text{NPML}_{\text{known pdf}}$ reaches quite close to the $\text{CRB}_{\text{impulsive}}$ in MSE terms. From Figure 4.8 it is observed that there is loss of 2dBs by using the kernel density estimator when comparing $\text{NPML}_{\text{estimated pdf}}$ and $\text{NPML}_{\text{known pdf}}$ curves. However there is significant gain by using the NPML algorithm instead of LS as apparent from comparing LS and $\text{NPML}_{\text{estimated pdf}}$ curves. Thus better (in terms of MSE) estimates and faster convergence is achieved by using NPML algorithm over LS for impulsive noise.

4.6.2 Multi-modal Gaussian mixture

In this set of simulations it can be assumed that the co-channel interference with additive Gaussian noise is a manifestation of a Gaussian mixture with different means, however the overall mean (in ideal condition) of the whole Gaussian mixture processes is zero. Two cases of multi-modal Gaussian mixture distribution are considered: first one for i.i.d. mixture and the next for correlated mixture typical of practical systems in the presence of CCI.

Since the CRB (eq. (4.29)) of section-4.5.2 is only valid for an i.i.d. Gaussian mixture, the algorithm of section-4.3 is first assessed by generating i.i.d. noise with pdf identical to that of $w(k)$ as in eq. (4.33). The performance of the proposed algorithm is compared in Figure 4.9. The legends LS, $\text{NPML}_{\text{estimated pdf}}$, $\text{NPML}_{\text{known pdf}}$, and CRB_{CCI} represents the least

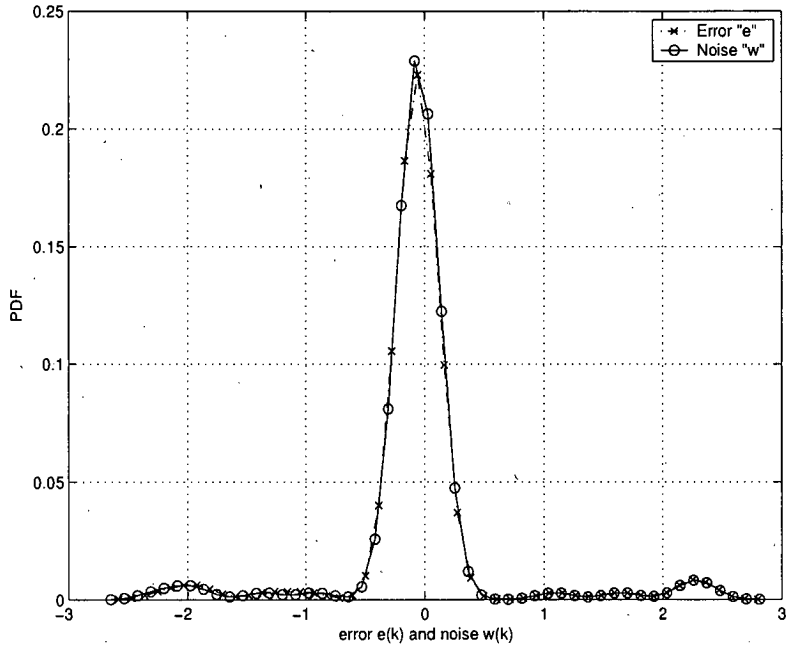


Figure 4.6: *Comparison of pdf fits achieved by NPML algorithm after convergence*

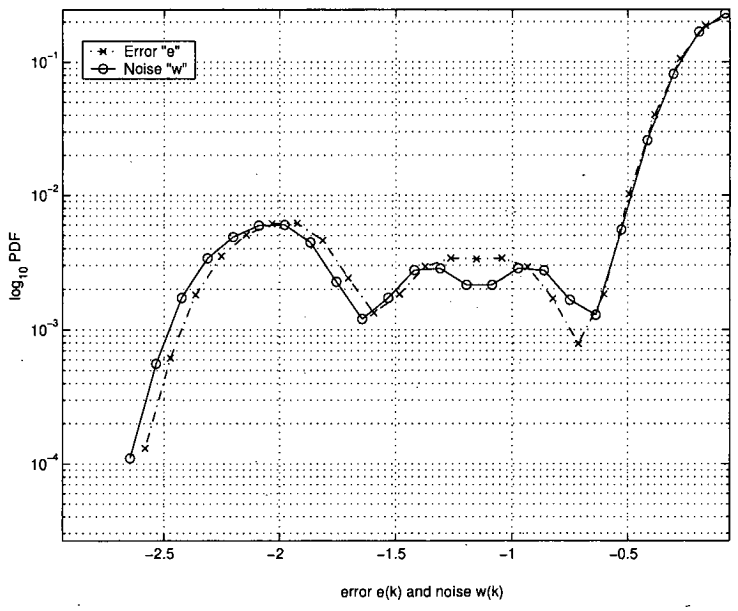


Figure 4.7: *Comparison of pdf tails fits achieved by NPML algorithm after convergence*

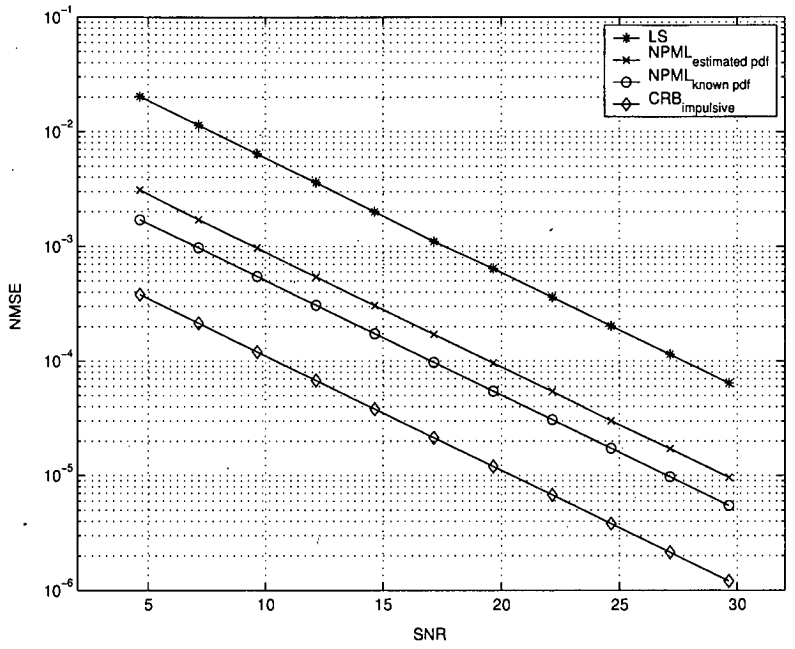


Figure 4.8: *NMSE plot for channel estimators for mixture when* $\forall\, l; \hat{w}_l = 0$

squares estimate, NPML estimate when the estimated log-likelihood (eq. (4.10)) is used in eq. (4.7), NPML estimate for known pdf likelihood function in eq. (4.7) and Cramér Rao bound as estimated by eq. (4.29) respectively. From comparing $\text{NPML}_{\text{estimated pdf}}$ with $\text{NPML}_{\text{known pdf}}$ in Figure 4.9, it is observed that there is not much loss in using the kernel density estimators to estimate the likelihood function. As also observed, the NPML algorithm gets closer to the CRB_{CCI} and also performs better than the LS estimator. Finally performance of the algorithm is considered when the CCI is correlated. A typical communication system affected by co-channel interference is shown in Figure 4.10. The corrupting (noise + CCI) noise pdf deviates from Gaussianity as apparent from the Figure 4.11 at SNR = 24dBs and SIR = 10dB. Also for the GSM scenarios it was shown in [82] that noise pdf deviates from Gaussianity in the presence of interference. The co-channels are each of order $N_T - 1$ and are represented as g_p and interfering signal as u_p for $p = 0, ..., P_I - 1$, where P_I represents number of interferers.

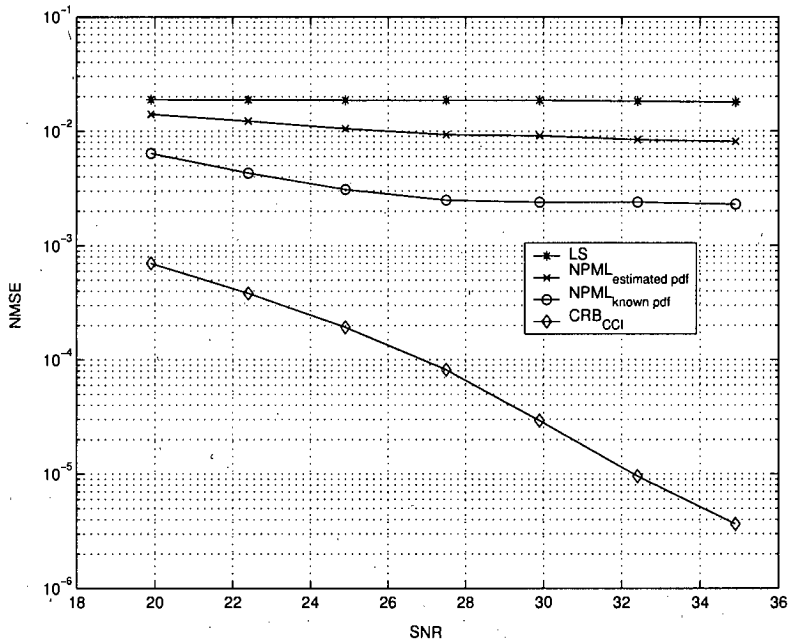


Figure 4.9: *NMSE plot for multi-modal noise (uncorrelated CCI) affected communication system where h = [-0.227 0.460 0.688 0.460 -0.227], w_l are interfering channel states, SIR=4.73dB for 100-symbols over an ensemble of 1000-runs*

The received signal can be represented as

$$y(k) = \sum_{n=0}^{N_T-1} h(n)x(k-n) + \sum_{p=0}^{P_I-1}\sum_{n=0}^{N_T-1} g_p(n)u_p(k-n) + n(k) \quad (4.31)$$

$$= \sum_{n=0}^{N_T-1} h(n)x(k-n) + w_l(k) + n(k) \quad (4.32)$$

$$= \sum_{n=0}^{N_T-1} h(n)x(k-n) + w(k) \quad (4.33)$$

where the middle (double summation) term on the RHS in eq. (4.31) represents the CCI (or the interfering noiseless channel states w_l). The interfering channel states forms the means of the multi-modal mixture distribution which can take finite states and are symmetrical. $n(k)$ is a zero mean, i.i.d., Gaussian noise process with variance σ_n^2 and $k = 1, \ldots, M$ represents the number of symbols. Please note that σ_w^2 in eq. (4.28) for CRB (eq. 4.29) is equal to σ_n^2.

The above presented algorithm is verified for real stationary channel for $N_T = 5$. The input signal is anti-podal random input sequence. The channels are assumed to be

h = [-0.227 0.460 0.688 0.460 -0.227] and

g_0 = [-0.10 0.40 1.0 0.40 -0.10]

where h suffers from amplitude and phase distortion [20], and g_0 is the co-channel considered for the simulation.

NPML formulation in eq. (4.8) assumes that $w(k)$ is i.i.d. and thus the algorithm is sub-optimal in this environment both in 'unknown' and 'known' pdf form. The $w(k)$ pdf is identical to the form used to generate Figure 4.9, however $w(k)$ is now generated from eq. (4.33). Figure 4.12 illustrates the performance of the algorithms in the correlated co-channel interference. The legends $\text{NPML}_{\text{estimated pdf}}$ and $\text{NPML}_{\text{known pdf}}$ are the same as above, whereas $\text{NPML}_{\text{cci(estimated pdf)}}$ and $\text{NPML}_{\text{cci(known pdf)}}$ represents the performance when the interference is generated from eq. (4.33). Figure 4.12 illustrates the performance loss incurred in using the algorithm in a correlated CCI environment. From the simulations for co-channel interference it is observed that the proposed algorithm has faster and stable convergence as shown in Figure 4.13 when step-size $\mu(k)$ in eq. (4.7) is chosen as $\frac{\sigma^2}{M}$. The effect of training symbol block length on various discussed estimators is also shown in Figure 4.14. There is a significant improvement in channel estimation with respect to standard LS and the estimated pdf version of

the algorithm is not degraded significantly w.r.t. the known pdf version. It is also interesting to note that the [81] algorithm is not suitable for CCI affected channels, as it converges to the LS solution when used for flat-fading channel estimation in presence of interference. The proposed NPML algorithm is robust to ISI and type of non-Gaussianity, as seen from the simulations and it reaches closer to the CRB than other algorithms.

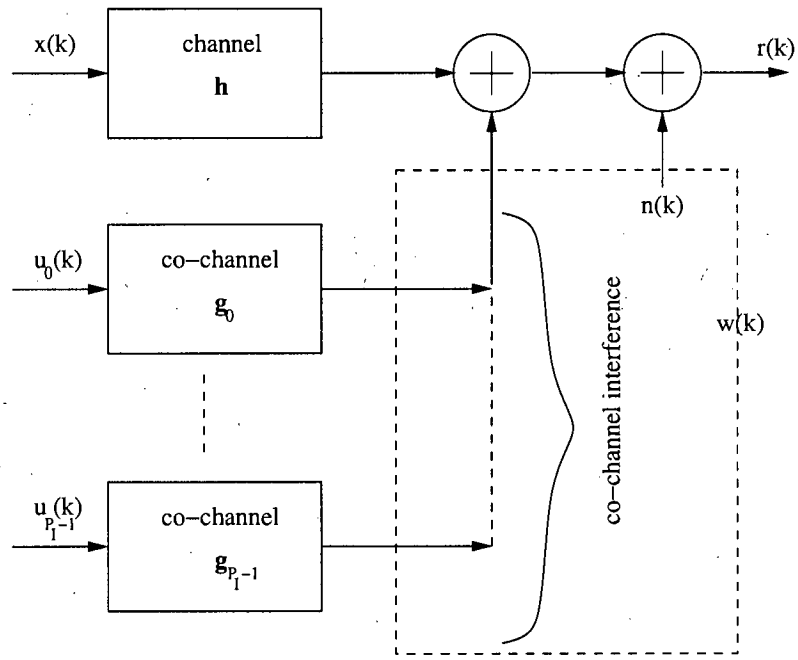


Figure 4.10: *A typical CCI affected communication system*

4.7 Conclusion

It was shown that the channel estimator based on Gaussian noise assumption (LS) is inferior in the non-Gaussian noise. The non-Gaussian noise was modelled and estimated as a Gaussian mixture. It is seen that the difference between NPML and MEE is trivial for the channel estimator. It was also shown that better channel estimates can be obtained by using iterative NPML based algorithm using Parzen's kernel for density estimation. The same algorithm (without making any change) can be used for channel estimation in uni-modal and multi-modal noise environments. It was demonstrated that an approximation to unknown (at receiver) noise pdf can be achieved iteratively by the NPML algorithm and that the quality of that approximation does not seems to significantly affect algorithm's performance. A generalised approach

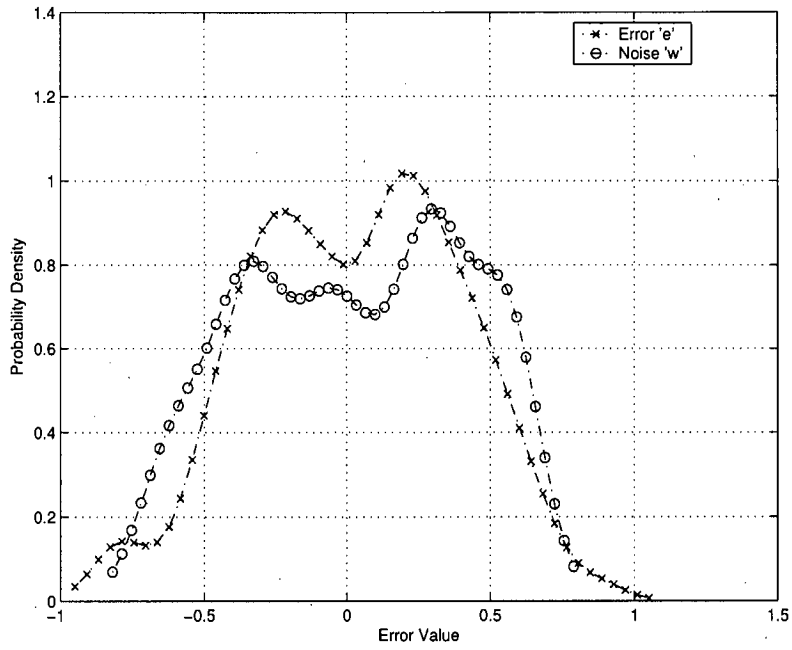


Figure 4.11: *Distribution of the estimation error by minimum entropy algorithm 'e' and additive noise 'w' for CCI corrupted channels*

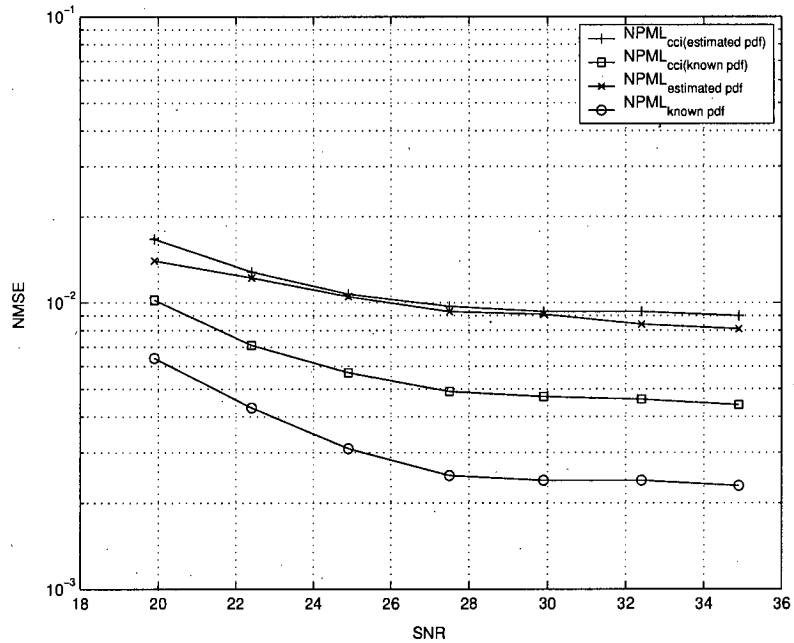


Figure 4.12: *NMSE plot for co-channel affected communication system where h = [-0.227 0.460 0.688 0.460 -0.227], g_0 = [-0.10 0.40 1.0 0.40 -0.10], SIR=4.73dB for 100-symbols over an ensemble of 1000-runs*

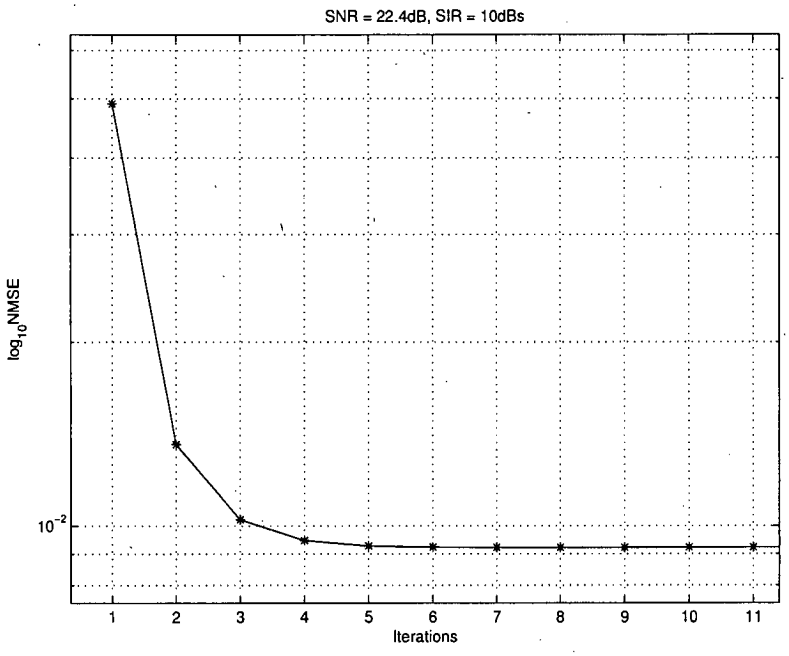


Figure 4.13: *Convergence plot for co-channel affected communication system for 100-samples over an ensemble of 100-runs at SNR = 22.4dB and SIR = 10dB*

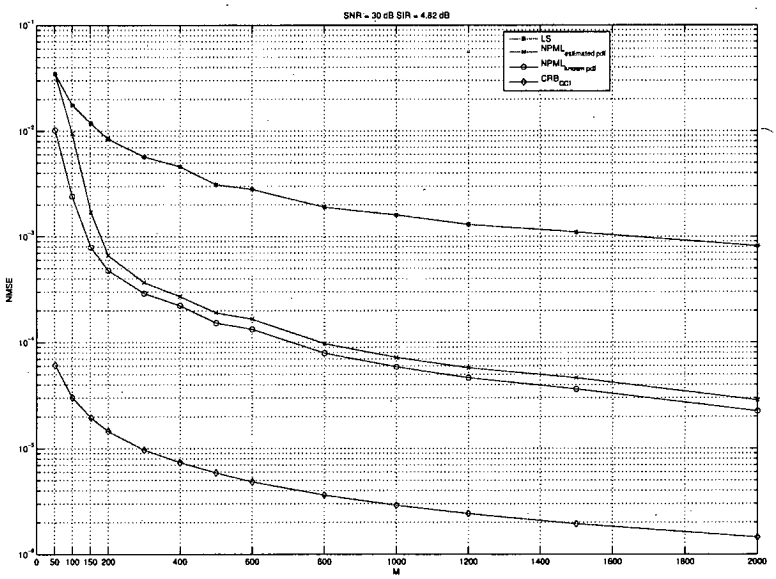


Figure 4.14: *NMSE with various training length*

to calculate theoretical lower bounds for channel estimation in non-Gaussian noise was also presented. From the results it can be observed that kernel density estimator does not result in much loss in performance. The proposed algorithm also has faster and stable convergence and is immune to sudden changes in noise conditions. Thus the channel estimator based on the ML criterion using kernel density estimator forms a robust channel estimator for various additive non-Gaussian noise sources.

Chapter 5

Non-parametric maximum likelihood channel estimator in the presence of correlated non-Gaussian noise

This chapter is an extension of the algorithm developed in the previous chapter. In chapter 4, the noise distribution was modelled as uni-modal and multi-modal distributions. The uni-modal distribution was modelled as a mixture of two independent finite Gaussian processes with zero-mean. For the multi-modal distribution, a finite Gaussian mixture process with various means was used to simulate the CCI. The NPML algorithm presented in the previous chapter assumes that the additive noise plus interference is independent. However by nature of CCI generation, the CCI is correlated. The performance loss due to the uncorrelated assumption was witnessed in the previous chapter.

Since it is known that the interference is correlated, this correlation is reduced by using an error whitening filter. Techniques which whiten the noise plus interference before suppressing the interference have been proposed earlier in [66] [65]. This forms a powerful technique to improve the performance, but since, in practice, the tap length of this whitening filter cannot be increased to a large value, the ideal assumption of white Gaussian noise (after the linear prediction error (LPE) filter [82]) does not hold. Thereby, in this chapter a joint maximum likelihood estimate of the channel taps and the whitening filter (LPE) coefficients is formulated.

In this chapter no such (Gaussian) assumption on the distribution of the whitened noise is made, which makes this technique robust to various noise distributions. After whitening, the unknown whitened likelihood pdf is estimated by using a kernel density estimator at the receiver. Thereby combining the log-likelihood as a cost function with a whitening filter and a kernel density estimate, a robust channel estimator for correlated noise environments is formed. The simulations for co-channel interference in the presence of Gaussian noise, confirms that a better estimate can be obtained by using the proposed technique as compared to the traditional least squares algorithm with whitening filter, which is optimal in Gaussian noise environments.

The chapter is organised as follows. First, the problem statement is formulated in section-5.1 for a general communication system with whitening filter. The non-parametric maximum likelihood channel estimator algorithm with LPE filter and kernel density estimator is discussed in section-5.2. In section-5.3 simulation results are presented. Conclusions based on analysis and simulation are drawn at the end.

5.1 Formulation of the problem

The discrete-time model in the low-pass equivalent form of the communication system channel estimator is shown in Figure 5.1. Without loss of generality, the input signal is assumed to be randomly generated binary anti-podal PAM signal, so that the transmitted symbols are $\mathbf{x} \epsilon \{\pm 1\}$. Here '$y$' represents the received signal and 'w' is a sequence of additive noise. The model is simplified by assuming that the channel is of order $N_T - 1$ i.e. $\mathbf{h} = [h(0), h(1), ..., h(N_T - 1)]$.

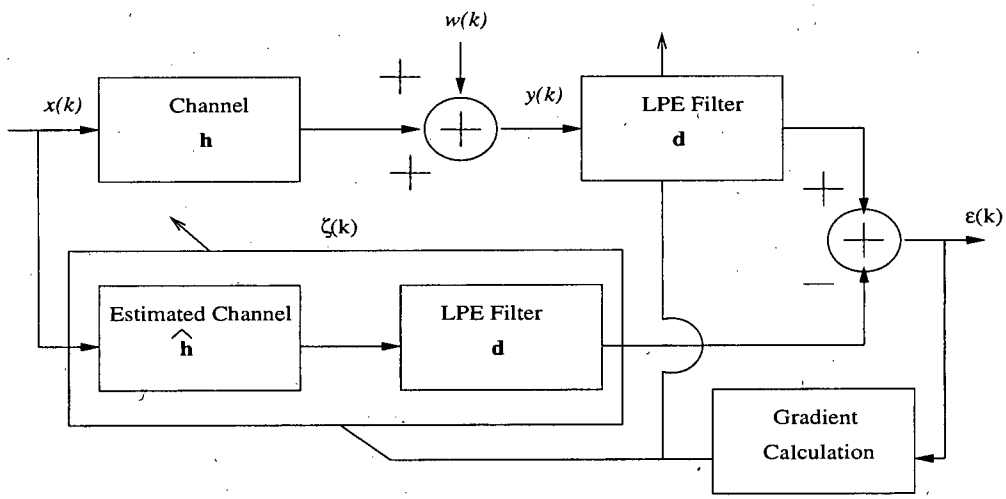


Figure 5.1: *Communication systems channel estimator with LPE filter*

More precisely, the received signal $y(k)$ sampled once per symbol can be written as

$$y(k) = \sum_{n=0}^{N_T-1} h(n)x(k-n) + w(k) \tag{5.1}$$

As in the previous chapter, the problem is to estimate the channel coefficients from the received signal assuming that the input signal (as in supervised training mode) and the channel (tap) length is known at the receiver. Thus the problem reduces to the well known problem of system

identification. There are various algorithms based on different criteria to estimate the channel taps. Usually the LS solution is taken as the optimum solution for the Gaussian noise environments where it is equivalent to a ML estimate [39]. However, here it is assumed that the noise in the presence of interference is correlated, thus LS does not provide the ML solution. To remove this correlation an LPE filter is used. Basically, as the order of the prediction error filter increases, successively the correlation between adjacent samples of the input process reduces, until ultimately a point is reached where for high enough an order it produces an output process that consists of a sequence of uncorrelated samples [16]. The whitening of the original process applied to the filter input will have thereby been accomplished. The problem then reduces to the one shown in Figure 5.1.

The problem can now be written as:

$$\sum_{i=0}^{P} d(i)y(k-i) = \sum_{l=0}^{N_T+P-1} x(k-l)\zeta(l) + \epsilon(k) \tag{5.2}$$

where $\mathbf{d} = [d(0) = 1\ d(1) = -\xi(1) \ldots d(P) = -\xi(P)]$ are the coefficients of the LPE filter and the equivalent channel taps vector $\boldsymbol{\zeta} = [\zeta(0)\,\zeta(1) \ldots \zeta(N_T + P - 1)]$, where $\zeta(l) = \sum_i d(i)h(l-i)$. Ideally the $\epsilon(k)$ is a zero-mean white Gaussian process. Since from eq. (5.1), the model eq. (5.2) corresponds to assuming

$$\sum_{i=0}^{P} d(i)y(k-i) = \sum_{i=0}^{P} d(i) \sum_{l=0}^{N_T-1} x(k-i-l)h(l) = \sum_{l=0}^{N_T+P-1} x(k-l)\zeta(l) \tag{5.3}$$

and

$$\sum_{i=1}^{P} d(i)w(k-i) = \epsilon(k) \tag{5.4}$$

Therefore, the effect of the LPE filter is that of whitening the additive disturbance $w(k)$. The formulation eq. (5.2) permits the description of the channel plus the whitening filter as a vector inner product, which in turn allows the simultaneous estimation of the LPE coefficients and the equivalent channel taps at the output of the LPE filter [66]. In fact, letting $\boldsymbol{\xi} = [\xi(1) \ldots \xi(P)]$,

eq. (5.2) can be rewritten as

$$[1 \; -\xi(1) \ldots \; -\xi(P)] \begin{bmatrix} y(k) \\ y(k-1) \\ \vdots \\ y(k-P) \end{bmatrix} = [\zeta(0)\,\zeta(1) \ldots \zeta(N_T+P-1)] \begin{bmatrix} x(k) \\ x(k-1) \\ \vdots \\ x(k-N_T-P+1) \end{bmatrix} + \epsilon(k) \quad (5.5)$$

or equivalently

$$y(k) = [\xi(1) \ldots \xi(P)] \begin{bmatrix} y(k-1) \\ \vdots \\ y(k-P) \end{bmatrix} + [\zeta(0)\,\zeta(1) \ldots \zeta(N_T+P-1)] \begin{bmatrix} x(k) \\ x(k-1) \\ \vdots \\ x(k-N_T-P+1) \end{bmatrix} + \epsilon(k) \quad (5.6)$$

$$= [\boldsymbol{\xi} \; \boldsymbol{\zeta}]\mathbf{v}(k) + \epsilon(k) \quad (5.7)$$

where $\mathbf{v}(k) = [y(k-1) \ldots y(k-P)\, x(k)\, x(k-1) \ldots x(k-N_T-P+1)]^T$ and with $\epsilon(k)$ white. Usually it is assumed that due to LPE filter $\epsilon(k)$ is Gaussian distributed, thus the log-likelihood function of $y(0), \ldots, y(M-1)$ given $[\boldsymbol{\xi} \; \boldsymbol{\zeta}]$ and $\mathbf{v}(0), \ldots, \mathbf{v}(M-1)$ is:

$$-\log f(y(0), \ldots, y(M-1) \mid [\boldsymbol{\xi} \; \boldsymbol{\zeta}], \mathbf{v}(0), \ldots, \mathbf{v}(M-1)) = -\log f(\epsilon(0), \ldots, \epsilon(M-1)) = \frac{1}{2\sigma^2} \sum_{k=0}^{M-1} \mid y(k) - [\boldsymbol{\xi} \; \boldsymbol{\zeta}]\mathbf{v}(k) \mid^2 \quad (5.8)$$

(having neglected the constant term $M \log 2\pi\sigma^2$). Therefore, assuming the knowledge of $\mathbf{v}(0), \ldots, \mathbf{v}(M-1)$, the ML estimate $[\hat{\boldsymbol{\xi}} \; \hat{\boldsymbol{\zeta}}]$ of the vector $[\boldsymbol{\xi} \; \boldsymbol{\zeta}]$ can be obtained by minimising eq. (5.8) with respect to $[\boldsymbol{\xi} \; \boldsymbol{\zeta}]$. This corresponds to the least-squares estimation of the unknown parameters $\boldsymbol{\xi}$ and $\boldsymbol{\zeta}$. Defining

$$\mathbf{R} = \frac{1}{M} \sum_{i=0}^{M-1} \mathbf{v}_i^* \mathbf{v}_i^T \quad (5.9)$$

$$\mathbf{p} = \frac{1}{M} \sum_{i=0}^{M-1} \mathbf{v}_i^* y_i \quad (5.10)$$

The LS estimate would be:

$$[\hat{\boldsymbol{\xi}}\ \hat{\boldsymbol{\zeta}}] = \mathbf{R}^{-1}\mathbf{p} \tag{5.11}$$

A regularisation term (usually $\ll 1$) was introduced in [8] for the least squares solution. This term is however ignored in deriving the above equation.

In a conventional receiver, the equaliser following the whitening filter uses the LPE filter output and the estimated channel $\hat{\boldsymbol{\zeta}}$. To have the Gaussian assumption (eq. 5.8) valid the tap-length of $\hat{\boldsymbol{\zeta}}$ should be large. However in practice the larger the tap-length of the LPE filter, the more the equaliser states, the more computational complexity [8] for maximum likelihood sequence equaliser. Thus in practice the tap-length of the LPE filter is usually restricted to either 1 or 2-taps [8] [65]. This restriction leads $\epsilon(k)$ to deviate from Gaussianity. The channel estimator proposed in this thesis does two tasks: (i) estimating the channel (and LPE coefficients); (ii) estimating the uncorrelated $\epsilon(k)$ pdf at the receiver.

5.2 Non-parametric maximum-likelihood channel estimation with LPE filter

For the communication system represented by eq. (5.2) the ML estimate forms the optimal estimator for the channel. This problem can be viewed as the joint optimisation problem [82], where the likelihood is maximised with respect to $\boldsymbol{\xi}$ and $\boldsymbol{\zeta}$. If the $\epsilon(k)$ was Gaussian then the LS solution as found in [82] could have been applied directly. However, since it is assumed that $\epsilon(k)$ is non-Gaussian and can be modelled as a Gaussian mixture, the kernel density estimator, as described in section-4.2, is used to estimate this unknown density. Since the kernel density estimator is essentially a Gaussian mixture formulation, a closed form estimate of the $\boldsymbol{\xi}$ and $\boldsymbol{\zeta}$ cannot be obtained. Thus an iterative joint optimisation technique, smiliar to the one in the previous chapter is used here:

$$\hat{\boldsymbol{\xi}}_k = \hat{\boldsymbol{\xi}}_{k-1} + \mu(k)\,\nabla_{\boldsymbol{\xi}}\mathcal{L}(\boldsymbol{\xi} \mid \mathbf{y}, \boldsymbol{\zeta})\mid_{\boldsymbol{\xi}=\hat{\boldsymbol{\xi}}_{k-1}, \boldsymbol{\zeta}=\hat{\boldsymbol{\zeta}}_{k-1}} \tag{5.12}$$

$$\hat{\boldsymbol{\zeta}}_k = \hat{\boldsymbol{\zeta}}_{k-1} + \mu(k)\,\nabla_{\boldsymbol{\zeta}}\mathcal{L}(\boldsymbol{\zeta} \mid \mathbf{y}, \boldsymbol{\xi})\mid_{\boldsymbol{\zeta}=\hat{\boldsymbol{\zeta}}_{k-1}, \boldsymbol{\xi}=\hat{\boldsymbol{\xi}}_{k-1}} \tag{5.13}$$

where $\mu(k)$ is the adaptation step-size, $\hat{\boldsymbol{\xi}}_k$ and $\hat{\boldsymbol{\zeta}}_k$ are estimates of $\boldsymbol{\xi}$ and $\boldsymbol{\zeta}$ respectively at time instance k. Since the channel estimator is assumed to have no *a priori* knowledge of the error

pdf $f_\epsilon(.)$, this unknown pdf is then estimated by using the kernel density estimator eq. (4.3) with Gaussian kernels as shown below. As the kernel density estimators are known to be effective in density estimation over short data record, a technique over the available data (error) record, of length M, to estimate this unknown density is used. From the definition of kernel density estimation [92]:

$$\hat{f}_\epsilon(\epsilon) = \frac{1}{M}\sum_{j=1}^{M} K(\epsilon - \epsilon(j)) \tag{5.14}$$

Thus the estimated (joint) log-likelihood function can be written as:

$$\begin{aligned}\hat{\mathcal{L}}(\boldsymbol{\xi}, \boldsymbol{\zeta} \mid \mathbf{y})\mid_{\boldsymbol{\xi}=\hat{\boldsymbol{\xi}}_{k-1}, \boldsymbol{\zeta}=\hat{\boldsymbol{\zeta}}_{k-1}} &= \sum_{i=1}^{M} \log\left(\frac{1}{M}\sum_{j=1}^{M} K(\epsilon(i) - \epsilon(j))\right) \\ &= \sum_{i=1}^{M} \log \sum_{j=1}^{M} K(\epsilon(i) - \epsilon(j)) - \log M \end{aligned} \tag{5.15}$$

The gradient $\hat{\boldsymbol{\xi}}$ of the joint log-likelihood can be formulated as:

$$\begin{aligned}\nabla_{\boldsymbol{\xi}} \hat{\mathcal{L}}(\boldsymbol{\xi} \mid \mathbf{y}, \boldsymbol{\zeta})\mid_{\boldsymbol{\xi}=\hat{\boldsymbol{\xi}}_{k-1}, \boldsymbol{\zeta}=\hat{\boldsymbol{\zeta}}_{k-1}} &= \frac{\partial}{\partial \boldsymbol{\xi}} \hat{\mathcal{L}}(\boldsymbol{\xi} \mid \mathbf{y}, \boldsymbol{\zeta})\Big|_{\boldsymbol{\xi}=\hat{\boldsymbol{\xi}}_{k-1}, \boldsymbol{\zeta}=\hat{\boldsymbol{\zeta}}_{k-1}} \\ &= \sum_{i=1}^{M} \frac{\sum_{j=1}^{M} \frac{\partial}{\partial \boldsymbol{\xi}} K(\epsilon(i) - \epsilon(j))}{\sum_{k=1}^{M} K(\epsilon(i) - \epsilon(k))}\Bigg|_{\boldsymbol{\xi}=\hat{\boldsymbol{\xi}}_{k-1}, \boldsymbol{\zeta}=\hat{\boldsymbol{\zeta}}_{k-1}} \end{aligned} \tag{5.16}$$

Similarly gradient for $\hat{\boldsymbol{\zeta}}$:

$$\begin{aligned}\nabla_{\boldsymbol{\zeta}} \hat{\mathcal{L}}(\boldsymbol{\zeta} \mid \mathbf{y}, \boldsymbol{\xi})\mid_{\boldsymbol{\zeta}=\hat{\boldsymbol{\zeta}}_{k-1}, \boldsymbol{\xi}=\hat{\boldsymbol{\xi}}_{k-1}} &= \frac{\partial}{\partial \boldsymbol{\zeta}} \hat{\mathcal{L}}(\boldsymbol{\zeta} \mid \mathbf{y}, \boldsymbol{\xi})\Big|_{\boldsymbol{\zeta}=\hat{\boldsymbol{\zeta}}_{k-1}, \boldsymbol{\xi}=\hat{\boldsymbol{\xi}}_{k-1}} \\ &= \sum_{i=1}^{M} \frac{\sum_{j=1}^{M} \frac{\partial}{\partial \boldsymbol{\zeta}} K(\epsilon(i) - \epsilon(j))}{\sum_{k=1}^{M} K(\epsilon(i) - \epsilon(k))}\Bigg|_{\boldsymbol{\zeta}=\hat{\boldsymbol{\zeta}}_{k-1}, \boldsymbol{\xi}=\hat{\boldsymbol{\xi}}_{k-1}} \end{aligned} \tag{5.17}$$

Thereby substituting the estimated gradients in eq. (5.12) and (5.13) respectively, and iterating till $\hat{\boldsymbol{\xi}}_k$ and $\hat{\boldsymbol{\zeta}}_k$ converge. After convergence the ML estimate of the channel $\mathbf{h}$ is obtained by deconvolution. The algorithm is initialised by the LS estimate and $\mu(k)$ is chosen as explained

in [82] and [87] respectively. During simulations the algorithm did not converge to a local maxima, however this is not always guaranteed (as explained in previous chapter).

Two possible update methods can be used here to maximise the joint likelihood. In the first method, the $\hat{\xi}$ is updated followed by $\hat{\zeta}$ and then the likelihood is estimated for the next update. The procedure is repeated till both $\hat{\xi}$ and $\hat{\zeta}$ converge. Second update method is to first have $\hat{\xi}$ converged and then have $\hat{\zeta}$ converged given the converged $\hat{\xi}$ on estimated likelihood, and then iterate till no significant changes are observed in $\hat{\xi}$ and $\hat{\zeta}$. In this chapter, the first method of update is used.

5.3 Simulation results

For simulation study a communication channel model, similar to GSM, considering CCI with Gaussian noise as a multi-modal, i.i.d., Gaussian mixture interference as discussed in [82] is used. The performance of the channel estimator is calculated by NMSE as shown in eq. (4.30). For all simulation results, the input symbols of length 100 and ensemble of 1000-runs are considered.

A typical communication system affected by co-channel interference is shown in previous chapter's Figure 4.10. The system representation in eq. (4.31)-(4.33) is still valid for the simulations in this chapter. The above presented algorithm is verified for real stationary channel for $N_T = 5$, and the LPE filter of order 2 is assumed. The input signal is anti-podal random input sequence. The channels are assumed to be

h = [-0.227 0.460 0.688 0.460 -0.227] and

g_0 = [0.1 0.3 0.8 0.3 0.1] where h suffers from amplitude and phase distortion [20], and g_0 is the co-channel considered for the simulation.

Two different sets of simulations are performed to verify the robustness of the proposed algorithm. First Figure 5.2 depicts the SNR performance plot for the channel estimator presented in this thesis, where the SIR is fixed at 5dBs. The legends 'LS', 'LS_{white}', 'NPML' and '$\text{NPML}_{\text{white}}$' represent LS without LPE filter, LS with LPE filter, NPML without LPE filter and NPML with LPE filter respectively. It is observed that the $\text{NPML}_{\text{white}}$ performs better than other proposed techniques. There is a considerable gain at high SNRs by whitening the noise and interference before doing the channel estimation. A little performance gain can be achieved

by using the traditional LS technique with LPE filter.

Secondly, the Figure 5.3 represents the performance plots when the channel estimation is performed with SNR fixed at 30dBs and interference power varied. As anticipated, the proposed algorithm performs better than other techniques in high interference channel conditions. Considerable power gains can be achieved at low SIRs. Thus it would be appropriate to conclude that the NPML algorithm with LPE filter provides better channel estimates in high SNR and low SIR environments, typical of indoor wireless communications channel.

5.4 Conclusion

It was shown that the channel estimator with whitening filter forms a robust channel estimator for correlated non-Gaussian noise (or CCI) channel. It was reconfirmed that the LS estimate with LPE filter produces better channel estimates for interference limited channels than LS estimate without LPE filter. Due to practical constraints, the Gaussian assumption on the whitened noise is not guaranteed, hence a kernel density estimate based ML channel estimator was proposed. From Figure 5.2 and Figure 5.3 it is observed that better channel estimates can be obtained by jointly estimating the whitening filter and the channel estimates by using the kernel density estimator. Thus by combining kernel density estimator with whitening filter forms a robust channel estimator for interference limited communication channels.

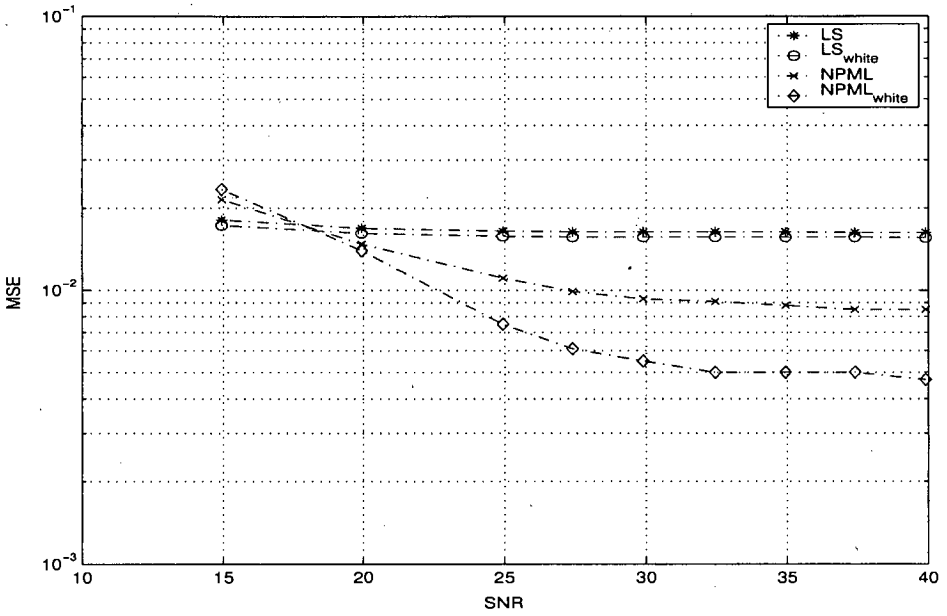


Figure 5.2: *MSE plot for co-channel affected communication system where h = [-0.227 0.460 0.688 0.460 -0.227], SIR=5dBs for 100-symbols over an ensemble of 1000-runs*

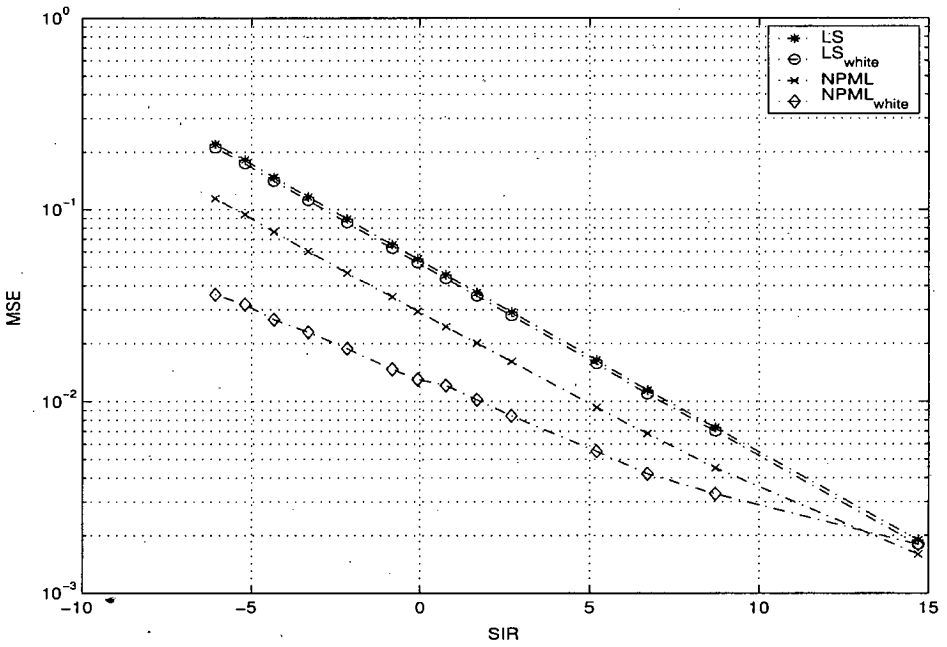


Figure 5.3: *MSE plot for co-channel affected communication system where h = [-0.227 0.460 0.688 0.460 -0.227], SNR=30dBs for 100-symbols over an ensemble of 1000-runs*

Chapter 6
Non-parametric maximum likelihood channel estimator and equaliser for OFDM systems

Orthogonal frequency division multiplexing is a promising multi-carrier digital communication technique for transmitting data at high bit-rates over wireless or wireline channels. The high-speed serial data are converted into many low bit rate streams that are transmitted in parallel, thereby increasing the symbol duration and reducing the ISI. These features have led to an increase in the use of OFDM or related techniques in many high bit rate communication systems. Discrete multi-tone modulation which is quite similar to OFDM is extensively used in DSL communication systems. OFDM has been chosen for digital audio broadcasting (DAB) and digital video broadcasting (DVB). It is also used for the 2.4 GHz wireless local area networks (WLAN) i.e. IEEE 802.11g and worldwide interoperability for microwave acccess (WiMAX) i.e. IEEE 802.16.

Coherent OFDM transmission invariable requires estimation of the channel frequency response (i.e. the gains of the OFDM tones). Currently there can be three possible solution: 1) blind, 2) semi-blind, and 3) pilot aided. In blind channel estimation techniques, the channel is estimated without the knowledge of the transmitted sequence. It is attractive as the throughput is higher as no bits are lost in training. However it requires large amount of data to be stored before channel estimation can begin, which invariably introduces delays. The pilot based technique estimates the channel by transmitting a known (at the receiver) training sequence along with the unknown data at the receiver. The receiver estimates the channel using some criterion based on comparing the change in these pilots due to channel. The semi-blind techniques try to reduce the size of the training sequence by exploiting both the known and the unknown (blind) portions of the data.

Channel estimation in OFDM is critical to the overall performance of the communication system. The insertion of pilots in OFDM symbols provides a base for reliable channel estimates.

There has been considerable increase in channel estimation research over the years [102], [103], [104] to name a few. However most of the current work is based on channel estimation for Gaussian channels or assuming that the interference is very low. This assumption is however not always valid in scenarios where there are a small number of interferers (e.g. Bluetooth device [105] or microwave oven operating in presence of a WLAN). With the co-existence of various wireless equipment in home and office environments the interference from neighbouring devices has become a major concern [106]. In interference affected channels the algorithms designed for Gaussian assumption are not optimal [42] [87]. As in the previous chapters, the traditional Gaussian assumption channel estimator (which assumes zero or negligible interference) is refered to as an LS channel estimator.

The algorithm discussed in [44] specifically deals with synchronous interference and after modifications for asynchronous interference [107]. However in [44], [107] and [84] it was noted that interference was modelled as Gaussian, which may not be the case if only a few or infact one major interferer as in [106] [105] are present. In this chapter no such *a priori* assumption on the distribution of the interfering received signal is made. Moreover no individual parameter of the interferer is estimated specifically. The channel is assumed to be stationary for a given block. In this chapter the fading channel is estimated in the presence of interference directly in time domain using an ML technique. In fact, the presence of interference along with Gaussian noise is jointly considered as a Gaussian mixture noise [87] and [82]. In this chapter it is noted that traditional zero forcing equalisation (usually used in OFDM receivers) technique falls short of the performance in the presence of interference. Thus a MAP symbol-by-symbol equaliser to improve the BER performance is proposed. Simulation results confirm the non-optimal estimates when LS is used and improved BER performance by using the proposed channel estimator and equaliser algorithms.

The chapter is organised as follows: section 6.1 the problem statement is formulated for a general OFDM communication system. A brief discussion on kernel density estimator for complex density estimation is described in section 6.2. The iterative NPML channel estimator for OFDM is described in section 6.3. Section 6.4 discusses the modified non-parametric symbol-by-symbol equaliser. To test the robustness of the algorithm, in section 6.5 the simulations results are presented for both flat and multi-path fading. Conclusions based on analysis and simulation are drawn in section 6.6.

6.1 Formulation of the problem

6.1.1 OFDM system model

The baseband equivalent representation of a typical OFDM system as in Figure 6.1 is considered here. The discussion on the estimation of one OFDM symbol instead of a sequence of symbols is justified below. At the transmitter side, the serial input data is converted into M parallel streams, and each data stream is modulated by a linear modulation scheme, such as QPSK, 16QAM or 64QAM. If QPSK is used, for instance, the binary input data of $2M$ bits will be converted into M QPSK symbols by the serial-to-parallel converter (S/P) and the modulator. The modulated data symbols, which are denoted by complex-valued variables $X(0), ..., X(m), ..., X(M-1)$, are then transformed by the inverse fast Fourier transform (IFFT), and the complex-valued outputs $x(0), ..., x(k), ..., x(M-1)$ are converted back to serial data for transmission. A guard interval is inserted between symbols to avoid ISI. If the guard interval is longer than the channel delay spread, and if the samples of the guard at the receiving end are discarded, the ISI will not affect the actual OFDM symbol. Therefore, the system can be analysed on a symbol-by-symbol basis. OFDM system is also exhibits high peak-to-average poer ratio (PAPR). Namely, the peak values of some of the transmitted signals could be much larger than the typical values. This could lead to a necessity of suing circuits with linear characteristics within a larger dynamic range, otherwise the signal clipping at high levels would yield a distortion of the transmitted signal and out-of-band radiations. In this chapter, it is assumed that the received OFDM signal is not affected by PAPR problem. At the receiver side, after converting the serial data to M parallel streams, the received samples $y(0), ..., y(k), ..., y(M-1)$ are transformed by the FFT into $Y(0), ..., Y(m), ..., Y(M-1)$ [102].

Using the notations for the OFDM symbols, the output of the channel can be written as

$$y(k) = \sum_{l=0}^{N_T-1} h^*(l)x(k-l) + \sum_{p=0}^{P_I-1}\sum_{l=0}^{N_T-1} g_p^*(l)u_p(k-l) + n(k), 0 \leq k \leq M-1 \quad (6.1)$$

where h and x represents desired user's channel and data respectively. Without loss of generality, the complex conjugate h^* instead of h is chosen in the above equation [16]. N_T represents the channel length and $n(k)$ is the i.i.d. additive white Gaussian noise. P_I represents the number of interferers where g_p and u_p are the interfering channel and signal respectively. Note that $y(k)$, $x(k)$, $n(k)$, $h(l)$, $u_p(k)$ and $g_p(l)$ are all complex valued. It is assumed that the channel

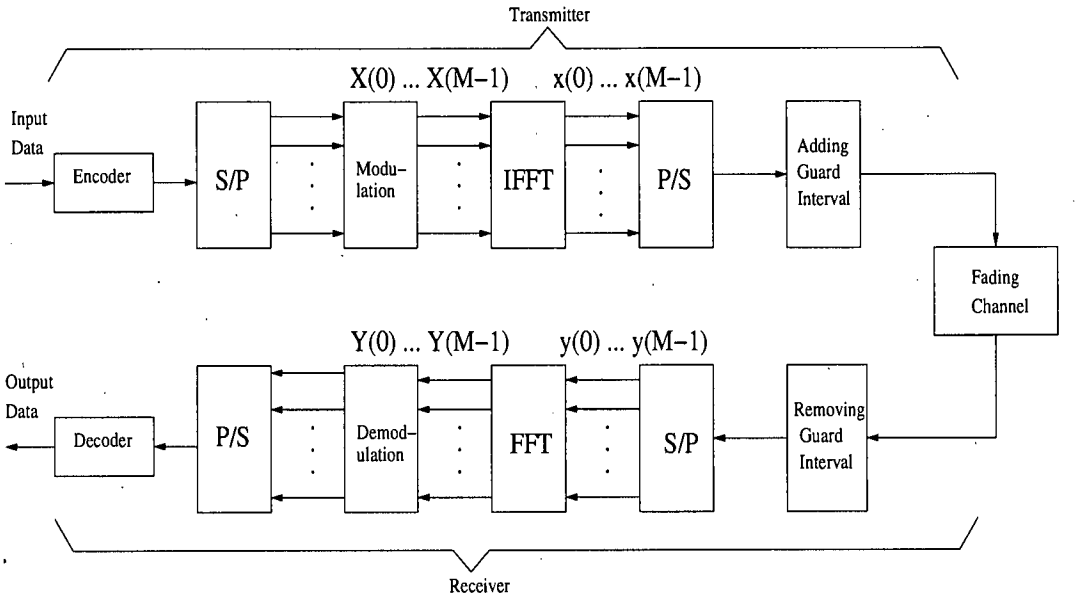


Figure 6.1: *A typical OFDM communication system*

and interference doesn't change during the block transfer (assuming quasi-stationary channel) and interference is synchronous which makes the above representation possible.

To maintain orthogonality between tones, it is necessary to ensure that the symbol time contains one or multiple cycles of each sinusoidal tone waveform. This is normally the case, because the system numerology is constructed such that tone frequencies are integer multiples of the symbol period. In absolute terms, to generate a pure sinusoidal tone requires the signal to start at time minus infinity. This is important, because tones are the only waveform than can ensure orthogonality. Fortunately, the channel response can be treated as finite, because multipath components decay over time and the channel is effectively band-limited. By adding a guard time, called a cyclic prefix, the channel can be made to behave as if the transmitted waveforms were from time minus infinite, and thus ensure orthogonality, which essentially prevents one subcarrier from interfering with another (called intercarrier interference, or ICI). Figure 6.2 shows three tones over a single symbol period, where each tone has an integer number of cycles during the symbol.

The cyclic prefix is actually a copy of the last portion of the data symbol appended to the front of the symbol during the guard interval. Multipath causes tones and delayed replicas of tones to arrive at the receiver with some delay spread. This leads to misalignment between sinusoids,

which need to be aligned to be orthogonal. The cyclic prefix allows the tones to be realigned at the receiver, thus regaining orthogonality. As discussed above, if a cyclic prefix is used for the

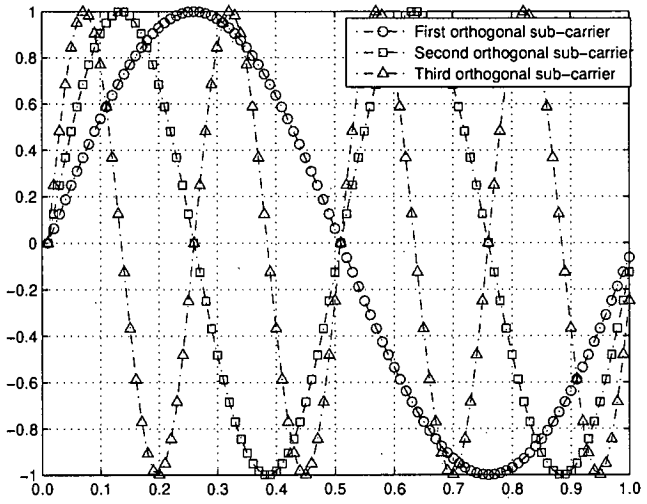


Figure 6.2: *Integer number of sinusoid periods*

guard interval, intercarrier interference (ICI) in a multipath channel can also be avoided. Then it can be shown that the following simple relation between $Y(m)$ and $X(m)$ holds:

$$\begin{aligned} Y(m) &= \left(\sum_{l=0}^{N_T-1} h^*(l)\exp^{(-j2\pi\frac{ml}{M})}\right)X(m) \\ &\quad + \left(\sum_{p=0}^{P_I-1}\sum_{l=0}^{N_T-1} g_p^*(l)\exp^{(-j2\pi\frac{ml}{M})}U_p(m)\right) + N(m) \qquad (6.2) \\ &= H(m)X(m) + I(m) + N(m), 0 \leq m \leq M-1 \qquad (6.3) \\ &= H(m)X(m) + N'(m), 0 \leq m \leq M-1 \qquad (6.4) \end{aligned}$$

where $H(m)$ is the complex frequency response of the channel at the subchannel m, $I(m)$ be the complex interference at that subchannel m and $N(0), ..., N(M-1)$ are the discrete Fourier transform (DFT) of $n(0), ..., n(M-1)$. From [42]-example 15.3 it is observed that DFT of i.i.d. Gaussian $n(0), ..., n(M-1)$ are i.i.d Gaussian as well. It is assumed that the interfering signals $U_p(0), ..., U_p(M-1)$ are also OFDM signals, with same block and cyclic pre-fix lengths, and they are block synchronous with the desired signal. The system is depicted in Figure 6.3. Eq. (6.4) shows that the received signal is the transmitted signal attenuated and phase shifted by the

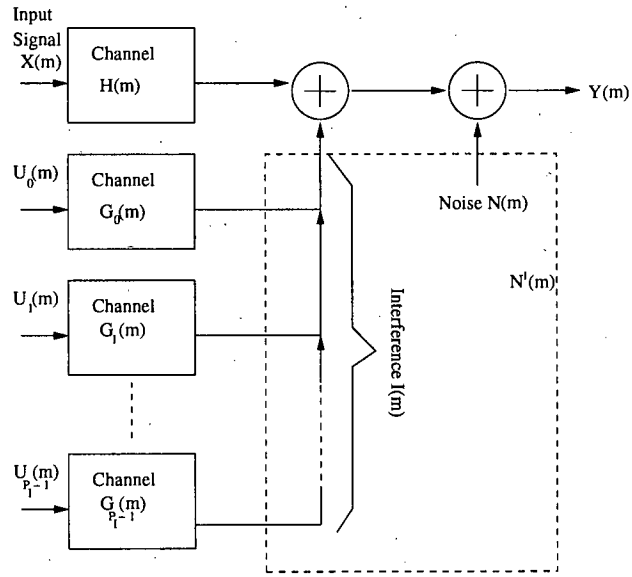


Figure 6.3: *OFDM communication system in presence of interference and noise*

frequency response of the channel at the subchannel frequencies due to fading in the presence of interference and noise [102]. It is assumed to be that noise is represented as complex i.i.d. with vector $\mathbf{n} = [n(0), n(1), ..., n(M-1)]^T$ with each component of $\mathbf{n}$ distributed as $\mathbb{C}\mathcal{N}(\mu_i, \sigma_{ni}^2)$ and are also independent. The multivariate complex Gaussian pdf is just the product of the marginal pdfs or

$$f(\mathbf{n}) = \prod_{i=0}^{M-1} f(n(i)) \tag{6.5}$$

which follows from the usual property of the pdf for real independent random variables, this can be written as

$$f(\mathbf{n}) = \frac{1}{\pi^M \prod_{i=0}^{M-1} \sigma_{ni}^2} \exp\left[-\sum_{i=0}^{M-1} \frac{1}{\sigma_{ni}^2} \mid n(i) \mid^2\right] \tag{6.6}$$

Since the joint pdf depends on $\Re(n)$ and $\Im(n)$ (where $\Re$ and $\Im$ stands for real and imaginary components of a complex variable) only through $\mathbf{n}$, the joint pdf can be viewed to be that of the 'scalar random variable n'. This pdf eq. (6.6) is called a 'complex Gaussian pdf' for a scalar complex random variable and is denoted by $\mathbb{C}\mathcal{N}(0, \sigma_{ni}^2)$ [42] (without loss of generality noise is assumed to be zero mean).

6.1.2 Asynchronous interferer

Asynchronous interference occurs when the cyclic prefix of the interferer does not align with the desired user's cyclic prefix. The receiver is usually synchronised with the desired user's transmitter. For an asynchronous interferer in a rich multipath environment, the received frequency-domain measurement is highly correlated. The frequency offset causes the loss of orthogonality and causes ICI in the interferer [108]. In many cases the algorithms designed for synchronous interference fail when the asynchronous interference is encountered. The algorithm discussed later in this chapter does not require any change when asynchronous interference is encountered.

The interference model described in eq. (6.4) is not valid for asynchronous interference. The interference in eq. (6.6) is now modelled as:

$$I(m) = \sum_{p=0}^{P_I-1} \sum_{k=0}^{M-1} U_p(k) G_p(k) \frac{\sin(\pi(k-m+\triangle F))}{M \sin(\pi(k-m+\triangle F)/M)} \exp\left(\frac{j\pi(k-m+\triangle F)(M-1)}{M}\right) \tag{6.7}$$

where U_p, G_p represent the p^{th} interferer's input signal and channel respectively. $\triangle F$ represents the (normalised) frequency offset [108] [109] [110], which is normalised by the subcarrier spacing which is roughly $\triangle F \approx f_{OFDM}/M$, where f_{OFDM} is the total occupied bandwidth by OFDM.

6.2 Kernel density estimation

Since both the noise and interference are complex, they can be modelled by a 'complex Gaussian mixture' pdf, where the real and imaginary components are assumed independent as discussed earlier. Parzen window or kernel density estimation assumes that the probability density is a smoothed version of the empirical sample. Its estimate $\hat{f}(y)$ of a complex random variable $y = \Re\{y\} + i\Im\{y\}$ is simply the average of radial kernel functions centered on the points in a sample M of the instance of y:

$$\hat{f}(y) = \frac{1}{M} \sum_{j=1}^{M} K(y - y(j)) \tag{6.8}$$

As in previous chapters, $K(.)$ is assumed to be Gaussian kernel (Parzen kernel) [82]:

$$K(y) = \mathcal{N}(0, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(\frac{-\mid y \mid^2}{2\sigma^2}\right) \tag{6.9}$$

variance defined as σ^2. The joint pdf $\hat{f}(y)$ depends on the real and complex components through y, the pdf can be viewed as that of the scalar random variable y, as the notation suggest [42].

6.3 Non-parametric ML channel estimation

Performance of the OFDM system depends highly on the quality of the channel estimate, this makes the channel estimator one of the most useful part of a OFDM communication receiver. Due to increased bit-rate performance of the OFDM systems, there has been considerable interest in developing better channel estimators for various channel conditions. However, for Gaussian noise channels a simple technique based on LS (also known as zero-forcing estimator) [111] is usually used. The LS channel estimate is formed as,

$$H(m) = X(m)^{-1}Y(m), \quad 0 \leq m \leq M-1 \tag{6.10}$$

for a training symbol, i.e. $X(m)$ is considered known at the receiver. There are various techniques for channel estimation in flat and Rayleigh fading channels ([104], [112], [113], [114] to name a few), however most of these techniques assume presence of Gaussian noise. The algorithm discussed in [44] can be used for channel estimation in the presence of interference, however it assumes interferers as Gaussian distributed. In this section, no assumption on the distribution of the interference is made, which makes this technique robust for both synchronous and asynchronous interference.

The typical channel impulse response components $\mathbf{h} = [h(0), ..., h(N_T - 1)]$ are independent complex-valued Gaussian random variables (which represents a frequency-selective Rayleigh fading channel). In a regular OFDM system, the channel delay spread N_T is much smaller than the number of subcarriers M, which leads to a high correlation among the channel frequency responses $H(m)$, even when $h(l) \forall l$, are independent [114]. Thus the channel impulse response $\mathbf{h} = [h(0), ..., h(N_T - 1)]$ is estimated directly as they are independently specified and the number of parameters in the time domain is smaller than that in the frequency domain [102].

The combined interference and AWGN $N'(m)$ in eq. (6.4) is together taken as a noise that is non-Gaussian because of the presence of interference [87] [82]. As also discussed in [87] [82] the least squares estimator does not find the optimal solution in the case of non-Gaussian noise. The mean square error criterion which minimises the energy between the desired and the system output does not converge to the optimal solution. Also the least squares estimator is not minimal sufficient statistics (refer Appendix-B), and hence the ML estimator is used for this Gaussian mixture formulation.

As discussed in previous chapters, if the noise is Gaussian then the LS estimate leads to the ML estimate. However, in communication systems where the noise is non-Gaussian (or approximated as Gaussian mixture) then no closed form ML solution exists for such non-Gaussian distributions. Thus an iterative algorithm to find the ML estimate of the channel is used. The algorithm is initialised with the LS channel estimate based on the pilots. The first likelihood estimate is obtained by using the LS channel taps. After estimating this likelihood, the ML solution is sought iteratively exploiting the pilot symbols. The classical stochastic gradient algorithm is used with log-likelihood being the cost function i.e. the gradient here is the first derivative of the log-likelihood function with a constant multiplier (similar to the well known gradient ascent algorithm) [115]. The update equation is represented as:

$$\hat{\mathbf{h}}_k = \hat{\mathbf{h}}_{k-1} + \mu(k)\,\nabla_{\mathbf{h}}\,\mathcal{L}(\mathbf{h} \mid \mathbf{Y})|_{\mathbf{h}=\hat{\mathbf{h}}_{k-1}} \tag{6.11}$$

where $\mu(k)$ is the adaptation constant and $\nabla_{\mathbf{h}}$ represents the gradient of the cost function. Referring to eq. (6.4) and eq. (6.11) the likelihood function can be written as:

$$L(\mathbf{h} \mid \mathbf{Y})|_{\mathbf{h}=\hat{\mathbf{h}}_{k-1}} = f(\mathbf{Y} \mid \mathbf{h}) = \prod_{i=1}^{M} f_{N'}(E(i))$$

where $f_{N'}(.)$ is scalar pdf of 'complex Gaussian mixture' of data length from $i = 1, ..., M$ and the previous estimation error is defined as:

$$E(i) = Y(i) - \left(\sum_{l=0}^{N_T-1} h_k^*(l) \exp^{(-j2\pi \frac{il}{M})} \right) X(i) \tag{6.12}$$

Kernel density estimators are known to be effective in estimating the pdf over short data record and also provide a differentiable smooth estimated pdf. From the definition of kernel density

estimator [92]:

$$\hat{f}_{N'}(E) = \frac{1}{M}\sum_{j=1}^{M} K(E - E(j)) \tag{6.13}$$

where M is the number of subcarriers. The log-likelihood function can then be written as:

$$\begin{aligned}\hat{\mathcal{L}}(\mathbf{h} \mid \mathbf{Y})\Big|_{\mathbf{h}=\hat{\mathbf{h}}_{k-1}} &= \sum_{i=1}^{M} \log\left(f_{N'}(E(i))\right) &(6.14)\\ &= \sum_{i=1}^{M} \log\left(\frac{1}{M}\sum_{j=1}^{M} K(E(i) - E(j))\right)\\ &= \sum_{i=1}^{M} \log \sum_{j=1}^{M} K(E(i) - E(j)) - \log M &(6.15)\end{aligned}$$

Maximising the log-likelihood function w.r.t the channel weight vector. By definition of complex vector differentiation [16],

$$\nabla_{\mathbf{h}}\, \hat{\mathcal{L}}(\mathbf{h} \mid \mathbf{Y})\Big|_{\mathbf{h}=\hat{\mathbf{h}}_{k-1}} = \frac{\partial \hat{\mathcal{L}}(\mathbf{h} \mid \mathbf{Y})}{\partial \mathbf{h}}\Big|_{\mathbf{h}=\hat{\mathbf{h}}_{k-1}} = \sum_{i=1}^{M} \frac{\sum_{j=1}^{M} \frac{\partial K(E(i)-E(j))}{\partial \mathbf{h}}}{\sum_{j=1}^{M} K(E(i) - E(j))}\Big|_{\mathbf{h}=\hat{\mathbf{h}}_{k-1}} \tag{6.16}$$

Thereby substituting this gradient in eq. (6.11) gives an iterative solution. As with any stochastic gradient algorithm the choice of optimal $\mu(k)$ varies with the application and requirements. As discussed in Appendix-A, $\mu(k) = \frac{\sigma^2}{M}$ is chosen in eq. (6.11) and convergence is witnessed in a few iterations. The σ is chosen as $\sigma = (1/M)^{(1/6)}\sigma_n$, as found analytically in [82] for estimating complex Gaussian distribution.

6.4 Non-parametric symbol-by-symbol MAP equaliser

Once the frequency-domain channel response $\hat{H}$ is found, the estimate of the transmitted signal can be obtained by solving

$$\hat{X}(m) = \arg\min_{X \in C} \mid Y(m) - \hat{H}(m)X(m) \mid^2, \quad 0 \leq m \leq M-1 \tag{6.17}$$

which leads to the final estimate of the transmitted signals (assuming $X \in C$) as follows:

$$\hat{X}(m) = \text{Quantisation}\left\{\frac{Y(m)}{\hat{H}(m)}\right\}, \quad 0 \leq m \leq M-1 \tag{6.18}$$

The above formulation is equivalent to the zero-forcing equaliser and is optimal for Gaussian noise assumption. This forms a popular 'one-tap equaliser' used in many practical OFDM communication systems.

Similar to the channel estimator discussed before, the conventional detector (equaliser [102]) is not optimal for the interference affected channels. The performance of this zero-forcing equaliser [102] is highly sensitive to the quality of estimated channel and the ratio of interfering received signal with estimated channel, in addition to typical constraints of a zero-forcing equaliser.

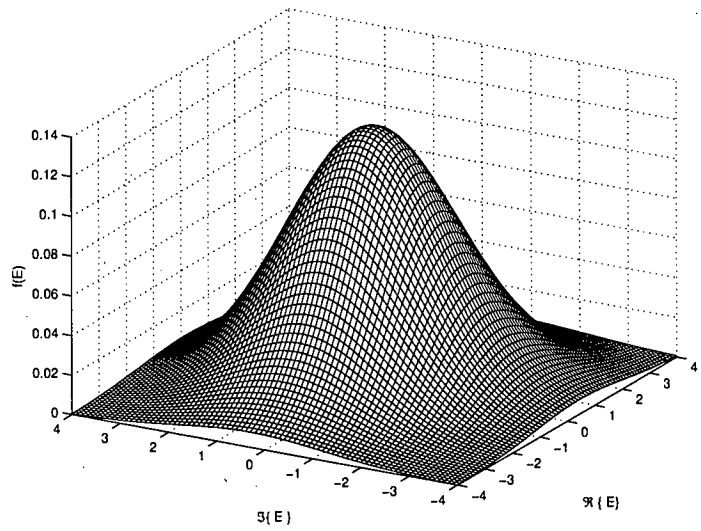
$$\hat{X}(m) = \text{Quantisation}\left\{\frac{Y(m)}{\hat{H}(m)}\right\} = \text{Quantisation}\left\{\frac{X(m)H(m) + I(m) + N(m)}{\hat{H}(m)}\right\} \tag{6.19}$$

The impact of imperfect channel estimation in case of Gaussian noise is studied in details in [116]. The decision boundary for the said equaliser is clearly non-linear due to the presence of interference and the imperfect channel estimation. Thus the assumption that the decision boundary being linear (based on Gaussian assumption) is no longer valid. It can be observed from Figure 6.4, that the noise in presence of interference forms a Gaussian mixture. This makes the decision boundary non-linear and as can be observed from eq. 6.19.

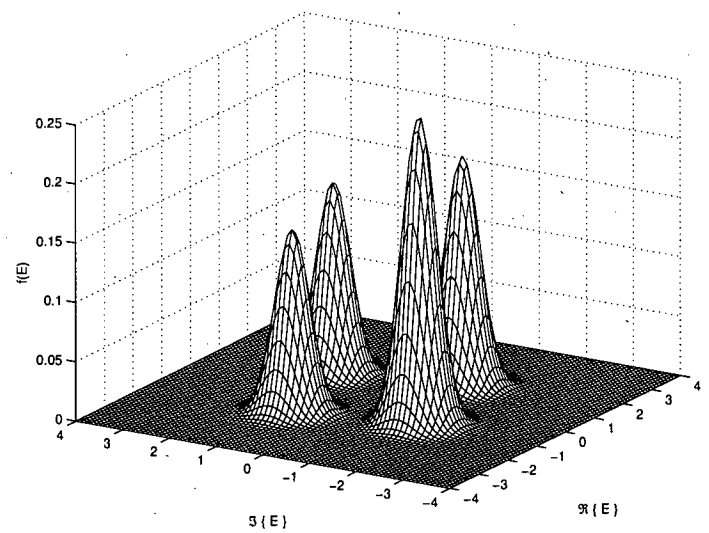
Here a probabilistic symbol-by-symbol MAP equaliser whose decision is based on the estimated likelihood is used. The endevour is to maximise the probability of correct decision for the received symbol i.e. maximise $F(X_c(m) \mid Y(m)); c = 1, ..., C$. The decision criterion is based on selecting the signal corresponding to the maximum of the set of posteriori probabilities $\{F(X_c(m) \mid Y(m))\}$. This decision criterion is called the maximum *a posteriori* probability criterion. Using Bayes' rule, the posterior probability is expressed as:

$$F(X_c(m) \mid Y(m)) = \frac{f(Y(m) \mid X(m))F(X(m))}{f(Y(m))} \tag{6.20}$$

where $f(Y(m) \mid X(m))$ is the conditional pdf of the observed vector given $X_c(m)$, and

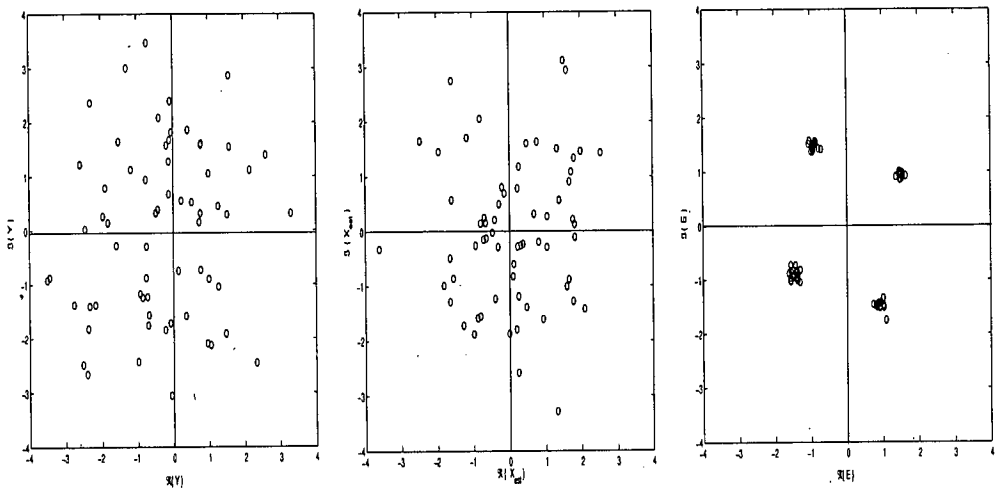


(a) PDF of Difference Signal (E) on Gaussian assumption



(b) PDF of Difference Signal (E) in presence of interference, SNR = 22.45dB, SIR = -1.44dB for synchoronous multi-path fading eq. (6.27)

Figure 6.4: *Distribution plot*



(a) Received signal (Y) for training symbol

(b) Estimated signal (X) for training symbol

(c) Difference signal (E) for training symbol

Figure 6.5: *Signals at various stages of receiver for synchronous multi-path fading channel in presence of synchronous interference*

$F(X_c(m))$ is the *a priori* probability of the c^{th} signal being transmitted. The denominator of eq. (6.20) may be expressed as

$$f(Y(m)) = \sum_{c=1}^{C} f(Y(m) \mid X_c(m))F(X_c(m)) \tag{6.21}$$

From eq. (6.20) and eq. (6.21), it is observed that the computation of the posterior probabilities $F(X_c(m) \mid Y(m))$ requires knowledge of the *a priori* probabilities $F(X_c(m))$ and the conditional pdfs $f(Y(m) \mid X_c(m))$ for $c = 1, ..., C$.

Some simplification in the MAP criterion is possible when the C signals are equally probable *a priori*, i.e., $F(X_c(m)) = 1/C \; \forall c$. Furthermore, the denominator in eq. (6.20) is independent of the transmitted symbol. Consequently the decision rule based on finding the signal that maximises $F(X_c(m) \mid Y(m))$ is equivalent to finding the signal that maximises $f(Y(m) \mid X_c(m))$ [20].

The conditional pdf $f(Y(m) \mid X_c(m))$ is called as the *likelihood function* as seen previously. Thus the decision criterion for the said problem, the MAP detection, reduces to detecting the

ML symbol. For the estimated channel impulse response $\hat{\mathbf{h}}_\mathbf{k}$ (after convergence) from eq. (6.11) and taking its Fourier transform, the ML estimate of the transmitted signal can be obtained by

$$\hat{X}(m) = \arg \max_{X_c \in C} (\hat{f}_E(Y(m) \mid \hat{H}(m))\Big|_{\mathbf{h}=\hat{\mathbf{h}}_k} \tag{6.22}$$

It should be noted that the estimated pdf $\hat{f}_E$ for detection is generated by using eq. (6.13). Based on the higher probability of occurrence the hard-decision (or Quantisation) is taken on $\hat{X}(0), ..., \hat{X}(M-1)$ to generate the output data as shown in Figure 6.1. From simulation results it is observed that significant BER improvement is achieved by using this probabilistic equaliser.

6.5 Simulation results

A packet based OFDM communication system (similar to WLAN) with the first symbol known at the receiver is assumed. The channel estimation is done on this first OFDM symbol, while the remaining payload is the useful information as in Figure 6.6. A typical OFDM communication system is considered as shown in Figure 6.1. The raw binary input data is fed to the rate $1/2$ encoder. The encoder used is specified in [117], where a original $1/3$ encoder is used and then puncturing is performed to form a $1/2$ rate coder. The encoded serial bit stream is then converted to parallel by S/P. The parallel data are then mapped (modulated) to QPSK signal constellation, where it is then passed to IFFT processor. The output of IFFT $x(0), ..., x(M-1)$ is then converted into serial stream by P/S and after appending guard interval (to avoid ISI), the serial stream is then transmitted over a fading channel. It is assumed throughout the chapter that the guard interval is long enough to avoid any ISI and a cyclic prefix is introduced in the guard interval to avoid any ICI. At the receiver, the reverse process as that at the transmitter takes place. First the guard bits are removed followed by serial to parallel converter. Then fast Fourier transform (FFT) is performed on $y(0), ..., y(M-1)$ to obtain $Y(0), ..., Y(M-1)$, which is then demodulated by QPSK demodulator. The demodulated bits are then converted to serial stream and then decoded using the Viterbi decoder [118]. Input and output serial data streams are then compared to calculate the bit error rate.

The robustness of the above proposed adaptive algorithm is verified for two special case of fading channels; flat-fading and multi-path fading. In multi-path fading two special cases of

synchronous and asynchronous interference are considered. The channel is considered to be slowly fading, i.e. the channel remains static (generally defined as quasi-static) for each OFDM packet (of size 64-subcarriers and 8-symbols). Using the formula discussed in [102], a bit rate of 1-Mbps at 1-GHz with vehicle speed of 6-mph (which is sufficient for indoor environment) can be achieved. To verify the robustness of the algorithm, simulations were carried out on Matlab for ensemble of 1000-runs. The performance measure is average BER for fixed SNR and for various values of SIR. The SIR is defined as (eq. 6.3):

$$SIR = \frac{\mathbb{E}\{(HX)(HX)^*\}}{\mathbb{E}\{II^*\}} \tag{6.23}$$

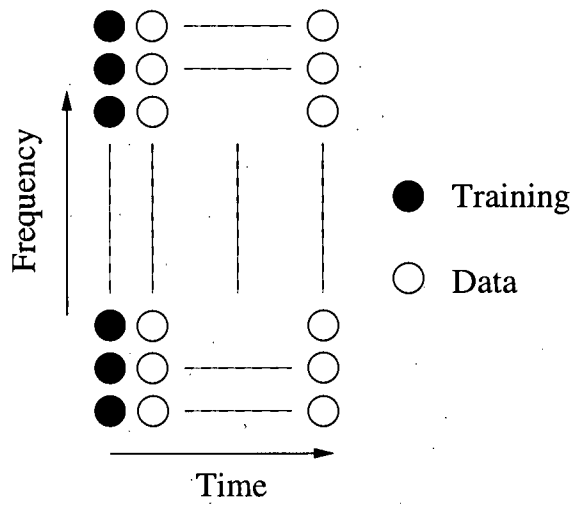


Figure 6.6: *OFDM Packet structure*

6.5.1 Flat fading

The performance evaluation for the proposed estimator and equaliser begins by using a flat-fading channel. Figure 6.7 and Figure 6.8 shows the performance for fixed SNR=16.5dBs and SIR varied for range of values. The channel model is chosen as:

$$H(z) = \alpha_0 \exp(j\theta_0) \tag{6.24}$$

whereas the interfering channel is chosen as:

$$G_0(z) = \alpha_1 \exp(j\theta_1) \tag{6.25}$$

where α_0, α_1 are the i.i.d random variables with Rayleigh distribution, and θ_0, θ_1 are i.i.d. random variables with uniform distribution.

The legend, in Figure 6.7, 'LS channel estimator' and 'NPML channel estimator' represent when the flat fading channel is estimated at the receiver by eq. (6.10) and eq. (6.11) respectively. It is seen that considerable improvement in terms of NMSE is achieved by using the proposed NMPL channel estimation algorithm. The legend, in Figure 6.8, 'Uncoded-LS' refers to the LS estimator for channel estimation and Gaussian assumption MAP equaliser i.e. eq (6.18) is used for detection of symbols following the training (pilot) sequence as in Figure 6.6. 'Uncoded-LS with MAP' refers to the scenario when the channel is estimated by LS estimator followed by a non-parametric equaliser (using this estimated channel and residual error) from eq (6.22). The 'Uncoded-NPML with MAP' refers to the NPML algorithm for channel estimation followed by non-parametric MAP equaliser. The 'Uncoded-Exact with MAP' represents when the exact channel is known at the receiver followed by a non-parametric MAP equaliser. The prefix 'Uncoded' is used when raw binary input is transmitted, whereas prefix 'Coded' is used when the raw binary input data stream is coded by 1/2-rate encoder. It is observed that the algorithm with NPML as channel estimator and non-parameteric symbol-by-symbol MAP equaliser gives the best performance, except in hypothetical case when receiver has *a priori* channel information. Comparing 'LS with MAP' and 'NPML with MAP' for both un-coded and coded input data highlights the performance gain achieved by using NPML channel estimator as compared to the LS estimator (as used in [8]). It is interesting to observe that gains achieved by using NPML decreases when the interference power decreases i.e. at high SIR. It is also observed that the coding for flat-fading doesn't improve the BER performance for the said simulation set-up, which can be attributed to the fact that the channel decoder is based on Gaussian noise assumption. Thus using NPML in conjunction with non-parametric MAP equaliser forms a robust communication receiver for interference limited channels.

6.5.2 Multipath fading

The channel is defined as a two-path Rayleigh fading channel with transfer function [102]:

$$H(z) = 0.8\alpha_0 \exp(j\theta_0) + 0.6\alpha_1 \exp(j\theta_1) z^{-1} \tag{6.26}$$

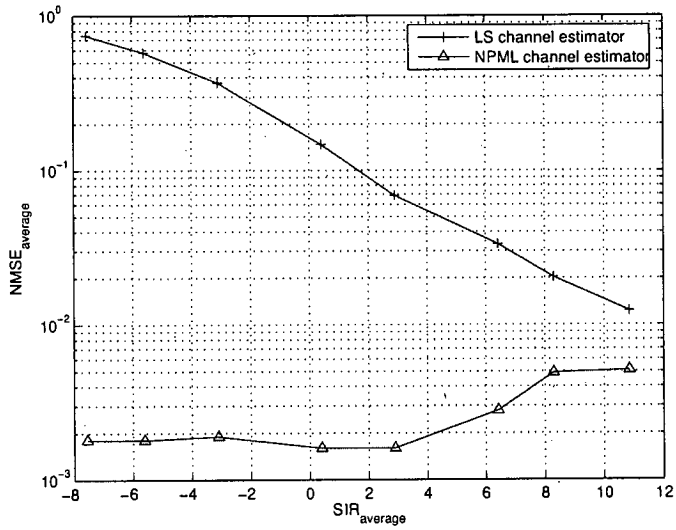


Figure 6.7: *Average NMSE performance in flat fading channel*

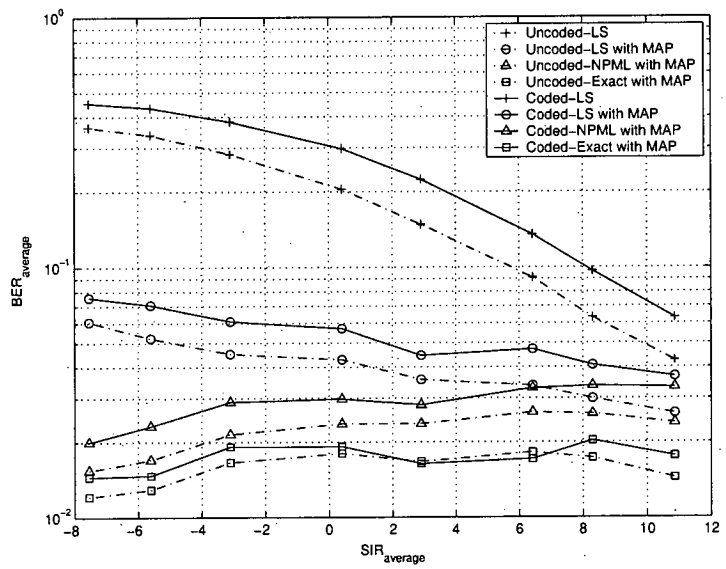


Figure 6.8: *Average BER performance in flat fading channel*

The interfering channel is defined as:

$$G_0(z) = 0.5\alpha_2 \exp(j\theta_2) + 0.1\alpha_3 \exp(j\theta_3)z^{-1} \tag{6.27}$$

where α_0, α_1, α_2, α_3 are the i.i.d random variables with Rayleigh distribution, and θ_0, θ_1, θ_2, θ_3 are i.i.d. random variables with uniform distribution.

6.5.2.1 Synchronous Interference

The synchronous interference model as represented by eq. (6.2) is considered for these set of simulations. The average NMSE plot is shown in Figure 6.9 and the average BER plot is shown in Figure 6.10. The legends define the similar techniques as discussed for flat-fading channel. The SNR is kept fixed at 17.63 dBs while SIR is varied over a large range. The significant BER improvement is obtained by using the non-parametric symbol-by-symbol MAP equaliser as compared to other techniques. By using the NPML channel estimator the performance gain of about 8dBs can be obtained by using NPML channel estimator as compared to LS estimator with non-parametric MAP equaliser estimating density on residual error from both the estimator respectively. It is also worth noting that there is a little difference when this equaliser is used with NPML estimator and exact channel knowledge, this also confirms that the NPML estimates are closer to the exact channel. As observed from the simulation results, the BER gain for NPML based channel estimator and equaliser amplifies for the coded bit stream as compared to the uncoded one. The received, estimated and interfering signal are illustrated in Figure 6.5 for the said simulation environment. It is also interesting to observe that the performance plots follow the pattern as noted in [82].

6.5.2.2 Asynchronous Interference

Figure 6.11 and Figure 6.12 shows the results for the asynchronous interference model as represented by eq. (6.7). The frequency offset is set as $\triangle F = 0.1$ for the results. The SNR is kept fixed at 17.63 dBs while SIR is varied over a large range. There is a huge performance gain by using proposed techniques as compared to the traditional approach of doing channel estimation and equalisation based on Gaussian assumptions. As seen earlier, the NPML estimator with non-parametric symbol-by-symbol MAP equaliser enhances performance instead of using LS estimator and its residual error pdf for MAP equalisation. The receiver's *a priori* knowledge of

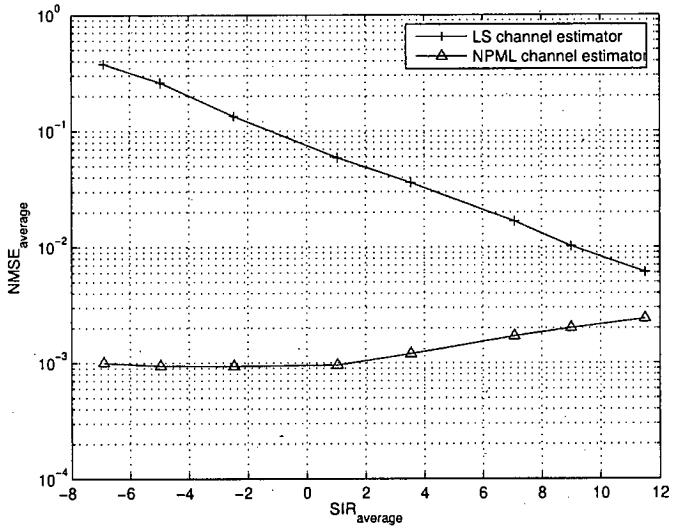


Figure 6.9: *Average NMSE performance in multipath frequency-selective fading channel with synchronous interference*

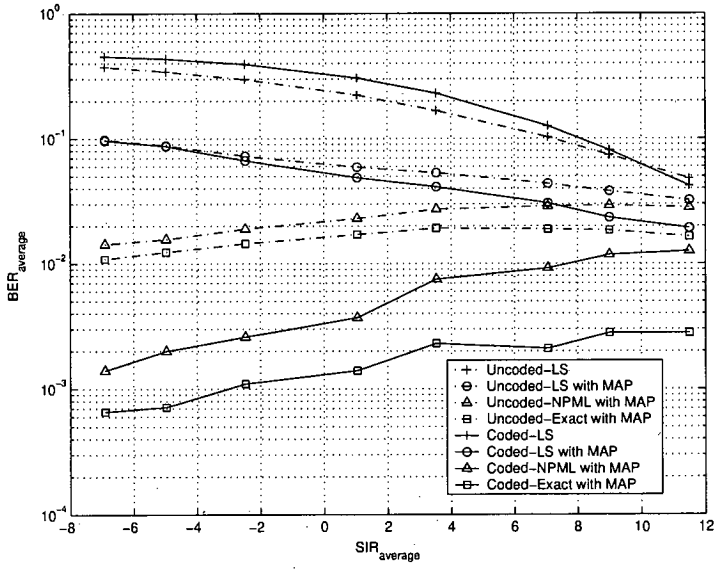


Figure 6.10: *Average BER performance in multipath frequency-selective fading channel with synchronous interference*

the transmission channel forms a lower bound on the performance. It is observed that when SIR increases the Gaussian assumption detector performs better than non-parametric MAP equaliser (except when exact channel is known at the receiver), this is because at high SIR the N′ in eq. (6.4) can be approximated as Gaussian distributed. The asynchronous interference being more correlated than synchronous interference, at low SIR, a considerable difference in performance of 'Coded-NPML with MAP' is observed in both Figure 6.10 and Figure 6.12. However, from simulation results it is safe to conclude that the proposed algorithm and technique could be used for both synchronous and asynchronous interferer without any modification.

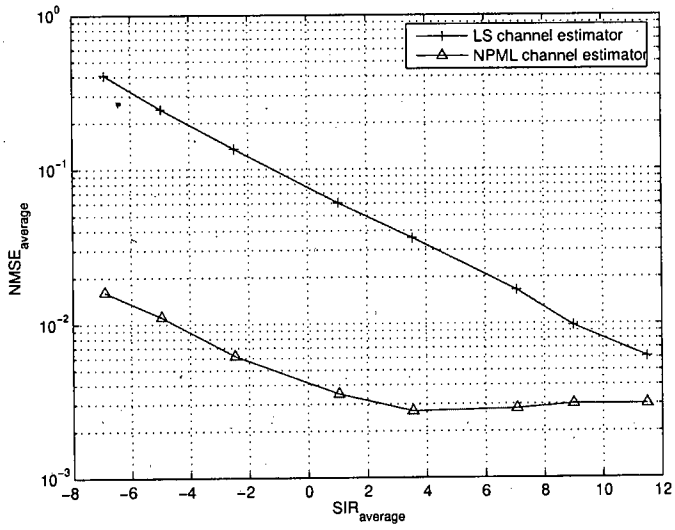


Figure 6.11: *Average NMSE performance in multipath frequency-selective fading channel with asynchronous interference, where* $\triangle F = 0.1$

6.6 Conclusion

It is shown that the channel estimator based on Gaussian noise assumption is inferior in interference affected channels. It is also shown that the traditional zero forcing equaliser produces non-optimal detection in non-Gaussian (interference affected channel), resulting in poor BER performance. The NPML channel estimator and MAP equaliser used together results in improved performance in non-Gaussian noise. This non-Gaussian noise was estimated using kernel density estimator to estimate the likelihood function. It is seen that significant performance gains were achieved for both flat and multipath fading scenarios. The algorithm showed robust-

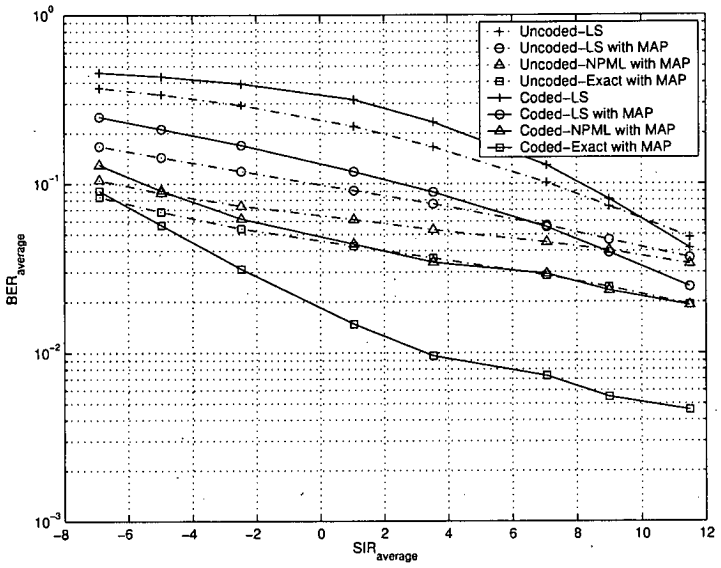


Figure 6.12: *Average BER performance in multipath frequency-selective fading channel with asynchronous interference, where* $\triangle F = 0.1$

ness towards non-Gaussianity from both synchronous and asynchronous interference. It was also highlighted that major performance gain is achieved by using the non-parametric symbol-by-symbol MAP equaliser in interference limited channels.

Chapter 7
Conclusion

The work described in this thesis is primarily concerned with developing adaptive algorithms with the adaptation cost function depending on the distribution. The developed algorithms are applied to two most common applications of adaptive signal processing in digital communications; channel estimation and equalisation. The deviation from the traditional approaches of using second order statistics based cost functions is the motivation for developing these techniques. The developed algorithm is applied to various theoretical models and is also shown to improve performance for practical communication systems. To this end, our analysis and results have shown that the distribution dependent based adaptive learning has better performance than the traditional approaches.

In the next section (section 7.1), the work performed is first summarised and specific achievements accomplished are highlighted. In section 7.2, the limitations of the current work accomplished is discussed and some new directions to future work are proposed.

7.1 Summary and specific achievements of work performed

The work examined in the thesis can be broken down into two major parts. In the first part, chapter 3, the problem of MBER equaliser for alpha stable noise is considered. In the second part, chapters 4, 5 and 6, the non-parameteric maximum-likelihood algorithm is developed, thoroughly analysed and applied to various digital communication systems. The major outline of each chapter is given in the following paragraphs.

The chapter 3 addresses the problem of developing an LMS style decision feedback equaliser algorithm for minimising bit error rate in impulsive noise environments characterised by the alpha stable distribution. The objective of chapter 3 was to develop an MBER DFE equaliser for alpha stable noise environment. First the details about the impulsive noise modelled as alpha stable distribution and various parameters characterising the alpha stable distribution were studied. A brief discussion about the state translated DFE structure was discussed next. The

development exploits the stable nature of the alpha distribution and the concepts build on earlier work [14] [15] in a Gaussian noise environment. By introducing a limiter at the receiver front-end both SNR and Wiener solution can be calculated theoretically and by simulations. Further, a Wiener-filter-with-limiter solution was also presented and used as a performance bench mark. It was shown that for minimum bit error design, the adaptation is a function of the noise density function. The comparison between various adaptive algorithms working in identical channel, noise and DFE structure has been drawn. The LBER-Cauchy and the state trans-Cauchy has faster convergence than the other adaptive algorithms in Cauchy noise environments, which is a special form of α-stable noise. Extensive simulations strongly suggest that the state-translated design for the α-stable noise has better convergence and BER performance than the other algorithms. It is also interesting to observe that the adaptive algorithms based on a Gaussian noise assumption despite slow convergence in impulsive noise environments perform closer to those designed with Cauchy noise assumption. Lastly as expected the LMS algorithm performs poorer that the other algorithms in α-stable noise environments. Observations from performance plots in chapter 3 suggest the MBER algorithms' superior performance with respect to the WSL solution.

The second part of the thesis examines the non-parameteric density estimation based adaptive algorithms for channel estimation and equalisation. The work in chapter 4 primarily deals with the channel estimation algorithm development in non-Gaussian noise scenarios. The problem statement was formulated in the beginning of chapter 4, and was shown that the proposed channel estimator does two major tasks; channel estimation and density estimation. The additive noise was modelled as a generalised Gaussian mixture process. The Gaussian mixture was used to generate (or model) both uni-modal and multi-modal noise distributions. The uni-modal mixture noise was used to model impulsive noise and the multi-modal mixture noise was used to model co-channel interference. A brief discussion about kernel density estimation was also presented. It was shown that the LS channel estimator does not find the optimal solution in the case of a Gaussian mixture noise and no-closed form solution exists for the Gaussian mixture noise channel estimator. Thus an iterative NPML channel estimator was presented and parallels with MEE criterion were also drawn. The bounds on the step-size selection for the adaptive algorithm were also formed in this chapter (as in Appendix A). With this step size and initialising the algorithm with the LS estimate, the proposed NPML algorithm was shown to be a robust algorithm to various noise conditions. The algorithm was compared with the traditionally used EM algorithm [62] [10] and was shown to have more flexibility than EM algorithm.

The proposed algorithm is robust to ISI, various noise conditions (uni-modal or multi-modal), and correlated or uncorrelated noise. The lower bounds on channel estimation, i.e. CRB, are also presented in this chapter. From simulations it is observed that the proposed algorithm performs close to the CRB. A further addition to the chapter is on channel order adaptation, where the assumption (for channel estimator) of known channel order is also relaxed from the proposed algorithm and the robustness of proposed algorithm to simultaneously adapt the channel weights and the order are discussed in Appendix C.

Chapter 5 extends the algorithm discussed in chapter 4 for CCI case. In this chapter the loss in performance due to the assumption that the mixture noise (AWGN + interference) is uncorrelated is discussed. The fundamental idea behind the whitening filter and its effect on the problem definition (of chapter 4) is discussed. As it is known that the CCI is correlated, the whitenining filter was used to whiten the CCI. After whitening, the tap weights are updated iteratively using the NPML algorithm. The enhanced MSE performance is achieved by using the proposed technique over the traditional Gaussian assumption based techniques [8] [65].

Chapter 6 considers the application of NPML techniques for channel estimation and equalisation. The purpose of the chapter was to introduce and compare the performance of the NPML channel estimator and equaliser with the traditional methods for the OFDM communications systems. The problem definition was formulated and the case of both synchronous and asynchronous interference was discussed. A brief discussion on traditional channel estimator for pilot training based OFDM systems was presented. The NPML channel estimator formulation for OFDM systems was discussed next. It was shown that the decision boundary for the OFDM linear equaliser was no longer linear due to channel estimation imperfections and the "interference estimated channel" ratio effect. Thus a non-parametric symbol-by-symbol MAP equaliser was proposed which uses the density estimated during the channel estimation procedure to make the decision on the received symbol. The simulation results for flat-fading, and multi-path frequency-selective fading for synchronous and asynchronous interference limited channels were also shown at the end.

7.2 Limitations of current work and proposal for future work

This section discusses some of the limitations of the performed work and directions for future research.

One major short coming of the work is to have an optimal kernel width parameter. The performance of the algorithm is dependent on the proper choice of the kernel width, specially when short training sequence is available.

Other limitation of the work is that the proposed NPML technique is based on block based training method. The scenarios for sample (symbol) based adaptation or development of tracking algorithm would be a interesting area to explore. Also the application of the algorithm for semi-blind channel estimation would be interesting to explore. In addition, as with iterative techniques, a method for finding global minima/maxima is warranted.

In this work the performance improvement in single input single output systems was considered. Recently there has been considerable work on multiple input multiple out (MIMO) systems [119] [120]. Thus extending this work for MIMO channels would be a natural next step forward.

Appendix A
Step size calculation

The Hessian $\mathbf{H}$ to get an estimate of the step-size $\mu(k)$, for the gradient ascent NPML algorithm is formulated [121] as:

$$\mu(k) = \frac{-\parallel \nabla_{\mathbf{h}} \mathcal{L}(\mathbf{h} \mid \mathbf{y}) \parallel}{\nabla_{\mathbf{h}} \mathcal{L}(\mathbf{h} \mid \mathbf{y})^t \mathbf{H} \nabla_{\mathbf{h}} \mathcal{L}(\mathbf{h} \mid \mathbf{y})} \tag{A.1}$$

where $\mathbf{H}$ depends on h, and thus indirectly on k. $\mathbf{H}$ is the *Hessian matrix* of second partial derivatives $\frac{\partial^2 \nabla_{\mathbf{h}} \mathcal{L}(\mathbf{h}|\mathbf{y})}{\partial h_u \partial h_v}$. This is then the optimal choice of $\mu(k)$ given the assumptions mentioned [121]. First the diagonal elements of the Hessian are calculated,

$$\frac{\partial^2 \hat{\mathcal{L}}(\mathbf{h} \mid \mathbf{r})}{\partial h_v^2} = \sum_{i=1}^{M} \frac{(\sum_{j=1}^{M} K(e(i)-e(j)))\{\frac{\partial}{\partial h_v^2} \sum_{j=1}^{M} K(e(i)-e(j))\} - (\sum_{j=1}^{M} \frac{\partial}{\partial h_v} K(e(i)-e(j)))^2}{(\sum_{j=1}^{M} K(e(i)-e(j)))^2} \tag{A.2}$$

$$= \sum_{i=1}^{M} \frac{1}{(\sum_{j=1}^{M} K(e(i)-e(j)))^2} \left[\left\{ \sum_{j=1}^{M} K(e(i)-e(j)) \left(\frac{\sum_{j=1}^{M} (x_v(i)-x_v(j))^2 (e(i)-e(j))^2 K(e(i)-e(j))}{4\sigma^4} - \frac{\sum_{j=1}^{M} (x_v(i)-x_v(j))^2 K(e(i)-e(j))}{2\sigma^2} \right) \right\} - \frac{1}{4\sigma^4} \left\{ \sum_{j=1}^{M} K(e(i)-e(j))(e(i)-e(j))(x_v(i)-x_v(j)) \right\}^2 \right] \tag{A.3}$$

$$= \sum_{i=1}^{M} \frac{1}{(\sum_{j=1}^{M} K(e(i)-e(j)))^2} \left[\frac{1}{4\sigma^4} \sum_{j=1}^{M} \sum_{p=1}^{M} K(e(i)-e(j)) K(e(i)-e(p)) \{(x(i)-x(j))(e(i)-e(j)) - (x(i)-x(p))(e(i)-e(p))\}^2 - \frac{1}{2\sigma^2} \sum_{j=1}^{M} K(e(i)-e(j)) \sum_{j=1}^{M} (x_v(i)-x_v(j))^2 K(e(i)-e(j)) \right] \tag{A.4}$$

Based on the assumptions about the system model discussed in chapter 4 and for large M

$$\frac{\partial^2 \hat{\mathcal{L}}(\mathbf{h} \mid \mathbf{r})}{\partial h_v^2} \approx \frac{-1}{2\sigma^2} \sum_{i=1}^{M} \frac{\sum_{j=1}^{M} (x_v(i) - x_v(j))^2 K(e(i) - e(j))}{\sum_{j=1}^{M} K(e(i) - e(j))} \tag{A.5}$$

$$\approx \frac{-M}{\sigma^2} \tag{A.6}$$

This formulation is also supported in [122] where large kernel size allowed the quadratic approximation and in [43] where the Taylor series expansion upto second order was taken as sufficient for channel estimation. Based on similar assumptions, the Hessian reduces to approximately diagonal form for large M, the non-diagonal term becomes insignificant, thus $\mu(k) \approx -\mathbf{H}^{-1}$. Thereby $\mu(k)$ in the channel estimator of eq. (4.7) discussed in this thesis is restricted as:

$$0 < \mu(k) \leq \frac{\sigma^2}{M} \tag{A.7}$$

Also note that eq. (A.6) being negative also ensures that local maximum are being sought by the iterative algorithm.

Appendix B
Minimal sufficient statistics

The maximum likelihood estimator reduces to the least squares estimator when the noise is Gaussian and in the absence of interference. The minimal sufficient statistic is formulated, which allows the greatest data reduction without loss of information when the estimated parameter is formulated and it is verified if the LS estimator forms the minimal sufficient statistics. The Lehmann and Scheffe's method for finding a minimal sufficient statistics for the given problem is used in eq. (6.4). By Lehmann and Scheffe's method, it is assumed that the r.v. Z is i.i.d and distributed as $f_{\mathbf{Y}}(\mathbf{Z} \mid \mathbf{h})$, the likelihood ratio is

$$\frac{L(\mathbf{h} \mid \mathbf{Y})}{L(\mathbf{h} \mid \mathbf{Z})} = \frac{\prod_{i=1}^{M}\{f_{\mathbf{Y}}(Y(i) \mid \mathbf{h})\}}{\prod_{i=1}^{M}\{f_{\mathbf{Y}}(Z(i) \mid \mathbf{h})\}} \tag{B.1}$$

For simplicity, let us assume $P = 1$ i.e. one major interferer and $N_T = 1$. Thus the interfering user's noiseless channel states are $\in \{\pm I\}$.

$$\frac{L(\mathbf{h} \mid \mathbf{Y})}{L(\mathbf{h} \mid \mathbf{Z})} = \frac{\prod_{i=1}^{M}\{\exp\left(\frac{-1}{2\sigma^2}(Y(i) - H(i)X(i) + I(i))^2\right) + \exp\left(\frac{-1}{2\sigma^2}(Y(i) - H(i)X(i) - I(i))^2\right)\}}{\prod_{i=1}^{M}\{\exp\left(\frac{-1}{2\sigma^2}(Z(i) - H(i)X(i) + I(i))^2\right) + \exp\left(\frac{-1}{2\sigma^2}(Z(i) - H(i)X(i) - I(i))^2\right)\}} \tag{B.2}$$

$$\begin{aligned}
&= \left\{ \exp\left(\frac{-1}{2\sigma^2}\sum_{i=1}^{M} \mid Y(i) \mid^2\right) \exp\left(\frac{-1}{2\sigma^2}\sum_{i=1}^{M} \mid H(i)X(i) \mid^2\right) \exp\left(\frac{1}{\sigma^2}\sum_{i=1}^{M} \Re\{H(i)X(i)Y^*(i)\}\right) \right. \\
&\left. \exp\left(\frac{-1}{2\sigma^2} I^*(i)I(i)\right) \prod_{i=1}^{M} \cosh\left(\frac{-1}{\sigma^2}\Re\{I(i)(H(i)X(i) - Y(i))^*\}\right) \right\} \Big/ \\
&\left\{ \exp\left(\frac{-1}{2\sigma^2}\sum_{i=1}^{M} \mid Z(i) \mid^2\right) \exp\left(\frac{-1}{2\sigma^2}\sum_{i=1}^{M} \mid H(i)X(i) \mid^2\right) \exp\left(\frac{1}{\sigma^2}\sum_{i=1}^{M} \Re\{H(i)X(i)Z^*(i)\}\right) \right. \\
&\left. \exp\left(\frac{-1}{2\sigma^2} I^*(i)I(i)\right) \prod_{i=1}^{M} \cosh\left(\frac{-1}{\sigma^2}\Re\{I(i)(H(i)X(i) - Z(i))^*\}\right) \right\}
\end{aligned} \tag{B.3}$$

$$= \exp(\frac{-1}{2\sigma^2}\sum_{i=1}^{M} \mid Y(i) \mid^2 - \sum_{i=1}^{M} \mid Z(i) \mid^2) \exp(\frac{1}{\sigma^2}\sum_{i=1}^{M} \Re\{H(i)X(i)Y^*(i)\} - \sum_{i=1}^{M} \Re\{H(i)X(i)Z^*(i)\})$$

$$\prod_{i=1}^{M} \frac{\cosh(\frac{-1}{\sigma^2}\Re\{I(i)(H(i)X(i) - Y(i))^*\})}{\cosh(\frac{-1}{\sigma^2}\Re\{I(i)(H(i)X(i) - Z(i))^*\})} \tag{B.4}$$

This is independent of $H^*(i)$ iff either $Z = Y$ or all the order statistics match i.e. $y_{(1)} = z_{(1)}, y_{(2)} = z_{(2)}, ..., y_{(M)} = z_{(M)}$. Therefore the minimal sufficient statistics for the Gaussian mixture are the set of order statistics for Y, and hence LS does not form the minimal sufficient statistics for the given case [123].

Appendix C
Channel order adaptation

The proposed non-parameteric maximum likelihood (NPML) channel estimator shows superior performance to the LS estimator in the presence of non-Gaussian noise. The derivation of the NPML estimator assumed perfect knowledge of the channel order, which, however, does not comply with most applications. In this section, first the study of effects of inaccurate order assumption on the NPML estimator is done and it is shown that the traditional order selection criteria like the Akaike's information criterion (AIC) are unreliable to apply for the NPML estimator. Then a simple method is proposed to trace the channel order where the order selection and channel estimation are carried out simultaneously.

Although an old topic, the order estimation remains an incompletely solved problem [124]. The most widely used order estimation criteria are the AIC [125], the Final Prediction Error (FPE) [125] and the Minimum Description Length (MDL) [126], all of which, however, are unreliable to apply when the noise is non-Gaussian. All other order estimation algorithm are evaluated against the above three popular criteria. Usually the criterion indices for the possible orders are calculated, before making the final order selection. This *brute-force* approach demands high computation, impeding its application to on-line systems. Although some approaches (e.g. [127]) can carry out the order selection and channel estimation simultaneously, they are limited to specific applications and hard to be applied to the NPML estimator.

For a channel estimator, the order "under-estimate" is more serious a problem than the order "over-estimate" in terms of performance. Thus in practice, it is usually not necessary, if not impossible, to have a precise order estimate as long as the order is not underestimated, thereby making it possible to use simpler methods to estimate the channel order. Recently, Gong (et al.) proposed a novel variable tap-length adaptive algorithm which can be used to track the channel order on-line [128]. However, based on the symbol-based adaptive algorithm such as the LMS algorithm, the proposed algorithm cannot be used for the NPML estimator which is block-based.

In this appendix, first the influences of the inaccurate order assumption on the NPML channel

estimator is investigated. Then, after showing that the classic AIC criterion is unreliable to apply in presence of non-Gaussian noise, a simple method to search for the channel order where the order selection and channel estimation can be carried out simultaneously is proposed. Simulation results are presented at the end.

C.1 Non-parametric ML channel estimator

p	3	4	5	6	7	8	9	10	11
NMSE	0.0961	0.0215	**0.0018**	0.0023	0.0029	0.0034	0.0038	0.0047	0.0055

Table C.1: *NMSE for different assumed channel order.*

According to Figure 4.1, and assuming M as the total number of samples and N_T as the true channel order, the channel output vector can be expressed as:

$$\mathbf{y} = \mathbf{X}_{N_T}\mathbf{h}_{N_T} + \mathbf{w}, \tag{C.1}$$

where $\mathbf{h}_{N_T}$ is the channel vector, $\mathbf{w}$ is the noise vector, and $\mathbf{X}_{N_T}$ is the channel input matrix which is given by:

$$\mathbf{X}_{N_T} = \begin{bmatrix} x(1) & 0 & 0\cdots & 0 \\ x(2) & x(1) & 0\cdots & 0 \\ \vdots & \vdots & & \vdots \\ x(M) & x(M-1) & \cdots & x(M-N_T+1) \end{bmatrix} \tag{C.2}$$

The ML estimator maximises the log-likelihood function

$$\mathcal{L}(\mathbf{h} \mid \mathbf{y}) = \log f(\mathbf{y} \mid \mathbf{h}) = \sum_{n=1}^{M} \log f_w(e(n)) \tag{C.3}$$

with respect to the channel estimator vector $\mathbf{h}_{N_T}$, where the assumed channel order is N_T and $f_w(.)$ is the scalar pdf of the channel noise $w(k)$.

As has been shown in chapter 4, the ML estimator can be obtained by the gradient ascent search

as:

$$\hat{\mathbf{h}}_{k+1} = \hat{\mathbf{h}}_k + \mu \left. \frac{\partial \mathcal{L}(\mathbf{h} \mid \mathbf{y})}{\partial \mathbf{h}} \right|_{\mathbf{h}=\hat{\mathbf{h}}_k} \tag{C.4}$$

Since, in our system model, it is assumed that the noise distribution is "unknown" (in (C.3)), a kernel density estimator is used to estimate this density as,

$$\hat{f}_w(e) = \frac{1}{M} \sum_{n=1}^{M} K(e - e(n)), \tag{C.5}$$

where $K(.)$ is the Gaussian kernel from chapter 4. Then from (C.3) and (C.5), and with some manipulations:

$$\left. \frac{\partial \mathcal{L}(\mathbf{h} \mid \mathbf{y})}{\partial \mathbf{h}} \right|_{\mathbf{h}=\mathbf{h}_k} = -\frac{1}{M\sigma} \sum_{n=1}^{M} \left. \frac{\sum_{i=1}^{M} (e(n) - e(i))(\mathbf{x}(n) - \mathbf{x}(i)) K(e(n) - e(i))}{\sum_{j=1}^{M} K(e(n) - e(j))} \right|_{\mathbf{h}=\hat{\mathbf{h}}_k} \tag{C.6}$$

Finally substituting (C.6) into (C.4) gives the NPML estimator.

C.2 Channel order mis-estimation

In general, if the channel order is assumed inaccurately, the estimation error comes from two parts: the coefficient-estimation error in the assumed model space and the space-estimation error between the true model and assumed model spaces [129]. As the assumed order increases, the coefficient-estimation error always increases, while the space-estimation error decreases until the assumed order is equal to, or larger than, the true channel order.

To be specific, if the channel order is under estimated, i.e. $l < N_T$, only the first l coefficients of the channel can be effectively estimated, and the received signal can be expressed as:

$$y(k) = \sum_{i=1}^{l} h(i)x(k-i) + w'(k), \qquad k = 1, \cdots, M \tag{C.7}$$

where $w'(k) = h(l+1)x(k-l-1) + \cdots + h(N_T)x(k-N_T) + w(k)$. Then the problem reduces to estimating the first l channel coefficients with the *equivalent channel noise* of $w'(k)$. Hence when $l < N_T$, beside that there are $N_T - l$ taps "missing", even the estimation errors corresponding to the first l coefficients is larger than those when $l = N_T$ since $\sigma^2_{w'(k)} > \sigma^2_{w(k)}$. Therefore the order under-estimate results in significant performance loss.

It is interesting to observe that $w'(k)$ basically forms a Gaussian mixture. Thus under rare cir-

cumstance, can $w'(k)$ be Gaussian. Further noting that NPML estimator demonstrates significantly superior performance to the LS estimator of chapter 4 in the presence of non-Gaussian noise. Thus the NPML estimator is always better than, or more robust to, the LS estimator when the channel order is under-estimated.

On another front, if the channel order is over-estimated (i.e. $l > N_T$), the "space-estimation error" disappears and only the "coefficient-estimation error" remains. Then the estimator vector can be expressed as:

$$\hat{\mathbf{h}} = [\mathbf{h}^{\mathrm{T}}\ 0^{\mathrm{T}}]^{\mathrm{T}} + \Delta_{\hat{\mathbf{h}}}, \tag{C.8}$$

where $\Delta_{\hat{\mathbf{h}}}$ can be regarded as a perturbation to the ideal estimate. In general, the larger the M is, the smaller the perturbation term is. Particularly, it can be easily verified that, if $x(k)$ and $w(k)$ are independent to each other and either of them has zero mean:

$$\lim_{N\to\infty} \Delta_{\hat{\mathbf{h}}_{LS}} = 0 \tag{C.9}$$

Thus if the data number is large enough, the last $l - N_T$ coefficients of the estimator are very small.

As an example, a system with presence of CCI is considered where SNR=20dB, $\mathbf{h} = [1\ 0.8\ 0.6\ 0.4\ 0.2]^T$ with $N_T = 5$, the interfering channel has SIR of 10dB, and the total number of samples is 100. Table C.1 shows the NMSE of the NPML estimator when the assumed channel order varies from 3 to 11 respectively. The NMSE is a performance index to measure the "goodness" of an estimator and is defined as

$$\mathrm{NMSE} = \frac{\mathbb{E}[\sum_{n=1}^{\infty}(h(n) - \hat{h}(n))^2]}{\mathbf{h}_{N_T}^T \mathbf{h}_{N_T}}. \tag{C.10}$$

It is clearly shown in Table C.1 that the NMSE reaches the minimum at $l = N_T$. But when $l \geq N_T$, the NMSE are within a narrow range, all significantly below those for $l < N_T$. This indicates that the order under-estimation is more serious a problem than the order over-estimate in terms of performance, though the latter imposes more complexity.

C.3 NPML estimator with order estimation

In this section, first it is shown that the traditional AIC is unreliable to apply for the case of estimated Gaussian pdf and the estimated non-Gaussian pdf, and then a simple method to estimate the channel order is proposed.

C.3.1 Order estimation based on AIC

p	SNR=20dB, N=100 No CCI		SNR=20dB, N=50 No CCI		SNR=40dB, N=50 SIR = 10dB	
	$\mathrm{AIC}_{\hat{\mathcal{L}}}$	$\mathrm{AIC}_{\hat{\sigma}^2}$	$\mathrm{AIC}_{\hat{\mathcal{L}}}$	$\mathrm{AIC}_{\hat{\sigma}^2}$	$\mathrm{AIC}_{\hat{\mathcal{L}}}$	$\mathrm{AIC}_{\hat{\sigma}^2}$
3	-85.39	-144.77	-35.50	-64.05	-33.59	-57.75
4	-179.63	-286.81	-85.21	-64.04	-54.92	-100.15
5	-464.65	**-455.82**	-193.96	**-211.84**	-57.72	-106.89
6	-465.80	-454.01	-192.40	-210.10	-55.93	-105.25
7	**-470.73**	-453.39	-201.67	-208.40	-54.21	-103.78
8	-468.73	-451.38	-199.69	-206.71	-55.64	-107.21
9	-467.33	-449.36	-198.09	-205.06	-53.89	-105.70
10	-466.22	-449.22	**-205.55**	-207.00	**-57.97**	**-111.96**
11	-468.58	-451.17	-203.81	-205.54	-56.78	-110.22

Table C.2: *AIC for different scenarios*

AIC is the most widely used order selection criterion which is defined as [125]:

$$\mathrm{AIC}_{\hat{\mathcal{L}}} = -2\mathcal{L}(\mathbf{h}_l \mid \mathbf{y}) + 2l, \tag{C.11}$$

where $\mathcal{L}(\mathbf{h}_l \mid \mathbf{y})$ is defined in (C.3). When the noise is Gaussian, (C.11) can be simplified to:

$$\mathrm{AIC}_{\hat{\sigma}^2} = M \log \hat{\sigma}^2 + 2l, \tag{C.12}$$

where $\hat{\sigma}^2 = (1/M)\sum_{n=1}^{M} e^2(n)$.

Unfortunately, neither $\mathrm{AIC}_{\hat{\mathcal{L}}}$ nor $\mathrm{AIC}_{\hat{\sigma}^2}$ is reliable to estimate the channel order for the NPML estimator: first, although the kernel density estimation (C.5) can be used to estimate the likelihood, it is not accurate enough to calculate the $\mathrm{AIC}_{\hat{\mathcal{L}}}$; second, $\mathrm{AIC}_{\hat{\sigma}^2}$ is only limited to Gaussian cases.

For illustration, the AIC for the same channel is calculated as that used in the previous section, and the results are shown in Table C.2, where the minimum values are highlighted in bold. Recall the true channel order N_T is 5. In the first case, a pure Gaussian channel is considered, where SNR=20dB, the number of sample $M = 100$ and no CCI. It is clear that $\mathrm{AIC}_{\hat{\sigma}^2}$ has its minimum at $l = 5$ but $\mathrm{AIC}_{\hat{\mathcal{L}}}$ at $l = 7$ which is biased away from N_T. In the second case,

the same channel but M is decreased to 50. It is observed that $\text{AIC}_{\hat{\sigma}^2}$ still finds the true order, but $\text{AIC}_{\hat{\mathcal{L}}}$ has the minimum which is further away from N_T. This is not surprising because, as M decreases, the density estimation becomes poorer and so does estimated $\text{AIC}_{\hat{\mathcal{L}}}$. In the last case, the interfering channel is introduced where SIR=10dB, the SNR is increased to 40dB, by which the channel becomes totally different from Gaussian. Under such scenario, neither $\text{AIC}_{\hat{\mathcal{L}}}$ nor $\text{AIC}_{\hat{\sigma}^2}$ estimates the channel order well. In conclusion, the AIC is unreliable for use with NPML estimator for order selection.

C.3.2 A simple order estimation method for the NPML estimator

A simple method to estimate the channel order is proposed below. The idea is based on the previous observation that, when the channel order is over-estimated, the extra taps are usually small compared to the others.

To be specific, at every iteration of the NPML estimation, the summation of squares for the last V coefficients of the estimator is measured. If it is smaller than V times of a pre-set threshold ϵ, then the order is decreased by 1; otherwise, if the summation of squares for the last $V-1$ taps is larger than $(V-1)\epsilon$, the order is increased by 1, and if none of the above then the order remains unchanged. In summary, the following procedure combining the order selection and the NPML channel estimation together is employed:

For every iteration , $k = 1, 2, 3, \cdots$
  Do the kernel density estimation based on (C.5).
  Update the estimator according to (C.4).
  if $\sum_{i=l(k)-V+1}^{l(k)} \hat{\mathbf{h}}_{l(k)}^2 < V \cdot \epsilon$
    $l(k+1) = l(k) - 1$
  else if $\sum_{i=l(k)-V+2}^{l(k)} \hat{\mathbf{h}}_{l(k)}^2 < (V-1) \cdot \epsilon$
    $l(k+1) = l(k)$
    else
    $l(k+1) = l(k) + 1$
    end
  end

In the above procedure, $l(k)$ is the tap-length at the kth iteration and V is an integer no less than 1. V has two effects: first, to create a "guard margin" so that the estimation is based on V, rather

than 1, coefficient values; second, to make the search escape from the local minima which are the zero coefficients within the range of the channel spread. Then if the threshold value ϵ is properly chosen, $l(k)$ will converge to within the range of $[N_T, N_T + V - 1]$. Obviously this method tends to over-estimate the order.

The threshold ϵ depends on both, the channel specifics and the channel estimator. When the number of samples M is large enough, the extra taps are normally very small, thus allowing us wider range to choose ϵ from. When M is small, the NPML estimator significantly outperforms the LS estimator as the former can explore the "local statistics" much better than the LS. In fact with a fixed $\epsilon = 0.01$ extensive simulations under different scenarios such as different channel, SNR, and M were under taken. The results show that the proposed method always works well as long as M is reasonably large (e.g. $M \geq 30$).

Alternatively, a dynamic threshold may be used, i.e. ϵ varies at each iteration. It has been shown in Section C.2 that the channel estimation consists of the true channel plus a perturbation term. It is obvious that, the larger M is, or the smaller $\hat{\sigma}^2$ is, the smaller the perturbation is and then the smaller the ϵ should be. Inspired by this observation, a dynamic threshold is suggested:

$$\epsilon(k) = \frac{C \cdot \hat{\sigma}^2(k)}{M}, \tag{C.13}$$

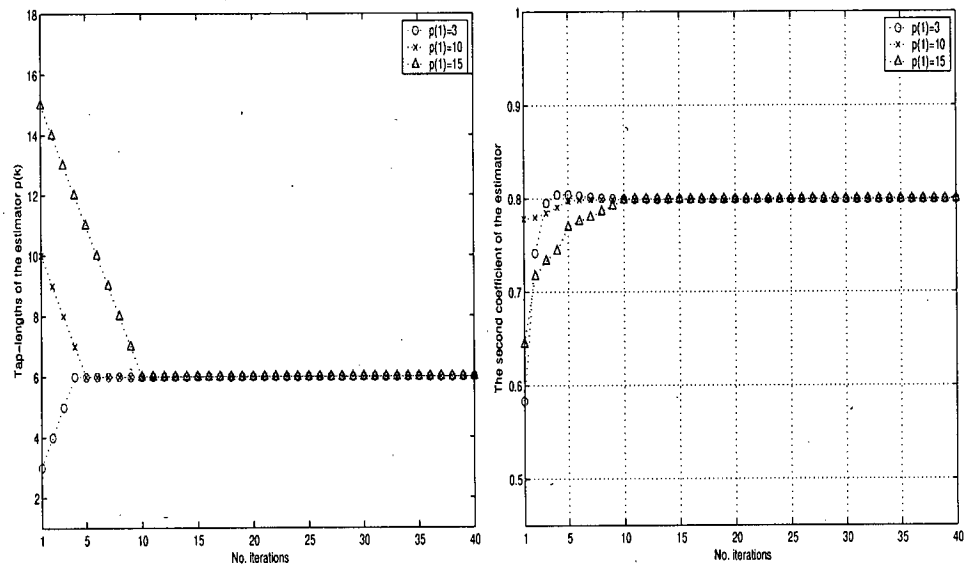
where C is a constant. To make the algorithm robust, it is ensured that the $\epsilon_{min} < \epsilon(k) < \epsilon_{max}$, where ϵ_{max} and ϵ_{min} are maximum and minimum values for the threshold respectively.

C.4 Simulation study

For the simulations in this section, the channel is the same as that for the previous examples in this paper, $V = 3$, the dynamic threshold based on (C.13) is used where $C = 10$, $\epsilon_{max} = 0.05$ and $\epsilon_{min} = 0.005$. All results are based on one typical run. The learning curves of the tap-length and the second tap coefficient of the estimator are shown in (a) and (b) respectively for each figure.

Figure C.1 investigates the proposed algorithm for different initialisation of the estimator's tap-length, where SNR= 20dB, SIR= 10dB and $M = 100$. It is clear that, for all initialisations, the individual tap-lengths converge to '6' which is in the range of $[N_T, N_T + V - 1]$ as expected.

Figure C.2 compares the results for different sample number M, where SNR=20dB and SIR=10dB.

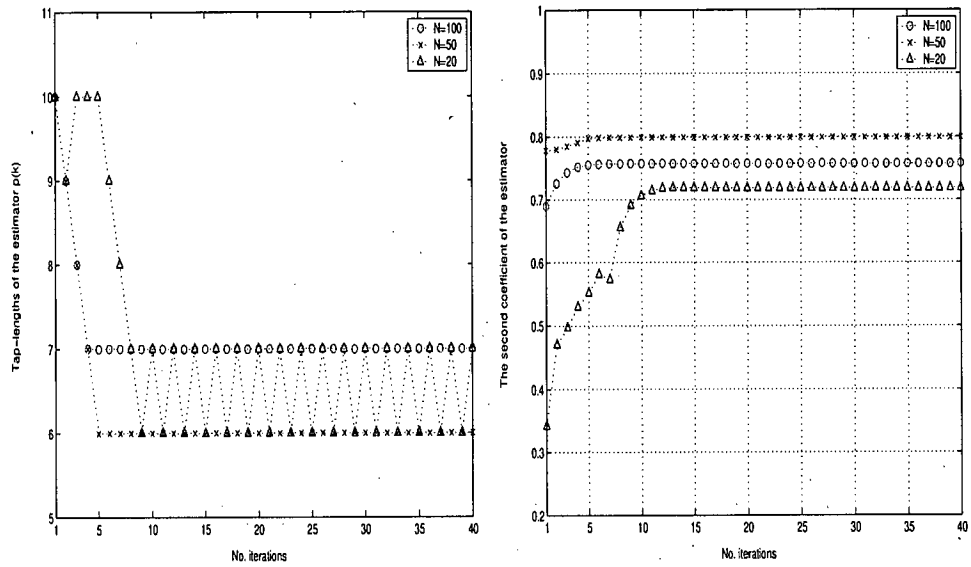


(a) Tap-length learning curve. (b) 2^{nd} tap learning curve.

Figure C.1: *Learning curves for different tap-length initialisation.*

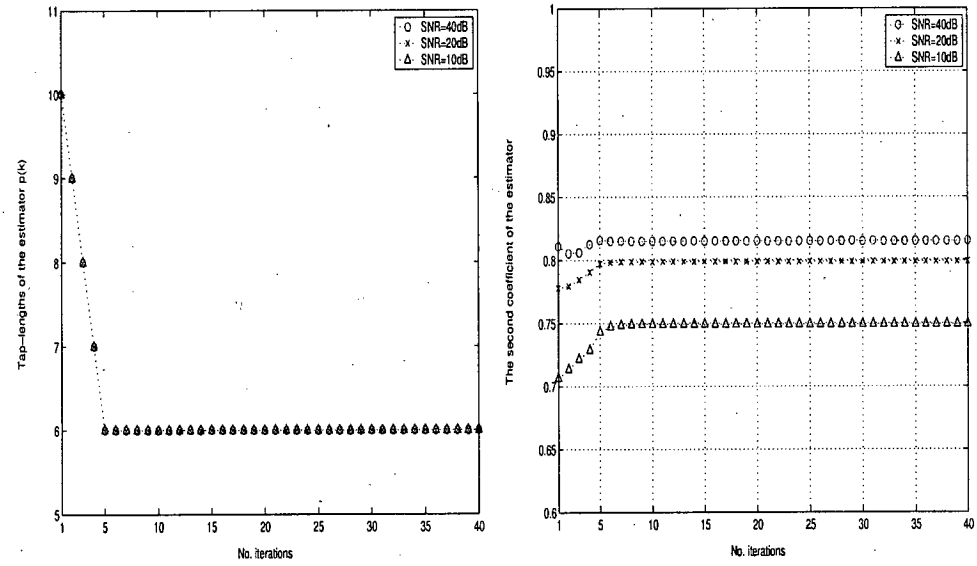
It is observed that even if M is as low as '20', the algorithm can still track the order, although it oscillates between '6' and '7' as shown in Figure C.2 (a). Accordingly, the slower coefficient convergence for $M = 20$ is also observed in Figure C.2 (b).

Figure C.3 shows the results for different SNR-s, where the SIR= 10dB and $M = 50$. It is obvious that the proposed algorithm works well for all these SNR-s. From Figure C.3 (b), it is interesting to note that NPML algorithm performs better for SNR=20dB than for 40dB, as the former converges closer to the true 2^{nd} coefficient (which is 0.8) of the channel. This is because that, in presence of CCI, the channel with SNR at 40dB is further "away" from Gaussian than that with SNR at 20dB, resulting in less accuracy for the kernel density estimation.



(a) Tap-length learning curve. (b) 2^{nd} tap learning curve.

Figure C.2: *Learning curves for different number of samples.*



(a) Tap-length learning curve. (b) 2^{nd} tap learning curve.

Figure C.3: *Learning curves for different SNR.*

Appendix D
Publications

V. Bhatia, B. Mulgrew and A. T. Georgialidis, "Stochastic Gradient Algorithms for Equalisation in Alpha Stable Noise", *Elsevier Journal for Signal Processing*, In Press, Aug. 2005.

V. Bhatia, B. Mulgrew and A. T. Georgialidis, "Minimum BER Equaliser for Alpha-Stable Noise", *Proc. of EUSIPCO*, (Vienna, Austria), Sep. 2004.

V. Bhatia and B. Mulgrew, "A EM-based channel estimator in Non-Gaussian Noise", *Proc. IEEE Veh. Tech. Conf.*, (Los Angeles, US), pp. 3871-3875, Sep. 2004.

V. Bhatia and B. Mulgrew, "A Minimum Error Entropy based channel estimator in Co-Channel Interference with Lower Bounds", *Proc. of IEEE Intl. Conf. on Signal Proc. and Comms.*, (Bangalore, India), pp. 41-45, Dec. 2004.

V. Bhatia and B. Mulgrew, "Non-parametric Maximum Likelihood Channel Estimator in Non-Gaussian Noise", *Proc. Signal Proc. for Wireless Comm.*,(Kings College, London), vol.1, Jun. 2004.

V. Bhatia, B. Mulgrew and D. D. Falconer, "Non-Parameteric ML Channel Estimator and Detector for OFDM", *Proc. of EUSIPCO*, (Antalya, Turkey), Sep. 2005.

Y. Gong, V. Bhatia, B. Mulgrew, and C. F. N. Cowan, "A Non-Parameteric ML Eestimator with Unkown Channel Order", *Proc. of EUSIPCO*, (Antalya, Turkey), Sep. 2005.

V. Bhatia and B. Mulgrew, "Error Whitening Non-Parameteric ML Channel Estimator", *IEEE Workshop on Stat. Signal Proc.*, (Bordeaux, France), Jul. 2005.

Stochastic Gradient Algorithms for Equalisation in Alpha Stable Noise

V. Bhatia, B. Mulgrew and A.T. Georgiadis
Signals and Systems Group, IDCOM,
Room 2.12, AGB Building,
School of Engineering and Electronics,
The University of Edinburgh,
Kings Buildings,
Mayfield Road,
Edinburgh EH9 3JL,
U.K.
Email: [v.bhatia, b.mulgrew, atg]@ee.ed.ac.uk

Abstract

This paper addresses the problem of developing a least mean squares (LMS) style decision feedback equaliser algorithm for minimising bit error rate (BER) in impulsive noise environments characterised by the alpha stable distribution. The development exploits the stable nature of the alpha distribution and the concepts build on earlier work in a Gaussian noise environment. Further, a Wiener-filter-with-limiter solution is also presented and used as a performance bench mark. An improvement in convergence and BER performance is achieved by using a minimum bit error rate (MBER) cost function instead of a conventional LMS based design. The ability of least BER (LBER) equalisers based on a Gaussian noise assumption to operate in an alpha stable noise environment is also highlighted.

1. INTRODUCTION

The Gaussian process has always been the dominant noise model in communications and signal processing, mainly because of the central limit theorem. In addition, the Gaussian assumption often leads to analytically tractable solutions [4]. Unfortunately, in some communication channels, the observation noise exhibits impulsive, as well as Gaussian characteristics. The sources of impulsive noise may be either natural (e.g. lightning, ice-cracking), or man-made. It may include atmospheric noise or ambient noise. It might come from relay contacts, electro-magnetic devices, electronic apparatus, or transportation systems, switching transients, and accidental hits in telephone lines [6], [7]. Most of the present day systems are optimised under the Gaussian assumption and their performance is degraded by the occurrence of impulsive noise [3].

Impulsive noise is more likely to exhibit sharp spikes or occasional bursts of outlying observations than one would expect from Gaussian distributed signals. A variety of impulsive noise models have been proposed in [7] and [8]. However, a common model to represent impulsive phenomena is the family of α-stable random variables [4]. Stable distributions share defining characteristics with the Gaussian distribution, such as the stability property and central limit theorems. The empirical data indicates that the probability density functions (pdf's) of the impulsive noise processes exhibit a simlarity to the Gaussian pdf, being bell shaped, smooth and symmetric, but at the same time having significantly heavier tails [6].

In [1] it was shown that adaptive linear equalisation based on probability of error performs better than that based on a least squared error cost function. Further, it was shown that the state-translated design

achieves a lower BER than conventional DFE structures [9]. However the adaptive least BER algorithm of [1] was derived on the basis that the noise was drawn from a Gaussian distribution. While general purpose adaptive algorithms for alpha-stable noise environments have been proposed (e.g. [4] and [5]), they are based on the L_p norm of the error rather than BER.

In this paper, we develop a class of adaptive equalisers (similar in complexity to the LMS algorithm) where the BER is minimized in an alpha stable noise environment. The least BER rate algorithms of [1] are shown to be particular cases of these algorithms when the noise is Gaussian. Generally, in adaptive equalisation, the Wiener solution is taken as a point of reference in measuring performance. However in alpha stable noise the variance of the input signal to the equaliser is infinite and the Wiener solution is not defined. In practice, every receiver has a finite input dynamic range which *limits* the amplitude of received samples and produces finite variances. Using this assumption we derive the 'Wiener solution with limiter' (WSL) for alpha stable noise environments. As pointed out in [2] the limiter facilitates the use of standard correlation based algorithms in alpha stable noise. Simulation results show that the LMS algorithm fails to converge to this WSL solution while the proposed alpha-stable-noise least-BER (LBER) algorithm seeks the optimum BER solution for comparable computational complexity. Robustness of the Gaussian-noise LBER algorithms of [1] in alpha stable noise is also demonstrated through simulation.

The paper is organised as follows: a brief overview of stable processes is provided in section 2; an overview of the state-translated DFE structure is presented in section 3; the WSL in alpha stable noise is derived in section 4; the LBER adaptive algorithm for alpha stable noise is derived in section 5; simulation techniques, assumptions and results are discussed in section 6; finally conclusions are drawn in section 7.

2. THE CLASS OF STABLE RANDOM VARIABLES

The main characteristics of a non-Gaussian stable random variable (RV) is that the tails of its probability density function (pdf) are heavier than those of the normal density. The symmetric α-stable (SαS) pdf is defined by means of its characteristic function $F(\omega) = \exp(\delta i\omega - \gamma|\omega|^{\alpha})$. The parameters α, γ and δ describe completely an SαS distribution. The characteristics exponent α ($0 < \alpha \leq 2$) controls the heaviness of the tails of the stable density: $\alpha = 2$ is the Gaussian case; smaller α values are associated with heavier tails. The dispersion parameter γ ($\gamma > 0$) plays an analogous role to the variance and refers to the spread of the distribution. Finally, the location parameter δ is comparable with the mean of the distribution.

Theoretical justification for using the stable distribution as a basic statistical modelling tool comes from the generalized central limit theorem. Unfortunately, no closed-form expressions exist for the stable density, except for the Gaussian ($\alpha = 2$), Cauchy ($\alpha = 1$) and Pearson ($\alpha = \frac{1}{2}$) distributions. An important property of all non-Gaussian stable distributions is that only the lower moments are finite. That is, if $\mathbf{x}$ is a non-Gaussian stable RV, then $\mathbf{E}_{\mathbf{x}}\{|x|^p\} < \infty$ iff $p < \alpha$. A well known consequence of this property is that all stable RV's with $\alpha < 2$ have infinite variance [3]. For further discussion on α-stable RV's and their properties can be found in [4].

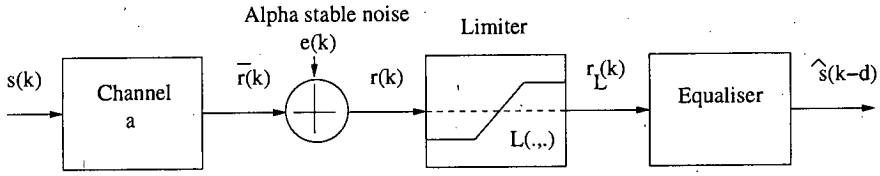


Fig. 1. Typical Communication System

3. EQUALISER STRUCTURES

The channel is modelled as a finite impulse response filter with an additive noise source, and the received signal at sample k is

$$r(k) = \bar{r}(k) + e(k) = \sum_{i=0}^{n_a-1} a_i s(k-i) + e(k)$$

where $\bar{r}(k)$ denotes the noiseless channel output; n_a is the channel length and a_i are the channel tap weights; the white noise $e(k)$ has zero mean and is drawn from an alpha stable distribution with dispersion γ and characteristic exponent α; the symbol sequence $\{s(k)\}$ is independently identically distributed (IID) and has a M-PAM (pulse amplitude modulation) constellation defined by the set [9]

$$s_i = 2i - M - 1, 1 \leq i \leq M$$

Throughout this paper $M = 2$ for 2-PAM is considered. For a conventional linear-combiner DFE the decision variable z at time k is a linear combination of received samples and past decisions:

$$z(k) = \mathbf{w}^T \mathbf{r}(k) - \mathbf{b}^T \hat{\mathbf{s}}_b(k)$$

where $\mathbf{r}(k) = [r(k)\, r(k-1) \ldots r(k-m+1)]^T$ is the channel observation vector, $\hat{\mathbf{s}}_b(k) = [\hat{s}(k-d-1)\, \hat{s}(k-d-2) \ldots \hat{s}(k-d-n)]^T$ is the past detected symbol vector, $\mathbf{w} = [w_0\, w_1 \ldots w_{m-1}]^T$ is the feedforward coefficient vector and $\mathbf{b} = [b_1\, b_2 \ldots b_n]^T$ is the feedback coefficient vector. The integers d, m and n will be referred to as the decision delay, the feedforward delay and feedback taps respectively. Without loss of generality, $d = n_a - 1$, $m = n_a$ and $n = n_a - 1$ will be used as this choice of DFE structure parameters which is sufficient to guarantee the linear separability of the subsets of the channel states related to the different decisions [9]. Alternatively the linear-combiner DFE can be expressed in state translated form [10]:

$$z(k) = \mathbf{w}^T (\mathbf{r}(k) - \mathbf{F}_2 \hat{\mathbf{s}}_b(k)) = \mathbf{w}^T \mathbf{r}'(k) \tag{1}$$

where $\mathbf{F}_2$ is constructed by partitioning the channel impulse response matrix $\mathbf{F} = [\mathbf{F}_1\, \mathbf{F}_2]$, where:

$$\mathbf{F}_1 = \begin{bmatrix} a_0 & a_1 & \cdots & a_{n_a-1} \\ 0 & a_0 & \ddots & \vdots \\ \vdots & \ddots & \ddots & a_1 \\ 0 & \cdots & 0 & a_0 \end{bmatrix}$$

$$\mathbf{F}_2 = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ a_{n_a-1} & 0 & \ddots & \vdots \\ a_{n_a-2} & a_{n_a-1} & \ddots & 0 \\ \vdots & \ddots & \ddots & 0 \\ a_1 & \cdots & a_{n_a-2} & a_{n_a-1} \end{bmatrix}$$

Since the linear-combiner DFE is a special case of the generic DFE structure, by performing translation of eq. (1), it is reduced to the equivalent linear equaliser 'without decision feedback':

$$f'(\mathbf{r}'(k)) = \mathbf{w}^T \mathbf{r}'(k) \tag{2}$$

The decision boundary of this equivalent linear equaliser consists of $M-1$ hyperplanes defined by: $\mathbf{r}' : \mathbf{w}^T \mathbf{r}' = 2i - M$, $1 \leq i \leq M-1$. These $M-1$ parallel hyperplanes can always be designed properly to separate the M subsets of the translated channel states $R^{(i)}$, $1 \leq i \leq M$. In particular, for $M = 2$, the decision boundary, $\mathbf{r}' : \mathbf{w}^T \mathbf{r}' = 0$, is a hyperplane passing through the origin of the $\mathbf{r}'$(k)-space. It is shown, in [9], that in the state translation the channel states remain separable despite translation. The states can be made separable by applying a simple initial condition. The performance of state translated

linear combiner DFE is shown to be better than conventional minimum mean square error (MMSE) DFE, however performance depends on the accuracy of the built-in channel estimator [9].

The Wiener or MMSE solution is often said to provide the optimal $\mathbf{w}$ and $\mathbf{b}$. It is however optimal only with respect to the mean square error criterion. Obviously, there must exist a solution $\mathbf{w}_{opt}$ which achieves the best equalisation performance for the structure of eq. (2). We refer to this $\mathbf{w}_{opt}$ as the minimum bit error rate (MBER) solution of the linear-combiner DFE. The MMSE linear-combiner DFE is generally not this MBER solution. A natural question is how different the MMSE and MBER solutions can be. The difference in performance of MMSE and MBER solutions for Gaussian distributed noise is demonstrated in [9].

4. Minimum bit error rate equalisation

It is obvious that the MBER and MMSE solutions are different as discussed in [9]. In this section we first describe the MBER criterion for a general DFE structure. The calculation of MMSE solution is not possible for α-stable noise because of infinite variance. However by introducing practical design constraint of a limited dynamic range we can estimate the Wiener solution (the conventional way). For clarity we describe it as the WSL.

A. MBER criterion

The bit error rate (BER) observed at the output of the equaliser is dependent on the distribution of the decision variable $z(k)$ which in turn is a function of the equaliser tap weights. To be more specific, the probability of error, P_E, is:

$$P_E = P(\text{sgn}(s(k-d))z(k) < 0)$$

The sign adjusted decision variable $z_s(k) = \text{sgn}(s(k-d))z(k)$ is drawn from a mixture process. From the definition of $z(k)$,

$$\begin{aligned} z_s(k) &= \text{sgn}(s(k-d))(\mathbf{w}^T\mathbf{F}\mathbf{s}(k) - \mathbf{b}^T\hat{\mathbf{s}}_b(k)) + \text{sgn}(s(k-d))\mathbf{w}^T\mathbf{e}(k) \\ &= \text{sgn}(s(k-d))z'(k) + e'(k) \end{aligned} \tag{3}$$

$\mathbf{e}(k) = [e(k)\, e(k-1) \ldots e(k-d-n)]^T$ is the vector of noise samples; $\mathbf{s}(k) = [s(k)\, s(k-1) \ldots s(k-d-n_{n_a})]^T$ is the vector of transmitted symbols. The first term on the right hand side of eq. (3), $\text{sgn}(s(k-d))z'(k)$, is the noise-free sign-adjusted equaliser output and is a member of a finite set with N_z elements - these are the local means of the mixture. Without noise the combination of channel and DFE is a finite state machine whose state is defined by the vector $\mathbf{s}(k)$. Thus if $\mathbf{s}(k) \in \{\mathbf{s}_1 \cdots \mathbf{s}_i \cdots \mathbf{s}_{N_z}\}$, the state $\mathbf{s}_i$ uniquely defines the state of $z'(k)$, $\mathbf{r}(k)$, $s(k-d)$ and $\hat{\mathbf{s}}_b(k)$ - label these z_i, $\mathbf{r}_i$, s_i and $\hat{\mathbf{s}}_{bi}$ respectively. Note that while $\mathbf{s}(k)$ has N_z states, $s(k-d)$ has 2 possible values (2-PAM). However since $s(k-d)$ is a component of the vector $\mathbf{s}(k)$, the state of $\mathbf{s}(k)$ uniquely defines the value of $s(k-d)$. The second term $e'(k)$ is a zero mean α-stable white noise process with dispersion $\gamma(\sum_{j=1}^{m}|w_j|^\alpha)^{\frac{1}{\alpha}}$ and characteristic exponent α - defining the distribution about the local means.

B. Wiener Solution with limiter

In an α-stable noise environment with $\alpha < 2$ the variance of the noise is infinite [4] making the use of the traditional Wiener solution meaningless. Nevertheless, all receivers in practice have a finite input dynamic range. This is achieved by using the structure as shown in Fig-1. The limiter at the front end of the receiver is assumed to be an ideal saturation device, with transfer function

$$L(x,G) = \begin{cases} x & : \ |x| \le G \\ \text{sgn}(x)G & : \ \text{elsewhere} \end{cases}$$

G being the saturation point of the limiter. The saturation limit level G is kept at a reasonable distance from noiseless channel states to preserve the noise structure and not limit (clip) the noiseless channel state instead.

Provided $G > \max(\bar{r}(k))$, the received signal at the output of the limiter, $r_L(k)$, is the sum of the noise-free channel output $\bar{r}(k)$ and what is termed a *truncated* $\alpha-$stable noise process, $e_L(k)$: $\forall k$. The pdf of this *truncated* $\alpha-$stable noise process is given by:

$$f_\alpha(s,G) = f_\alpha(s)\prod(s,G) + I_l(-G)\delta(s+G) + I_r(G)\delta(s-G) \tag{4}$$

where

$$\prod(s,G) = \begin{cases} 1 & : \quad -G \leq s \leq G \\ 0 & : \quad \text{elsewhere} \end{cases}$$

$$I_l = \int_{-\infty}^{G} f_\alpha(s)ds, I_r = \int_{G}^{\infty} f_\alpha(s)ds$$

where $f_\alpha(s)$ represents the alpha stable distribution. The pdf of the channel states (assuming equi-probable) is impulses at the channel centres.

$$f_{\overline{c_i}}(s) = \frac{1}{N_{sc}}\sum_{i=1}^{N_{sc}}\delta(s-\overline{c_i}) \tag{5}$$

Since the truncated alpha stable noise process of eq. (4) and the noise-free scalar channel states of eq. (5) are independent, the combined pdf is given by:

$$f_{r_L}(s) = \frac{1}{N_{sc}}\sum_{i=1}^{N_{sc}} f_{r_L|\overline{c_i}}(s) = \frac{1}{N_{sc}}\sum_{i=1}^{N_{sc}} f_\alpha(s-\overline{c_i}, -G-\overline{c_i}, G-\overline{c_i}) \tag{6}$$

where $N_{sc} = 2^{n_a}$ is the number of the scalar centres $\overline{c_i}$ of the channel, i.e., $\overline{c_i} = \mathbf{a^T}.\mathbf{s_{ch_i}}(i = 1, 2, ..., N_{sc})$, where $\mathbf{a} = [a_0 ... a_{n_a-1}]^T$ and $\mathbf{s_{ch_i}} = [s(k) ... s(k-n_a+1)]^T$ are all the possible combinations for the channel input vector. This pdf is same as that observed at the output of the receiver, which confirms independence. The limiter "$L(x,G)$" truncates the pdf of the received signal and its tails are concentrated at the points $\pm G$, where they appear as *Dirac* impulses $\delta(s)$. The noise variance can be calculated theoretically from [3], with knowledge of α, limiting level G and noiseless channel states $\overline{c_i}$.

From classical Wiener filter theory [11], the WSL is $\mathbf{w}_o = \mathbf{R}^{-1}\mathbf{p}$, where $\mathbf{w}_o$ is the optimum tap-weight vector, $\mathbf{R} = \mathbb{E}\{\mathbf{r_L}\mathbf{r_L}^T\}$ is the input autocorrelation matrix, $\mathbf{p} = \mathbb{E}\{\mathbf{r_L}s\}$ is the cross-correlation vector and $\mathbf{r_L} = [r_L(k)\, r_L(k-1) \ldots r_L(k-m+1)]^T$. The autocorrelation matrix is simply the sum of two autocorrelation matrices: (i) the autocorrelation matrix associated with the noise free channel output; (ii) a scaled identity matrix. The scale factor is the variance of the truncated alpha stable process and thus the scale factor is $\int s^2 f_\alpha(s,G)ds$. The cross-correlation matrix is simply the cross-correlation of the noise free channel output with the target symbol. Because the variance of the truncated alpha stable noise process is a function of both the parameter α and the limiter value G, the WSL will be as well.

Thus we can calculate the theoretical Wiener solution after the limiter using the independence property, which was not obvious from [3].

5. Stochastic Gradient Adaptive Equalisers

In this section we directly address the problem of minimising BER in an alpha stable noise environment and derive a stochastic gradient algorithm for the task. As the development is in terms of probability of error rather than mean squared error the requirement for a limiter is removed.

Consider the noise density function $p(x)$ associated with the zero mean random variable x. The density function is symmetrical and normalised such that the variance or dispersion is unity. The associated

distribution function is $P(x)$. The "generalised" error function is $Q(x) = 1 - P(x)$ and its derivative is $Q'(x) = -p(x)$. The probability of error at the output of a linear or state translation equaliser with N noise free states as a function of the weight m-vector $\mathbf{w}$ is:

$$P_E(\mathbf{w}) = \frac{1}{N}\sum_{i=1}^{N} Q(g_i(\mathbf{w}))$$

where $g_i(\mathbf{w})$ is the signed decision variable associated to the i^{th} state, normalized by the "strength" of the noise. In the Gaussian case [1] :

$$g_i(\mathbf{w}) = \frac{\mathbf{w}^T \mathbf{r}_i s_i}{||\mathbf{w}||\sigma} \tag{7}$$

where $\mathbf{r}_i$ is the i^{th} noise free received vector; the Euclidean norm is $||\mathbf{w}|| = (\sum_{j=1}^{m} |w_j|^2)^{\frac{1}{2}}$; s_i is the transmitted symbol associated with that vector; σ^2 is the noise variance. In the α-stable case:

$$g_i(\mathbf{w}) = \frac{\mathbf{w}^T \mathbf{r}_i s_i}{||\mathbf{w}||_\alpha \gamma^{\frac{1}{\alpha}}} \tag{8}$$

where the "α-norm" is defined as: $||\mathbf{w}||_\alpha = (\sum_{j=1}^{m} |w_j|^\alpha)^{\frac{1}{\alpha}}$ For adaptive filters, derivatives of the form $\partial P_E / \partial w_j : \forall j$ are required.

$$\begin{aligned}\frac{\partial P_E}{\partial w_j} &= \frac{1}{N}\sum_{i=1}^{N} Q'(g_i(\mathbf{w}))\frac{\partial g_i(\mathbf{w})}{\partial w_j} \\ &= -\frac{1}{N}\sum_{i=1}^{N} p(g_i(\mathbf{w}))\frac{\partial g_i(\mathbf{w})}{\partial w_j}\end{aligned}$$

In the Gaussian case the derivative of eq. (7) is given by:

$$\begin{aligned}\frac{\partial g_i(\mathbf{w})}{\partial w_j} &= \frac{\partial}{\partial w_j}\left(\frac{\mathbf{w}^T}{||\mathbf{w}||}\right)\frac{\mathbf{r}_i s_i}{\sigma} \\ &= \frac{1}{||\mathbf{w}||}\left(\mathbf{1}_j^T - \frac{\mathbf{w}^T w_j}{||\mathbf{w}||^2}\right)\frac{\mathbf{r}_i s_i}{\sigma}\end{aligned}$$

where $\mathbf{1}_j$ is an m-vector with all zero elements apart from the j^{th} entry which is unity. In the α-stable case the derivative of eq. (8) is taken:

$$\begin{aligned}\frac{\partial g_i(\mathbf{w})}{\partial w_j} &= \frac{\partial}{\partial w_j}\left(\frac{\mathbf{w}^T}{||\mathbf{w}||_\alpha}\right)\frac{\mathbf{r}_i s_i}{\gamma^{\frac{1}{\alpha}}} \\ &= \frac{1}{||\mathbf{w}||_\alpha}\left(\mathbf{1}_j^T - \frac{\mathbf{w}^T |w_j|^{\alpha-1}\mathrm{sgn}(w_j)}{||\mathbf{w}||_\alpha^\alpha}\right)\frac{\mathbf{r}_i s_i}{\gamma^{\frac{1}{\alpha}}}\end{aligned}$$

Since the α-stable case is more general we will work with it from now on. Multiply out gives:

$$\frac{\partial g_i(\mathbf{w})}{\partial w_j} = \frac{1}{||\mathbf{w}||_\alpha}\left(r_{ij} - \frac{z_i |w_j|^{\alpha-1}\mathrm{sgn}(w_j)}{||\mathbf{w}||_\alpha^\alpha}\right)\frac{s_i}{\gamma^{\frac{1}{\alpha}}}$$

where r_{ij} is the j^{th} element of $\mathbf{r}_i$ and $z_i = \mathbf{w}^T \mathbf{r}_i$ i.e. the equaliser output associated with the i^{th} noise free state. Collecting partial derivatives together to form a gradient vector we have:

$$\nabla P_E(\mathbf{w}) = -\frac{1}{N}\sum_{i=1}^{N} p(g_i(\mathbf{w}))\frac{1}{||\mathbf{w}||_\alpha}\left(\mathbf{r}_i - \frac{z_i \langle \mathbf{w} \rangle_\alpha}{||\mathbf{w}||_\alpha^\alpha}\right)\frac{s_i}{\gamma^{\frac{1}{\alpha}}}$$

where $\langle \mathbf{w} \rangle_\alpha$ is an m-vector with j^{th} element is $|w_j|^{\alpha-1}\text{sgn}(w_j)$. Since the norm of the weight vector does not affect P_E in the binary signalling case it can be set to unity at each iteration thus:

$$\nabla P_E(\mathbf{w}) = -\frac{1}{N\gamma_\alpha^{\frac{1}{\alpha}}} \sum_{i=1}^{N} p\left(\frac{z_i s_i}{\gamma_\alpha^{\frac{1}{\alpha}}}\right) (\mathbf{r}_i - \langle \mathbf{w} \rangle_\alpha z_i)\, s_i$$

Using the kernel density ideas developed in [1] leads to an LMS-style least bit error rate (LBER) algorithm.

Filter output:

$$z(k) = \mathbf{w}^T(k)\mathbf{r}(k)$$

Update weights:

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \mu p\left(\frac{z(k)s(k-d)}{\gamma_\alpha^{\frac{1}{\alpha}}}\right) (\mathbf{r}(k) - \langle \mathbf{w}(k) \rangle_\alpha z(k)) \frac{s(k-d)}{\gamma_\alpha^{\frac{1}{\alpha}}} \tag{9}$$

The equaliser tap weights are normalised after each update. The final decision, $\hat{s}(k-d)$, is made on the filter output $\mathbf{w}^T(k)\mathbf{r}'(k)$.

6. Simulation Study

In this paper, the SNR of the limited received signal $r_L(k)$ is used for performance evaluation in environments where the noise variance is infinite. By using the limiter the SNR is always finite and hence measurable. This is referred as the SNR *at the receiver*. Simulations were performed for anti-podal signalling ($M = 2$), assuming that the noise is Cauchy distributed i.e. $\alpha = 1$ and the limiter, at DFE front-end, is at ± 4 [3] to avoid being close to noiseless channel states at the transmitter output. The variance of the truncated alpha stable process $e_L(k)$ is calculated as discussed in [3]. Fig-1 represents the receiver architecture considered in simulations.

As the performance of equalisers are highly dependent on the nature of the channel considered two channels which have been well studied in the literature were chosen to characterize performance. These channels have impulse responses [0.3482 0.8704 0.3482] and [1.0 0.50 0.25]. The DFE structure is chosen to be $d = 2$, $m = 3$ and $n = 2$. The legends in Fig-2, Fig-3, Fig-4 and Fig-5 depict: a) 'LMS' refers to a conventional LMS algorithm for both the feedforward and feedback taps of a conventional DFE, b) 'LBER-Gaussian' refers to a LBER algorithm for adapting both feedforward and feedback equaliser taps of a conventional DFE assuming that the noise is Gaussian [1], c) 'LBER-Cauchy' refers to adapting both the feedforward and feedback taps of a conventional DFE assuming Cauchy distributed noise using eq. (9), d) 'state trans-Gaussian' refers to the same adaptive algorithm as (b) but with state translated design [9], e) 'state trans-Cauchy' refers to the same adaptive algorithm as (c) but with a state translated design, f) 'modified Wiener' represents WSL calculated after the limiter using $r_L(k)$ as discussed in section-4. A total of 10^5 samples were used to generate the convergence and performance plots using Matlab. In order to make a fair comparison of the relative performance of the algorithms the adaptation constant μ is fixed as $\frac{1}{6(m+n)}$ for all the adaptive algorithms compared in this paper. A large sample size and ensemble for simulations was taken to reach conclusions because of the impulsive (high variations in input signal amplitude) nature of alpha-stable noise.

An ensemble of 100-runs was taken to generate convergence plots as shown in Fig-2 and Fig-4 at a SNR of 7.9 dB's. As can be observed the convergence behaviour of the LMS is unstable. This can be attributed to the fact that the LMS is dependent on the magnitude of the instantaneous error, which varies a lot in an impulsive noise environments. Algorithms designed to minimise BER in a Gaussian noise environment converge more slowly than those specifically designed for the Cauchy noise environment. It is safe to conclude that the state translated design for Cauchy noise has faster and more stable convergence than the other algorithms.

To observe the BER performance of these algorithms an ensemble of 1000-runs was taken. The equalisers were trained using the first 1000-samples of a particular run after which training was inhibited

and the BER for that run measured. The final BER estimate was obtained by averaging over 1000 such runs in the ensemble. Fig-3 and Fig-5 summarize the results for the two channels used. At a BER of 5×10^{-3} we can gain approximately 5 dB's by using a minimum-BER criterion instead of an LMS algorithm. Again the Gaussian noise based LBER algorithms perform well with respect to Cauchy noise based LBER algorithms which are tailored to the particular environment. The state translated Cauchy noise based LBER DFE performs better than the other algorithms as is apparent from both Fig-3 and Fig-5. It is also interesting to observe that this MBER algorithm performs better than the WSL.

While the WSL provides an optimal solution in a MSE sense it does not minimise MBER. However the LMS algorithm, which would normally find the MSE solution, fails to converge to this solution in this environment. The LBER algorithms, by their nature, seek the desired optimum MBER solution. LBER algorithms have been demonstrated to find the optimum BER solution with a computational complexity similar to that of the LMS. From the simulations we observe that the state-translated DFE for Cauchy distributed noise has better convergence and BER performance than the other algorithms considered. LBER algorithms based on Gaussian noise [1] assumptions have also been demonstrated to perform well in α-stable noise environments.

7. Conclusions

An minimum bit error rate adaptive algorithm for impulsive noise modelled as α-stable noise has been proposed in this paper. By introducing a limiter at the receiver front-end both SNR and Wiener solution can be calculated theoretically and by simulations. It is shown that for minimum bit error design, the adaptation is a function of the noise density function. The comparison between various adaptive algorithms working in identical channel, noise and DFE structure has been drawn. The LBER-Cauchy and the state trans-Cauchy has faster convergence than the other adaptive algorithms in Cauchy noise environments, which is a special form of α-stable noise. Extensive simulations strongly suggest that the state-translated design for the α-stable noise has better convergence and BER performance than the other algorithms. It is also interesting to observe that the adaptive algorithms based on a Gaussian noise assumption despite slow convergence in impulsive noise environments perform closer to those designed with Cauchy noise assumption. Lastly as expected the LMS algorithm performs poorer that the other algorithms in α-stable noise environments. Observations from Fig-3 and Fig-5 suggests MBER algorithms' superior performance with respect to the WSL solution.

8. Acknowledgment

The authors acknowledge the support of the UK Engineering and Physical Sciences Research Council (EPSRC) for this work.

REFERENCES

[1] B. Mulgrew and S. Chen, "Adaptive Minimum-BER Linear Decision Feedback Equalisers for Binary Signalling," Signal Processing, vol. 81, no. 7, July 2001, pp. 1479-1489.

[2] A. Swami and B. Sadler, "On some detection and estimation problems in heavy-tailed noise," Signal Processing, vol. 82, no. 12, Dec. 2002, pp. 1829-1846.

[3] A.T. Georgiadis and B. Mulgrew, "Adaptive Bayesian Decision Feedback Equaliser for Alpha-Stable Noise Environments," Signal Processing, vol. 81, no. 8, Aug. 2001, pp. 1603-1623.

[4] M. Shao and C. L. Nikias, "Signal Processing with fractional lower order moments: Stable processes and their applications," Proc. IEEE, vol. 81, no. 7, July 1993, pp. 986–1009.

[5] O. Arikan and A. Enis Cetin and E. Erzin, "Adaptive filtering for non-Gaussian stable processes," IEEE Signal Processing Letters, vol. 1, Nov. 1994, pp. 163–165.

[6] G. A. Tsihrintzis and C. L. Nikias, "Performance of optimum and suboptimum receivers in the presence of impulsive noise modeled as an alpha-stable process," Proc. MILCOM'93, vol. 2, July 1993 pp. 658–662.

[7] L. F. Lind and N. A. Mufti, "Efficient method for modelling implusive noise in a communication system," IEE Electronics Letters, vol. 32, no. 16, Aug. 1996, pp. 1440–1441.

[8] I. Mann, S. McLaughlin, W.Henkel, R. Kirby, and T. Kessler, "Impulse generation with appropriate amplitude, length, inter-arrival, and spectral characteristics," IEEE Journal on selected areas in com., vol. 20, no. 5, July 2002, pp. 901–912.

[9] S. Chen, B. Mulgrew, E. S. Chng and G. J. Gibson, "Space translation properties and the minimum-BER linear-combiner DFE," IEE Proc. Communications, vol. 145, no. 5, 1998, pp. 316–322.

[10] A. P. Clark, L. H. Lee and R. S. Marshall, "Developments of the conventional non-linear equaliser," IEE Proc. Part F, vol. 129, no. 2, 1982, pp. 85-94.

[11] Simon Haykins, Adaptive Filter Theory, 4 ed., Pearson Education Asia Inc., 2002, Chapter 2, pp. 104.

[12] E. A. Lee and D.G. Messerschmitt, Digital Communication, 1 ed., Kluwer Academic Publishers, 1988, Chapter 7, pp. 257.

[13] E.E. Kuruoglu, C. Molina and W.J. Fitzgerald, "Approximation of alpha-stable probability densities using finite Gaussian mixtures," Proc. EUSIPCO 98, Signal Processing IX: Theories and Applications, vol. 2, Sept. 1998, pp. 989–992.

[14] R.J. Kozick, R.S. Blum, and B.M. Sadler, "Signal processing in non-Gaussian noise using mixture distributions and the EM algorithm," Thirty-First Asilomar Conference on Signals, Systems and Computers, vol. 1, Nov. 1997, pp. 438–442.

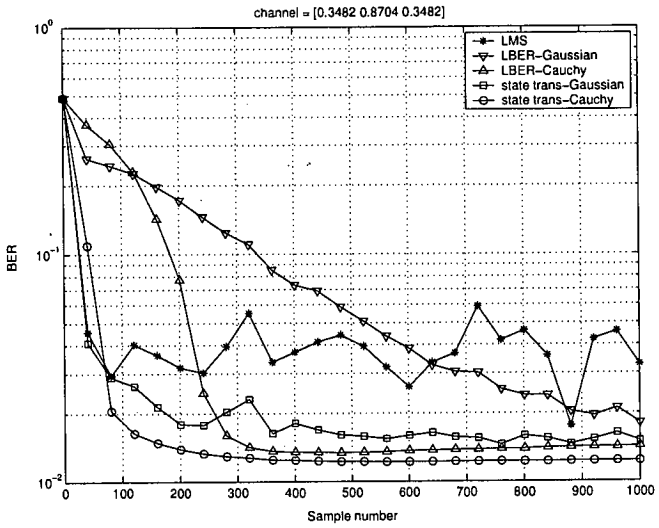


Fig. 2. Convergence plot for Cauchy (α = 1) distributed noise

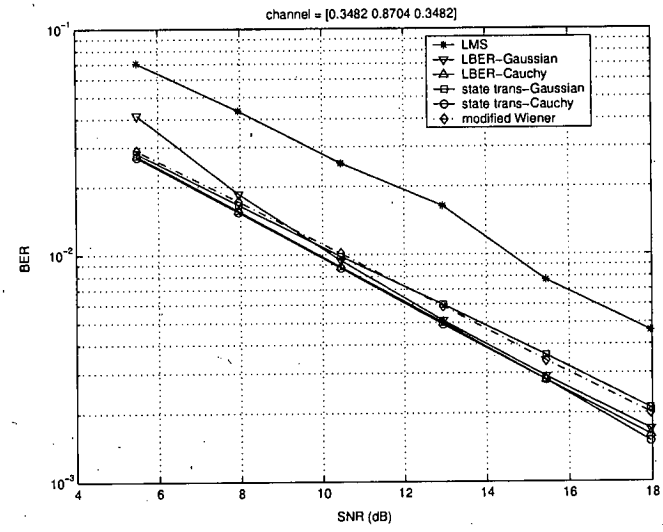


Fig. 3. Performance plot for Cauchy (α = 1) distributed noise

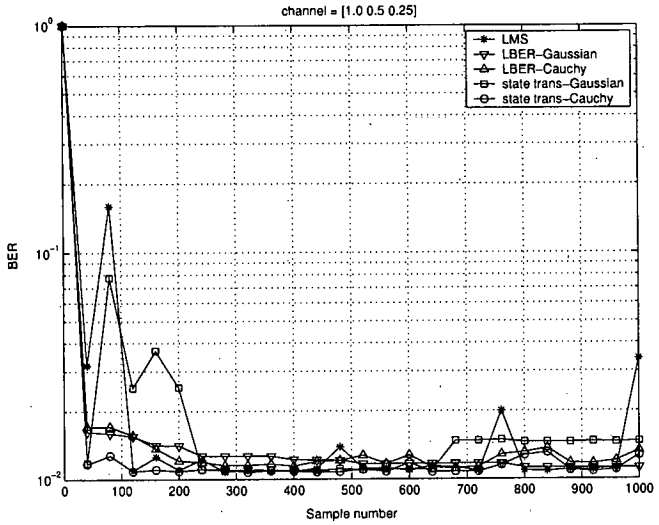


Fig. 4. Convergence plot for Cauchy (α = 1) distributed noise

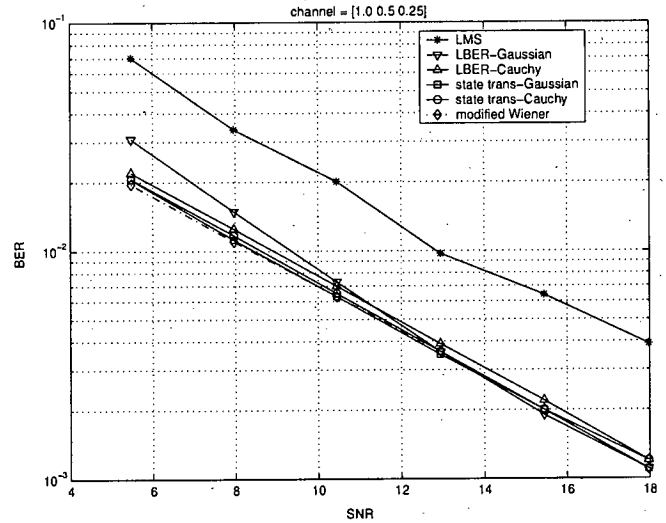


Fig. 5. Performance plot for Cauchy (α = 1) distributed noise

NON-PARAMETRIC ML CHANNEL ESTIMATOR AND DETECTOR FOR OFDM

Vimal Bhatia[1], *Bernard Mulgrew*[1], *and David D. Falconer*[2]

[1]Institute for Digital Communications,
School of Engineering and Electronics,
The University of Edinburgh,
Edinburgh, EH9 3JL, United Kingdom
email: [v.bhatia, bm]@ee.ed.ac.uk

[2]Broadband Communications and Wireless System,
Department of Systems and Computer Engineering,
Carleton University,
Ottawa, ON K1S 5B6, Canada
email: ddf@sce.carleton.ca

ABSTRACT

A maximum-likelihood channel estimator for the orthogonal frequency division multiplexing (OFDM) communication environments, in presence of interference is discussed here. We study a training based scenario, where the channel is estimated based on pilots that precede the transmission of the information. To reduce the number of estimation parameters, we estimate the channel iteratively in time-domain. Since interference from other users provides no useful information we do not estimate parameters of the interference and neither we neglect the affect of the interference instead interference along with Gaussian noise is perceived as non-Gaussian noise. The algorithm assumes no *apriori* knowledge about the interfering channel and signal at the receiver, further no-assumption on the statistical properties of the interferer is assumed which makes this algorithm robust. The estimated channel information along with the estimated distribution are then utilized to equalize the subsequent data blocks.

1. INTRODUCTION

Orthogonal frequency division multiplexing (OFDM) is a promising multi-carrier digital communication technique for transmitting data at high bit-rates over wireless or wire-line channels. The high-speed serial data is converted into many low bit rate streams that are transmitted in parallel, thereby increasing the symbol duration and reducing the intersymbol interference (ISI). These features have led to an increase in the use of OFDM or related techniques in many high bit rate communication systems. Discrete multi-tone modulation which is quite similar to OFDM is extensively used in digital subscriber line (xDSL) communication systems. OFDM has been chosen for digital audio broadcasting (DAB) and digital video broadcasting (DVB). It is also used for the 2.4 GHz wireless local area networks (IEEE 802.11g).

Coherent OFDM transmission invariable requires estimation of the channel frequency response (i.e. the gains of the OFDM tones). Currently there can be three possible solutions: 1) blind, 2) semi-blind, and 3) pilot aided. In blind channel estimation techniques, the channel is estimated without the knowledge of the transmitted sequence. It is attractive as the throughput is higher as no bits are lost in training. However it requires large amount of data to be stored before channel estimation can begin, which invariably introduces delays. The pilot based technique estimates the channel by transmitting a known (at the receiver) training sequence along with the unknown data at the receiver. The receiver estimates the channel using some criterion based on comparing the change in these pilots due to channel. The semi-blind techniques try to reduce the size of the training sequence by exploiting both the known and the unknown (blind) portions of the data.

Channel estimation in OFDM is critical to the overall performance of the communication system. Insertion of pilots in OFDM symbols provides a base for reliable channel estimates. There has been considerable increase in channel estimation research over the years [1], [2] etc. However most of the current work is based on channel estimation for Gaussian channels or assuming that the interference is very low. This assumption is usually based on two reasons: first the interference to have tractable mathematical models and by central limit theorem. This assumption is however not always valid in scenarios where there are a small number of interferers (e.g. Bluetooth device or microwave oven operating in presence of a WLAN). With the co-existence of various wireless equipments in home or office environments the interference from neighboring devices has become a major concern [3]. In interference affected channels we can be sure that algorithms designed for Gaussian assumption are not optimal [4]. From here on we refer to the traditional Gaussian assumption estimator (which assumes zero or negligible interference) as least squares (LS) estimator.

Here we estimate the fading channel in presence of interference directly in time domain using maximum likelihood (ML) technique. The channel is assumed to be deterministic for a given block. The algorithm discussed in [2] specifically deals with the synchronous interference, however it was noted that interference was modelled as Gaussian, which may not be the case if only a few (or in fact one major interferer as in [5]) are present. In this paper we make no such *apriori* assumption on the interfering received signal distribution. Moreover no parameter of the interferer is estimated specifically. In fact, the presence of interference along with Gaussian noise is jointly considered as a Gaussian mixture noise [4] and [6]. It is noted that traditional zero forcing equalization technique fall short of performance in presence of interference. Simulation results confirm the non-optimal estimates when LS is used and improved bit error rate (BER) performance by using the presented algorithm. Throughout the paper capitalized variables represents frequency domain values while the bold variables represents vectors. Also $\Re$ and $\Im$ represents real and imaginary part.

The paper is organized as follows. In section-2 the problem statement is formulated for a general OFDM communication system followed by brief discussion on den-

This research was sponsored by the UK Engineering and Physical Sciences Research Council and IEE Hudswell Bequest Fellowship

sity estimation. The iterative non-parametric maximum-likelihood (NPML) channel estimator is described in section-3. Section-4 discusses the modified non-parametric symbol-by-symbol equalizer. To test the robustness of the algorithm, in section-5, the simulation results are presented. Conclusions based on analysis and simulation are drawn at the end.

2. FORMULATION OF THE PROBLEM

2.1 OFDM System Model

The baseband equivalent representation of a typical OFDM system as in Fig-1 is considered here. We focus our discussion on estimation of one OFDM symbols instead of a sequence of symbols for the reasons justified below. At the transmitter side, the serial input data is converted into M parallel streams, and each data stream is modulated by a linear modulation scheme, such as QPSK, 16QAM or 64QAM. If QPSK is used, for instance, the binary input data of $2M$ bits will be converted into M QPSK symbols by the serial-to-parallel converter (S/P) and the modulator. The modulated data symbols, which are denoted by complex-valued variables $X(0),...,X(m),...,X(M-1)$, are then transformed by the IFFT, and the complex-valued outputs $x(0),...,x(k),...,x(M-1)$ are converted back to serial data for transmission. A guard interval is inserted between symbols to avoid inter-symbol interference (ISI). If the guard interval is longer than the channel delay spread, and if we discard the samples of the guard at the receiving end, the ISI will not affect the actual OFDM symbol. Therefore, the system can be analyzed on a symbol-by-symbol basis. At the receiver side, after converting the serial data to M parallel streams, the received samples $y(0),...,y(k),...,y(M-1)$ are transformed by the FFT into $Y(0),...,Y(m),...,Y(M-1)$ [1]. Using the notations for the OFDM symbols, the output of the channel can be written as

$$y(k) = \sum_{l=0}^{L-1} h^*(l)x(k-l) + \sum_{p=0}^{P-1}\sum_{l=0}^{L-1} g_p^*(l)u_p(k-l) + n(k), \quad (1)$$
$$0 \le k \le M-1$$

where h and x represents desired user's channel and data respectively. Without loss of generality we choose complex conjugate h^* instead of h in above equation [7]. L represents the channel length and $n(k)$ is the additive white Gaussian noise. P represents the number of interferers where g_p and u_p is the interfering channel and signal respectively. Note that $y(k)$, $x(k)$, $n(k)$, $h(l)$, $u_p(k)$ and $g_p(l)$ are all complex valued. It is assumed that the channel and interference doesn't change during the block transfer and interference is synchronous which makes the above representation possible.

If cyclic prefix is used for the guard interval, intercarrier interference (ICI) in multipath channel can also be avoided. Then it can be shown that the following simple relation between $Y(m)$ and $X(m)$ holds:

$$\begin{aligned} Y(m) &= \left(\sum_{l=0}^{L-1} h^*(l)\exp^{(-j2\pi\frac{ml}{M})}\right)X(m) \\ &\quad + \left(\sum_{p=0}^{P-1}\sum_{l=0}^{L-1} g_p^*(l)\exp^{(-j2\pi\frac{ml}{M})}U_p(m)\right) + N(m) \quad (2) \\ &= H(m)X(m) + I(m) + N(m), 0 \le m \le M-1 \quad (3) \\ &= H(m)X(m) + N'(m), 0 \le m \le M-1 \quad (4) \end{aligned}$$

where $H(m)$ is the complex frequency response of the channel at the subchannel m, $I(m)$ be the complex interference at that subchannel m and $N(0),...,N(M-1)$ are the DFT of $n(0),...,n(M-1)$. If $n(0),...,n(M-1)$ are i.i.d. Gaussian random variables, so are the transformed variables $N(0),...,N(M-1)$. It is assumed that the interfering signals $U_p(0),...,U_p(M-1)$ are also OFDM signals, with same block and cyclic pre-fix lengths, and they are block synchronous with the desired signal. Eq. (4) shows that the received signal is the transmitted signal attenuated and phase shifted by the frequency response of the channel at the subchannel frequencies due to fading in presence of interference and noise [1]. It is assumed to be that noise is represented as complex independent identically distributed (i.i.d.) with vector $\mathbf{n} = [n(0),n(1),...,n(M-1)]^T$ with each component of $\mathbf{n}$ distributed as $\mathbb{C}\,\mathcal{N}(\mu_i,\sigma_i^2)$ and are also independent. The multivariate complex Gaussian pdf is just the product of the marginal pdf or

$$f(\mathbf{n}) = \prod_{i=0}^{M-1} f(n(i)) \quad (5)$$

which follows from the usual property of the pdf for real independent random variables, this can be written as

$$f(\mathbf{n}) = \frac{1}{\pi^M \prod_{i=0}^{M-1}\sigma_i^2}\exp\left[-\sum_{i=1}^{M-1}\frac{1}{\sigma_i^2}\mid n(i)\mid^2\right] \quad (6)$$

Since the joint pdf depends on $\Re$ and $\Im$ only through $\mathbf{n}$, we can view the pdf to be that of the 'scalar random variable n'. This pdf eq. (6) is called a 'complex Gaussian pdf' for a scalar complex random variable and is denoted by $\mathbb{C}\,\mathcal{N}(0,\sigma_i^2)$ [8].

3. KERNEL DENSITY ESTIMATION

Since we have complex noise and interference we can model it as a 'complex Gaussian mixture' pdf, where the real and complex are assumed independent as discussed earlier. Parzen window or kernel density estimation assumes that the probability density is a smoothed version of the empirical sample. Its estimate $\hat{f}(y)$ of a complex random variable $y = \Re\{y\} + i\Im\{y\}$ is simply the average of radial kernel function centered on the points in a sample M of the instance of y:

$$\hat{f}(y) = \frac{1}{M}\sum_{j=1}^{M}\phi(y - y(j)) \quad (7)$$

We here assume ϕ to be Gaussian kernel (Parzen kernel) [6]:

$$\phi(y) = \mathcal{N}(0,\sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}}\exp\left(\frac{-\mid y\mid^2}{2\sigma^2}\right) \quad (8)$$

variance defined as σ^2. The joint pdf $\hat{f}(y)$ depends on the real and complex components through y, we can view the pdf to be that of the scalar random variable y, as the notation suggest [8]. Other choices of kernel like *Epanechnikov kernel* are also possible. It can be shown that under the right conditions $\hat{f}(y)$ will converge to the true density $f(y)$ as $\mid M \mid \to \infty$.

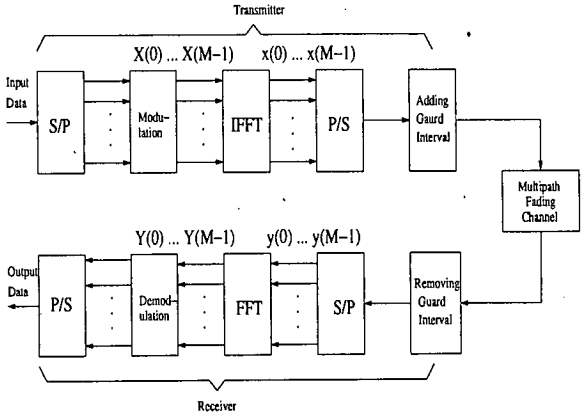


Figure 1: A typical OFDM communication system

4. NON-PARAMETRIC ML CHANNEL ESTIMATION

The channel impulse response $\mathbf{h} = [h(0),...,h(L-1)]$ are independent complex-valued Gaussian random variables (which represents a frequency-selective Rayleigh fading channel). In regular OFDM system, the channel delay spread L is much smaller than the number of subcarriers. This leads to a high correlation between the channel frequency responses $H(m), 0 \leq m \leq M-1$, even when $h_l, 0 \leq l \leq L-1$,are independent [1]. We estimate the channel impulse response $\mathbf{h} = [h(0),...,h(L-1)]$ directly, as the channel frequency response $H(0),...,H(M-1)$ are generally correlated among each other (as discussed above) and the impulse response may be independently specified, thus the number of parameters in the time domain is smaller than that in the frequency domain.

The combined interference and AWGN $N'(m)$ in eq. (4) is together taken as a noise that is non-Gaussian because of the presence of interference [6]. As also discussed in [6] the LS estimator does not find the optimal solution in the case of non-Gaussian noise. If the noise was Gaussian then the solution to the ML leads to the LS estimate. However, in communication systems where the noise is non-Gaussian (or Gaussian mixture) i.e. Gaussian in presence of interference, no closed form ML solution exists for such non-Gaussian distributions. Thus we rely on the iterative algorithm to find the ML estimate of the channel. In this algorithm we first initialize channel update algorithm with LS estimate, then we estimate the likelihood on the pilots. After estimating likelihood we find the ML solution iteratively on the pilot symbol. The classical stochastic gradient algorithm is used with a log-likelihood being the cost function i.e. the gradient here is the first derivative of the log-likelihood function with a constant multiplier (similar to well known gradient ascent algorithm) [9]. The update equation is:

$$\hat{\mathbf{h}}_k = \hat{\mathbf{h}}_{k-1} + \mu(k) \nabla_{\mathbf{h}} \mathscr{L}(\mathbf{h} \mid \mathbf{Y})|_{\mathbf{h}=\hat{\mathbf{h}}_{k-1}} \quad (9)$$

where $\mu(k)$ is the adaptation constant and $\nabla_{\mathbf{h}}$ represents the gradient of the cost function. Referring to eq. (4) and eq. (9) the likelihood function can be written as:

$$L(\mathbf{h} \mid \mathbf{Y})|_{\mathbf{h}=\hat{\mathbf{h}}_{k-1}} = f(\mathbf{Y} \mid \mathbf{h}) = \prod_{i=1}^{M} f_{N'}(E(i))$$

$f_{N'}(.)$ is scalar pdf of 'complex Gaussian mixture' of data length from $i = 1,...,M$ and the previous estimation error is defined as:

$$E(i) = Y(i) - \left(\sum_{l=0}^{L-1} h_k^*(l) \exp^{(-j2\pi \frac{il}{M})} \right) X(i) \quad (10)$$

Kernel density estimators are known to be effective in estimating the pdf over short data record and also provide a differentiable smooth estimated pdf. Using kernel density estimator we obtain:

$$\hat{f}_{N'}(E) = \frac{1}{M} \sum_{j=1}^{M} \phi(E - E(j)) \quad (11)$$

where M is the number of subcarriers.

$$\mathscr{L}(\mathbf{h} \mid \mathbf{Y})|_{\mathbf{h}=\hat{\mathbf{h}}_{k-1}} = \sum_{i=1}^{M} \log(f_{N'}(E(i)))$$
$$= \sum_{i=1}^{M} \log \sum_{j=1}^{M} \phi(E(i) - E(j)) - \log \mid M \mid \quad (12)$$

Maximizing the log-likelihood function w.r.t to channel weight vector. By definition of complex vector differentiation [7] we obtain,

$$\nabla_{\mathbf{h}} \mathscr{L}(\mathbf{h} \mid \mathbf{Y})|_{\mathbf{h}=\hat{\mathbf{h}}_{k-1}} = \sum_{i=1}^{M} \frac{\sum_{j=1}^{M} \frac{\partial \phi(E(i)-E(j))}{\partial \mathbf{h}}}{\sum_{j=1}^{M} \phi(E(i) - E(j))} \quad (13)$$

Thereby substituting this gradient in eq. (9) gives an iterative solution. As with any stochastic gradient algorithm the choice of optimal $\mu(k)$ varies with application and requirements. As discussed in [9] we choose $\mu(k) = \frac{\sigma^2}{M}$ in eq. (9) (where σ is chosen as in [6]) and witnessed convergence in a few iterations.

5. NON-PARAMETRIC SYMBOL-BY-SYMBOL EQUALIZER

Similar to the channel estimator discussed before, the conventional detector (equalizer [1]) is based on the Gaussian assumption that is again not optimal for the interference affected channels. The performance of this zero-forcing equalizer [1] is highly sensitive to the quality of estimated channel and the ratio of interfering received signal with estimated channel. Thus for the said equalizer structure the decision boundary is clearly non-linear. Thereby we use a probabilistic equalizer whose decision is based on the estimated likelihood. For the estimated channel impulse response $\hat{\mathbf{h}}_\mathbf{k}$ (after convergence) from eq. (9) the ML estimate of the transmitted signal can be obtained by

$$\hat{X}(m) = argmax_{X=\hat{X}} \left(\hat{f}_E(Y(m) \mid \hat{H}(m)) \right)|_{\mathbf{h}=\hat{\mathbf{h}}_k} \quad (14)$$

where $\hat{H}(m)$ is the frequency response of the estimated channel and without loss of generality it is assumed that X is

ERROR WHITENING NON-PARAMETRIC MAXIMUM LIKELIHOOD CHANNEL ESTIMATOR

Vimal Bhatia and Bernard Mulgrew

Institute for Digital Communications,
The University of Edinburgh
Kings Buildings, Mayfield Road
Edinburgh, U.K. EH9 3JL
Email: [v.bhatia, b.mulgrew]@ee.ed.ac.uk

ABSTRACT

The presence of co-channel interference has been a major hindrance in improving the performance of present day communication systems. In this paper we discuss a iterative block based maximum-likelihood algorithm using kernel density estimates to improve channel estimation in presence of co-channel interference. As it is known that the interference is correlated, we first reduce this correlation by using a whitening filter. After whitening, we estimate this unknown whitened likelihood pdf by using kernel density estimator at the receiver. Thereby combining log-likelihood as cost function with whitening filter and kernel density estimate, a robust channel estimator for correlated noise environments is formed. The simulations for co-channel interference in presence of Gaussian noise, confirms that a better estimate can be obtained by using the proposed technique as compared to the traditional least squares algorithm, which is optimal in the Gaussian noise environments.

1. INTRODUCTION

In communication systems that experience multiple access interference (MAI) or co-channel interference (CCI) the observed noise (noise plus interference) deviates from Gaussianity [1]. Most of the present day systems are optimized under the Gaussian assumption and their performance is degraded by the occurrence of correlated non-Gaussian noise i.e. Gaussian noise in presence of coloured interference. The least squares (LS) criterion is considered optimal and is equivalent to maximum-likelihood (ML) for channel estimation when the transmitted symbols are equi-probable, unknown parameters are deterministic and the noise is additive Gaussian. However, in scenarios where the received data is not a sufficient statistics [1] or is corrupted by non-Gaussian noise, the traditional LS-based methods are inefficient and the LS estimator may not be as efficient (or equivalent) to ML estimator. From here on, in order to avoid confusion, the traditional (Gaussian assumption) ML is referred to as the LS solution.

In this paper we take a training based channel estimator, where the channel is estimated over a block of data (similar to GSM) [1]. As discussed above due to presence of co-channel interference the observed noise at the receiver does not remain Gaussian. This degrades the performance of traditional LS based channel estimators. We developed a channel estimator in [2] which showed considerable improvement in the estimates in presence of interference. However in [2] we assumed that the input noise plus interference although correlated, could be modelled by independent and identical distribution. In this paper, we extend this earlier work by first whitenening the noise plus intereference and finding a ML estimnate for the channel as well as the whitening filter. Techniques which whiten the noise plus interference before suppressing the interference has been proposed in [3, 4]. This forms a powerful technique to improve the performance, but since, in practice, the tap length of this whitening filter cannot be increased to a large value, the ideal assumption of white Gaussian noise (after the linear prediction error (LPE) filter [1]) does not hold. In this paper we make no such (Gaussian) assumption on the distribution of the whitened noise, which makes this technique robust to the various noise distributions.

It is observed from [1, 5, 6] that various types of noises encountered in communications can be modelled as a Gaussian mixture. In order to estimate this unknown noise pdf at the receiver we use the kernel density estimator. This is a non-parametric method of density estimation that allows the data to define the density directly. We here propose kernel density estimation based technique operating iteratively on a given block of data at each iteration. It is assumed that the corrupting noise pdf can be effectively modelled by a Gaussian mixture. We here make no *apriori* assumption on the number of Gaussian mixtures or their relative probabilities. Instead the received data is exploited using kernel

density estimators to estimate this pdf. An improvement in mean square error (MSE) performance in channel estimation over the traditional LS estimate is observed by using the proposed algorithm.

The paper is organized as follows. First, the problem statement is formulated in section-2 for a general communication system. Followed by short discussion on kernel density estimators in section-3. The non-parametric maximum likelihood algorithm using LPE filter and kernel density estimator is discussed in section-4. In section-5 simulation results are presented. Conclusions based on analysis and simulation are drawn at the end.

2. FORMULATION OF THE PROBLEM

The discrete-time model in the low-pass equivalent form of the communication system channel estimator is shown in Fig. 1. Without loss of generality, the input signal is assumed to be randomly generated binary anti-podal PAM signal, so that the transmitted symbols are $\mathbf{a}\epsilon\{\pm 1\}$. Here '$r$' represents the received signal and 'w' is a sequence of additive noise. The model is simplified by assuming that the channel is of order $N-1$ i.e. $\mathbf{h} = [h(0), h(1), ..., h(N-1)]$.

More precisely, the received signal $r(k)$ sampled once per symbol can be written as

$$r(k) = \sum_{n=0}^{N-1} h(n)a(k-n) + w(k) \qquad (1)$$

The problem is to estimate the channel coefficients from the received signal assuming that the input signal (as in supervised training mode) and the channel (tap) length is known at the receiver. Thus the problem reduces to the well known problem of system identification. There are various algorithms based on different criterions to estimate the channel taps. Usually the LS solution is taken as the optimum solution for the Gaussian noise environments where it is equivalent to a ML estimate [7]. However here we assume that the noise in presence of interference is correlated, thus LS does not provide the ML solution. To remove this correlation we use a LPE filter. The problem then reduces to the one shown in Fig. 1.

The problem can now be written as:

$$\sum_{i=0}^{P} z(i)r(k-i) = \sum_{l=0}^{L+P-1} a(k-l)\zeta(l) + \epsilon(k) \qquad (2)$$

where $\mathbf{z} = [z(0) = 1 z(1) = -\alpha(1)...z(p) = -\alpha(p)]$ are the coefficients of the LPE filter and the equivalent channel taps vector $\boldsymbol{\zeta} = [\zeta(0)\zeta(1)...\zeta(L+P-1)]$, where $\zeta(l) = \sum_i z(i)h(l-i)$. Ideally the $\epsilon(k)$ is a zero-mean white Gaussian process. Since from eq. (1), the model eq. (2) corresponds to assuming

$$\sum_{i=0}^{P} z(i)r(k-i) = \sum_{i=0}^{P} z(i) \sum_{l=0}^{L-1} a(k-i-l)h(l) = \sum_{l=0}^{L+P-1} a(k-l)\zeta(l) \qquad (3)$$

and

$$\sum_{i=1}^{P} z(i)w(k-i) = \epsilon(k) \qquad (4)$$

Therefore, the effect of the LPE filter is that of whitening the additive disturbance $w(k)$. The formulation eq. (2) permits the description of the channel plus the whitening filter as a vector inner product, which in turn allows the simultaneous estimation of the LPE coefficients and the equivalent channel taps at the output of the LPE filter [3]. In fact, letting $\boldsymbol{\alpha} = [\alpha(1)...\alpha(p)]$, eq. (2) can be rewritten as

$$[1 - \alpha(1)... - \alpha(p)] \begin{bmatrix} r(k) \\ r(k-1) \\ \vdots \\ r(k-P) \end{bmatrix}$$

$$= [\zeta(0)\zeta(1)...\zeta(L+P-1)] \begin{bmatrix} a(k) \\ a(k-1) \\ \vdots \\ a(L+P-1) \end{bmatrix} + \epsilon(k) \qquad (5)$$

or equivalently

$$r(k) = [\alpha(1)...\alpha(p)] \begin{bmatrix} r(k-1) \\ \vdots \\ r(k-P) \end{bmatrix}$$

$$+[\zeta(0)\zeta(1)...\zeta(L+P-1)] \begin{bmatrix} a(k) \\ a(k-1) \\ \vdots \\ a(L+P-1) \end{bmatrix} + \epsilon(k) \qquad (6)$$

$$= [\boldsymbol{\alpha}\ \boldsymbol{\zeta}]\mathbf{v}(k) + \epsilon(k) \qquad (7)$$

where $\mathbf{v}(k) = [r(k-1)...r(k-P)b(k-1)...b(k-L-P+1)]^T$ and with $\epsilon(k)$ white. Usually it is assumed that due to LPE filter $\epsilon(k)$ is Gaussian distributed. However in practice the larger the tap-length of the LPE filter, the more the equalizer states, the more computational complexity [1] for maximum likelihood sequence equalizer. Thus in practice the tap-length is usually restricted [1, 4]. This restriction leads $\epsilon(k)$ to non-Gaussianity. The channel estimator proposed in this paper does two tasks: (i) estimating the channel; (ii) estimating the uncorrelated $\epsilon(k)$ pdf at the receiver.

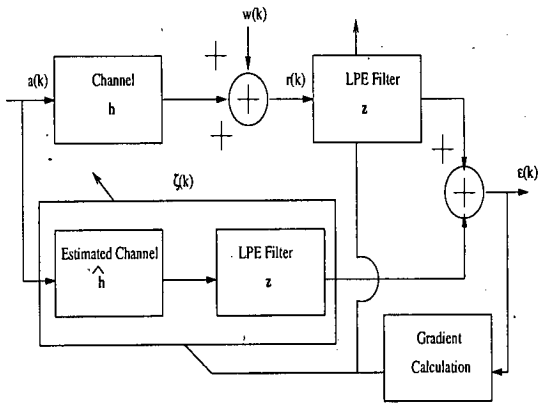


Fig. 1. Communication systems channel estimator with LPE filter

3. KERNEL DENSITY ESTIMATION

To estimate the pdf at the receiver we use the kernel density estimator technique. Parzen window or kernel density estimation assumes that the probability density is a smoothed version of the empirical sample. Its estimate $\hat{f}(y)$ of a random variable Y is simply the average of radial kernel functions centered on M-realizations of Y:

$$\hat{f}(y) = \frac{1}{M}\sum_{j=1}^{M}\phi(y - y(j)) \tag{8}$$

We here assume ϕ to be Gaussian kernel (Parzen kernel) [1]:

$$\phi(y) = \mathcal{N}(0,\sigma) = \frac{1}{\sqrt{2\pi\sigma^2}}\exp\left(\frac{-y^2}{2\sigma^2}\right) \tag{9}$$

where σ^2 is defined as the kernel variance (or width) [8]. Other choices of kernel like *Epanechnikov kernel* are also possible. It can be shown that under the right conditions $\hat{f}(y)$ will converge to the true density $f(y)$ as $\mid M \mid \rightarrow \infty$ [9].

4. NON-PARAMETRIC MAXIMUM-LIKELIHOOD (NPML) CHANNEL ESTIMATION WITH LPE

For the communication system represented by eq. (2) the ML estimate forms the optimal estimator for the channel. This problem can be viewed as the joint optimization problem [1], where we maximize the likelihood with respect to α and ζ. If the $\epsilon(k)$ was Gaussian then the LS solution as found in [1] could have been applied directly. However, since we assume that $\epsilon(k)$ is non-Gaussian and can be modelled as a Gaussian mixture we use the kernel density estimator to estimate this density. Since the kernel density estimator is essentially a Gaussian mixture formulation we can't get a closed form estimate of the α and ζ. We then use the iterative scheme as used in [2]:

$$\hat{\alpha}_k = \hat{\alpha}_{k-1} + \mu(k)\,\nabla_\alpha \mathcal{L}(\alpha \mid \mathbf{r},\zeta)\mid_{\alpha=\hat{\alpha}_{k-1},\zeta=\hat{\zeta}_{k-1}} \tag{10}$$

$$\hat{\zeta}_k = \hat{\zeta}_{k-1} + \mu(k)\,\nabla_\zeta \mathcal{L}(\zeta \mid \mathbf{r},\alpha)\mid_{\zeta=\hat{\zeta}_{k-1},\alpha=\hat{\alpha}_{k-1}} \tag{11}$$

where $\mu(k)$ is the adaptation step-size. Since the channel estimator is assumed to have no *apriori* knowledge of the pdf $f_\epsilon(.)$, this unknown pdf is then estimated by using the kernel density estimator eq. (8) with Gaussian kernels as shown below. As the kernel estimators are known to be effective in density estimation over short data record, we use this technique over the available data (error) record, of length M, to estimate the unknown density. Using the kernel density estimator [9] we obtain:

$$\hat{f}_\epsilon(\epsilon) = \frac{1}{M}\sum_{j=1}^{M}K(\epsilon - \epsilon(j)) \tag{12}$$

Thus the estimated (joint) log-likelihood function can be written as:

$$\begin{aligned}&\hat{\mathcal{L}}(\alpha,\zeta \mid \mathbf{r})\mid_{\alpha=\hat{\alpha}_{k-1},\zeta=\hat{\zeta}_{k-1}}\\ &= \sum_{i=1}^{M}\log\left(\frac{1}{M}\sum_{j=1}^{M}K(\epsilon(i)-\epsilon(j))\right)\\ &= \sum_{i=1}^{M}\log\sum_{j=1}^{M}K(\epsilon(i)-\epsilon(j)) - \log\mid M\mid\end{aligned} \tag{13}$$

The gradient $\hat{\alpha}$ of the log-likelihood can be formulated as:

$$\begin{aligned}\nabla_\alpha\hat{\mathcal{L}}(\alpha \mid \mathbf{r},\zeta)\mid_{\alpha=\hat{\alpha}_{k-1},\zeta=\hat{\zeta}_{k-1}} &= \frac{\partial}{\partial\alpha}\hat{\mathcal{L}}(\alpha \mid \mathbf{r},\zeta)\Big|_{\alpha=\hat{\alpha}_{k-1},\zeta=\hat{\zeta}_{k-1}}\\ &= \sum_{i=1}^{M}\frac{\sum_{j=1}^{M}\frac{\partial}{\partial\alpha}K(\epsilon(i)-\epsilon(j))}{\sum_{k=1}^{M}K(\epsilon(i)-\epsilon(k))}\end{aligned} \tag{14}$$

Similarly gradient for $\hat{\zeta}$:

$$\begin{aligned}\nabla_\zeta\hat{\mathcal{L}}(\zeta \mid \mathbf{r},\alpha)\mid_{\zeta=\hat{\zeta}_{k-1},\alpha=\hat{\alpha}_{k-1}} &= \frac{\partial}{\partial\zeta}\hat{\mathcal{L}}(\zeta \mid \mathbf{r},\alpha)\Big|_{\zeta=\hat{\zeta}_{k-1},\alpha=\hat{\alpha}_{k-1}}\\ &= \sum_{i=1}^{M}\frac{\sum_{j=1}^{M}\frac{\partial}{\partial\zeta}K(\epsilon(i)-\epsilon(j))}{\sum_{k=1}^{M}K(\epsilon(i)-\epsilon(k))}\end{aligned} \tag{15}$$

Thereby substituting the estimated gradients in eq. (10) and (11) respectively, and iterating till $\hat{\alpha}_k$ and $\hat{\zeta}_k$ converge we get the ML estimated channel $\mathbf{h}$ by deconvolution. The algorithm is initialized by the LS estimate and μ as explained in [1] and [2] respectively. During simulations we

did not converge to a local maxima, however this is not always gauranteed (as with most iterative techniques). However, initializing the channel and whitening filter's taps with the LS estimate reduces the chances of converging to local maxima.

Two possible update methods could be used here to maximize the likelihood. In the first method, we update $\hat{\alpha}$ then update $\hat{\zeta}$ and then estimate the updated likelihood for the next update. The procedure is repeated till both $\hat{\alpha}$ and $\hat{\zeta}$ converge. Second update method is to first have $\hat{\alpha}$ converged and then have $\hat{\zeta}$ converged given the converged $\hat{\alpha}$ on estimated likelihood, and then iterate till no significant change is observed in $\hat{\alpha}$ and $\hat{\zeta}$. In this paper we use the first method of update.

5. SIMULATION RESULTS

For simulation study, we assume a communication channel model, like global system for mobile (GSM), considering CCI with Gaussian noise as a multi-modal, iid, Gaussian mixture interference as discussed in [1]. The performance of channel estimator is calculated by normalized-mean square error (NMSE), as shown in eq.(16).

$$NMSE = \frac{\mathbb{E}\{(h-\hat{h})^2\}}{\mathbb{E}\{h^2\}} \tag{16}$$

where h is the actual channel and $\hat{h}$ is the estimated channel (after deconvolution). For all simulation results, the input symbols of length 100 and ensemble of 1000-runs is considered.

A typical communication system effected by co-channel interference is shown in Fig. 2. The co-channels are each of order $N-1$ and are represented as h_i and interfering signal as a_i for $i = 2, ..., I$, where $I-1$ represents number of interferers. The received signal can be represented as

$$\begin{aligned} r(k) &= \sum_{n=0}^{N-1} h_1(n)a_1(k-n) + \sum_{i=2}^{I}\sum_{n=0}^{N-1} h_i(n)a_i(k-n) + n(k) \\ &= \sum_{n=0}^{N-1} h_1(n)a_1(k-n) + w(k) \end{aligned} \tag{17}$$

where the middle (double summation) term on the RHS in eq. (17) represents the CCI and $n(k)$ is a zero mean, iid, Gaussian noise process and $k = 1, \ldots, M$ represents the number of symbols.

The above presented algorithm is verified for real stationary channel for $N = 5$. The input signal is anti-podal random input sequence. The channels are assumed to be

h_1 = [-0.227 0.460 0.688 0.460 -0.227] and

h_2 = [1.0 0.8 0.6 0.4 0.2] where h_1 suffers from amplitude and phase distortion [10], and h_2 is the co-channel considered for the simulation.

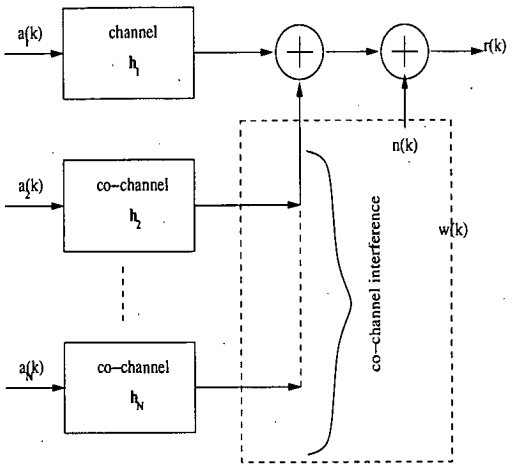


Fig. 2. A typical CCI effected communication system

Fig. 3 depicts the performance plot for the channel estimator presented in this paper. The legends 'LS', 'LS_{white}' and '$NPML_{white}$' represent LS without LPE filter, LS with LPE filter and NPML with LPE filter respectively. To observe the proformance of the algorithm, the signal to noise ratio (SNR) is kept fixed at 30dBs while signal to interference ratio (SIR) is varied from -13dBs to 9dBs. We can observe that by using the LPE filter with NPML based technique we can gain upto 3.5 dBs at NMSE of 10^{-1}.

6. CONCLUSION

It was shown that after noise whitening better channel estimates can be obtained. It was reconfirmed that the LS estimate with LPE filter produces better channel estimates for interference limited channels than LS estimate without LPE filter. Due to practicle constriants, the Gaussian assumption on the whitened noise is not gauranteed, hence a kernel density estimate based ML channel estimator was proposed. From Fig. 3 we observe that better channel estimates can be obtained by jointly estimating the whitening filter and the channel estimates by using kernel density estimator. Thus by combining kernel density estimator with whitening filter forms a robust channel estimator for interference limited communication channels.

Acknowledgment

The authors would like to thank UK-EPSRC for supporting this work.

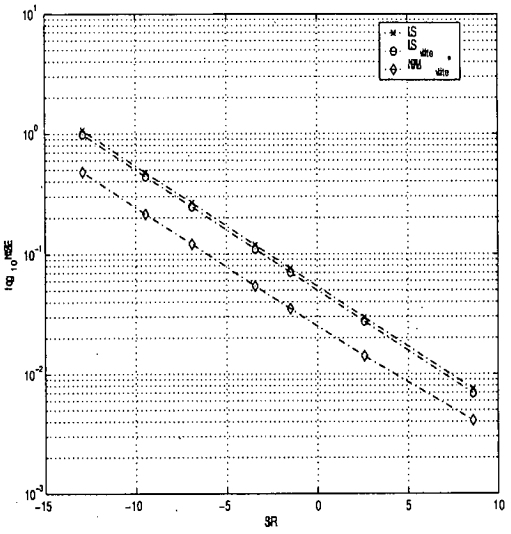


Fig. 3. MSE plot for co-channel effected communication system where h_1 = [-0.227 0.460 0.688 0.460 -0.227], SNR=30dBs for 100-symbols over an ensemble of 1000-runs

7. REFERENCES

[1] C. Luschi and B. Mulgrew, "Nonparametric trellis equalization in the presence of non-Gaussian interference," *IEEE Transactions on Communications*, vol. 51, no. 2, pp. 229–239, Feb 2003.

[2] V. Bhatia and B. Mulgrew, "A EM-based channel estimator in non-Gaussian noise," in *VTC-2004 (Fall), LA, US*, Sept 2004.

[3] D. Astely and B. Ottersten, "MLSE and spatio-temporal interference rejection combining with antenna arrays," in *Proc. EUSIPCO*, Sept 1998, vol. III, pp. 1341–1344.

[4] J-Wei Liang, Jiunn-Tsair Chen, and A. J. Paulraj, "A two-stage hybrid approach for CCI/ISI reduction with space time processing," *IEEE Communications Letters*, vol. 1, no. 6, pp. 163–165, Nov 1997.

[5] E.E. Kuruoglu, C. Molina, and W.J. Fitzgerald, "Approximation of alpha-stable probability densities using finite gaussian mixtures," in *Proceedings of EUSIPCO 98, Signal Processing IX: Theories and Applications*, Sept 1998, vol. 2, pp. 989–992.

[6] R.J. Kozick, R.S. Blum, and B.M. Sadler, "Signal processing in non-Gaussian noise using mixture distributions and the EM algorithm," in *Thirty-First Asilomar Conference on Signals Systems and Computers*, Nov 1997, vol. 1, pp. 438–442.

[7] J. K. Tugnait, Tong Lang, and Ding Zhi, "Single-user channel estimation and equalization," *IEEE Signal Processing Magazine*, vol. 17, no. 3, pp. 17–28, May 2000.

[8] B. W. Silverman, Ed., *Density Estimation for Statistics and Data Analysis*, Chapman Hall, London, 1986.

[9] N. N. Schraudolph, "Gradient-Based Manipulation of Non-parametric Entropy Estimates," *IEEE Transaction on Neural Networks*, vol. 15, no. 4, Jul 2004.

[10] J. G. Proakis, Ed., *Digital Communications*, McGraw-Hill, 1989.

References

[1] R. Lucky, "Automatic equalization of digital communication," *Bell System Tech. J.*, vol. 44, pp. 547–588, Apr. 1965.

[2] B. Widrow and M. Hoff(Jr), "Adaptive switching circuits," *IRE WESCON Conv.*, vol. 4, pp. 94–104, May. 1960.

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

[4] G. D. Forney, "The Viterbi algorithm," *Proceedings of the IEEE*, vol. 61, pp. 268–278, Mar. 1973.

[5] F. R. M. Jr and J. G. Proakis, "Adaptive maximum-likelihood sequence estimation for digital signaling in the presence of intersymbol interference," *IEEE Transaction on Information Theory*, vol. IT-19, pp. 120–124, Jan. 1973.

[6] N. H. Nedev, ed., *Analysis of the Impact of Impulsive Noise in Digital Subscriber Line Systems*. PhD Thesis, Department of Electronics and Electrical Engineering, The University of Edinburgh, 2003.

[7] S. K. Patra, ed., *Development of Fuzzy Based channel equalisers*. PhD Thesis, Department of Electronics and Electrical Engineering, The University of Edinburgh, 1998.

[8] C. Luschi, ed., *Probabilistic Techniques for Equalization of the Mobile Radio Channel in the Presence of Co-Channel Interference*. PhD Thesis, Department of Electronics and Electrical Engineering, The University of Edinburgh, 2002.

[9] A. Georgiadis and B. Mulgrew, "Adaptive Bayesian decision feedback equaliser for alpha-stable noise environments," *Signal Processing*, vol. 81, pp. 1603–1623, Aug. 2001.

[10] R. S. Blum, R. J. Kozick, and B. M. Sadler, "An adaptive spatial diversity receiver for non-Gaussian interference and noise," *IEEE Trans on Signal Processing*, vol. 47, pp. 2100–2111, Aug. 1999.

[11] J. Seo, S. Cho, and K. Feher, "Impact of non-Gaussian impulsive noise on the performance of high-level QAM," *IEEE Transactions on Electromagn. Compat.*, vol. EMC-31, pp. 177–180, May. 1989.

[12] L. Izzo and L. Paura, "Error probability for fading CPSK signals in Gaussian and impulsive atmospheric noise environments," *IEEE Transactions on Aerospace and Electronic Systems*, vol. AES, pp. 719–722, Sep. 1981.

[13] A. T. Georgiadis, ed., *Adaptive Equalisation for Impulsive Noise Environments*. PhD Thesis, Department of Electronics and Electrical Engineering, The University of Edinburgh, 2000.

[14] S. Chen, B. Mulgrew, E. S. Chng, and G. J. Gibson, "Space translation properties and the minimum-BER linear-combiner DFE," *IEE Proc. Communications*, vol. 145, pp. 316–322, Oct. 1998.

[15] B. Mulgrew and S. Chen, "Adaptive minimum-BER decision feedback equalisers for binary signalling," *Signal Processing*, vol. 81, pp. 1479–1489, Jul. 2001.

[16] S. Haykins, ed., *Adaptive Filter Theory 4 ed.* Pearson Education Asia Inc., 2002.

[17] B. Widrow, ed., *Adaptive Signal Processing*. New Jersey: Prentice-Hall, Inc., 1985.

[18] B. Mulgrew, P. Grant, and J. Thompson, eds., *Digital Signal Processing 2 ed.* Palgrave Macmillan, 2003.

[19] Virginia Tech. http://scholar.lib.vt.edu/theses/available/ etd-122099-153321/ unrestricted/ Chapter01.pdf.

[20] J. G. Proakis, ed., *Digital Communications*. McGraw-Hill, 1989.

[21] M. E. Austin, "Decision-feedback equalisation for digital communication over dispersive channels," *MIT Lincoln Labs, Lexington, M.A.*, vol. Tech. Rep 437, Aug. 1967.

[22] C. A. Belfiore and J. J. H. Park, "Decision feedback equalisation," *Proc. IEEE*, vol. 67, pp. 1143–1156, 1979.

[23] J. Salz, "Optimum mean-square decision feedback equalisation," *Bell System Technical Journal*, vol. 52, pp. 1341–1373, 1973.

[24] L. Ahlin and J. Zander, eds., *Principles of Wireless Communications 2 ed.* Sweden: Studentlitteratur, 1997.

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

[26] E. Lee and D. G. Messerschmitt, eds., *Digital Communication*. London: Kluwer Academic Publishers, 1988.

[27] E. Sharmash and K. Yao, "On the structure and performance of a linear decision feedback equalizer based on the minimum error probability criterion,," in *IEEE ICC '74*, 1974.

[28] C.-C. Yeh and J. R. Barry, "Adaptive minimum bit-error rate equalization for binary signalling," *IEEE Transactions on Communications*, vol. 48 (7), pp. 1226–1235, Jul. 2000.

[29] S. Chen, B. Mulgrew, and L. Hanzo, "Least bit error rate adaptive nonlinear equaliser for binary signalling," *IEE Proc. Comms*, vol. 150, pp. 29 – 36, Feb. 2003.

[30] S. Chen, N. N. Ahmad, and L. Hanzo, "Adaptive minimum bit-error rate beamforming," *IEEE Trans. on Wireless Comms.*, vol. 4, pp. 341 – 348, Mar. 2005.

[31] S. Chen, L. Hanzo, N. N. Ahmad, and A. Wolfgang, "Adaptive minimum bit error rate beamforming assisted QPSK receiver," in *IEEE ICC'04*, vol. 6, pp. 3389–3393, Jun. 2004.

[32] S. Chen, A. K. Samingan, B. Mulgrew, and L. Hanzo, "Adaptive minimum-BER linear multiuser detection for DS-CDMA signals in multipath channels," *IEEE Trans. on Signal Processing*, vol. 49, pp. 1240 – 1247, Jun. 2001.

[33] R. C. de Lamare and R. Sampaio-Neto, "Adaptive multiuser receivers for DS-CDMA using minimum BER gradient-Newton algorithms," in *IEEE PMIRC '02*, vol. 3, pp. 1290–1294, Sep. 2002.

[34] M. Y. Alias, A. K. Samingan, S. Chen, and L. Hanzo, "Multiple antenna aided OFDM employing minimum bit error rate multiuser detection," *Electronics Letters*, vol. 39, pp. 1769 – 1770, Nov. 2003.

[35] T. S. Rappaport, ed., *Wireless Communications, Principles and Practice*. New York: IEEE Press, 1996.

[36] D. N. Godard, "Self-recovering equalization and carrier tracking in two-dimensional data communication systems," *IEEE Trans. Commun.*, vol. COM-28, pp. 1867–1875, Nov. 1980.

[37] G. B. Giannakis and J. M. Mendel, "Identification of non-minimum phase systems using higher-order statistics," *IEEE Trans. Acoust. Speech, Signal Process.*, vol. 37, pp. 360–377, Mar. 1989.

[38] J. K. Tugnait, "Blind equalization and estimation of digital communication FIR channels using cumulant matching," *IEEE Trans. Commun.*, vol. 43, pp. 1240–1245, Feb./Mar./Apr. 1996.

[39] J. K. Tugnait, T. Lang, and D. Zhi, "Single-user channel estimation and equalization," *IEEE Signal Processing Magazine*, vol. 17, pp. 17–28, May. 2000.

[40] X. Li, "Blind channel estimation and equalization in wireless sensor networks based on correlations among sensors," *IEEE Trans. Signal Process.*, vol. 53, pp. 1511–1519, Apr. 2005.

[41] P. Stoica and O. Besson, "Training sequence design for frequency offset and frequency-selective channel estimation," *IEEE Trans. Comms.*, vol. 51, pp. 1910–1917, Nov. 2003.

[42] S. M. Kay, ed., *Fundamentals of Statistical Signal Processing:Estimation Theory*. Prentice Hall International, Inc., 1993.

[43] H. Zamiri-Jafarian and S. Pasupathy, "EM-based recursive estimation of channel parameters," *IEEE Trans. Communications*, vol. 47, pp. 1297–1302, Sep. 1999.

[44] A. Jeremic, T. A. Thomas, and A. Nehorai, "OFDM channel estimation in presence of interference," *IEEE Transactions on Signal Processing*, vol. 52, pp. 3429–3439, Nov. 2004.

[45] T. K. Moon, "The expectation-maximization algorithm," *IEEE Signal Processing Magazine*, pp. 47–60, Nov. 1996.

[46] J. Bilmes, "A gentle tutorial of the EM algorithm and its application to parameter estimation for gaussian mixtures and hidden Markov models," *International Computer Science Institute*, vol. TR-97-021, Apr. 1998.

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

[48] E. D. Carvalho and D. T. M. Slock, "Cramer-Rao bounds for semi-blind, blind and training sequence based channel estimation," in *IEEE Workshop on Signal Proc. Advances in Wireless Communications*, pp. 129–132, Apr. 1997.

[49] E. D. Carvalho and D. T. M. Slock, "Maximum likelihood blind FIR multi-channel estimation with Gaussian prior for the symbols," in *Proc. ICASSP*, pp. 3593–3596, Apr. 1997.

[50] A. Gorokhov and P. Loubaton, "Semi-blind second order identification of convolutive channels," in *Proc. ICASSP*, pp. 3905–3908, Apr. 1997.

[51] H. A. Cirpan and M. K. Tsatsanis, "Stochastic maximum likelihood methods for semi-blind channel estimation," *IEEE Signal Proc. Letters*, vol. 5, pp. 21–24, Jan. 1998.

[52] L. E. Baum, T. Petrie, G. Soules, and N. Weiss, "A maximization technique occuring in the statistical analysis of probabilistic functions of Markov chains," *Ann. Math. Stat.*, vol. 41, pp. 164–171, 1970.

[53] M. Erkurt and J. G. Proakis, "Joint data detection and channel estimation for rapidly fading channels," in *Proc. GLOBECOM'92*, pp. 910–914, 1992.

[54] G. K. Kaleh and R. Vallet, "Joint parameter estimation and symbol detection for linear or nonlinear unknown channels," *IEEE Transaction on Communications*, vol. 42, pp. 2406–2413, Jul. 1994.

[55] S. Perreau, L. B. White, and P. Duhamel, "An equalizer including a soft channel decoder," in *Proc. IEEE Workshop on Signal Processing Advances in Wireless Communications*, pp. 9–12, Apr. 1997.

[56] L. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proc. IEEE*, vol. 77, no. 2, pp. 257–286, 1989.

[57] G. Li and Z. Ding, "Semi-blind channel identification for individual data burst in GSM wireless systems," *Signal Processing*, vol. 80, pp. 2017–2031, Oct. 2000.

[58] G. Leus, "Semi-blind channel estimation for rapidly time-varying channels," in *IEEE ICASSP'05*, vol. 3, pp. 773–776, Mar. 2005.

[59] A. M. Kuzminskiy, F. J. Mullany, and C. B. Papadias, "Semi-blind channel estimation at the receiver for steered-STS transmit antenna architecture in CDMA2000," in *IEEE ICASSP'05*, vol. 3, pp. 565–568, Mar. 2005.

[60] T. Cui and C. Tellambura, "Semi-blind channel estimation and data detection for OFDM systems over frequency-selective fading channels," in *IEEE ICASSP'05*, vol. 3, pp. 597–600, Mar. 2005.

[61] K. Li and H. Liu, "On estimation of multipath channels and residual carriers in CDMA communications," in *IEEE Vehicular Technology Conference*, no. 1, pp. 557–561, May. 1998.

[62] R. Kozick and B. Sadler, "Maximum-likelihood array processing in non-Gaussian noise with Gaussian mixtures," *IEEE Transactions on Signal Processing*, vol. 48, pp. 3520–3535, Dec. 2000.

[63] A. Hafeez, K. J. Molnar, H. Arslan, G. E. Bottomley, and R. Ramsh, "Adaptive joint detection of co-channel signals for TDMA handsets," *IEEE Trans on Communications*, vol. 52, pp. 1722–1732, Oct. 2004.

[64] H. Schoeneich and P. A. Hoeher, "Single antenna interference cancellation: iterative semi-blind algorithm and performance bound for joint maximum-likelihood interference cancellation," in *IEEE GLOBECOM '03*, vol. 3, pp. 1716–1720, Dec. 2003.

[65] J.-W. Liang, J.-T. Chen, and A. J. Paulraj, "A two-stage hybrid approach for CCI/ISI reduction with space time processing," *IEEE Communications Letters*, vol. 1, pp. 163–165, Nov. 1997.

[66] D. Astely and B. Ottersten, "MLSE and spatio-temporal interference rejection combining with antenna arrays," in *Proc. EUSIPCO*, vol. III, pp. 1341–1344, Sep. 1998.

[67] J. G. Proakis and J. H. Miller, "An adaptive receiver for digital signaling through channels with intersymbol interference," *IEEE Transaction on Information Theory*, vol. IT-15, pp. 484–497, Jul. 1969.

[68] G. A. Tsihrintzis and C. L. Nikias, "Performance of optimum and suboptimum receivers in the presence of impulsive noise modelled as an alpha-stable process," *Proceedings of MILCOM'93*, vol. 2, pp. 658–662, Jul. 1993.

[69] L. F. Lind and N. A. Mufti, "Efficient method for modelling impulsive noise in a communication system," *IEE Electronics Letters*, vol. 32, pp. 1440–1441, Aug. 1996.

[70] I. Mann, S. McLaughlin, W. Henkel, R. Kirby, and T. Kessler, "Impulse generation with appropriate amplitude, length, inter-arrival and spectral characteristics," *IEEE Journal on Select Areas in Comm.*, vol. 20, pp. 901–912, Jun. 2002.

[71] M. Shao and C. L. Nikias, "Signal processing with fractional lower order moments: Stable processes and their applications," *Proceedings of the IEEE*, vol. 81, pp. 986–1009, Jul. 1993.

[72] O. Arikan, A. E. Cetin, and E. Erzin, "Adaptive filtering for non-Gaussian stable processes," *IEEE Signal Processing Letters*, vol. 1, pp. 163–165, Nov. 1994.

[73] A. Swami and B. Sadler, "On some detection and estimation problems in heavy-tailed noise," *Signal Processing*, vol. 82, pp. 1829–1846, Dec. 2002.

[74] H. Bergstrom, "On some expansions of stable distribution functions," *Ark. Math.*, vol. 2, pp. 375–378, 1954.

[75] A. P. Clark, L. H. Lee, and R. S. Marshall, "Developments of the conventional non-linear equaliser," *IEE Proc. Part F*, vol. 129, no. 2, pp. 85–94, 1982.

[76] E. S. Chng, ed., *Applications of non-linear filters with the linear-in-the-parameter structure*. PhD Thesis, Department of Electronics and Electrical Engineering, The University of Edinburgh, 1995.

[77] S. Chen, E. S. Chng, B. Mulgrew, and G. J. Gibson, "Miminum-BER linear-combiner DFE," in *ICC'96*, vol. 2, pp. 1173–1177, Jun. 1996.

[78] J. M. Chambers, C. L. Mallows, and B. W. Stuck, "A method for simulating stable random variables," *Journal of the American Statistical Association*, vol. 71, pp. 340–344, Jun. 1976.

[79] S. Bates, ed., *Traffic characterisation and modelling for call admission control schemes on asynchronous transfer mode networks*. PhD Thesis, Department of Electronics and Electrical Engineering, The University of Edinburgh, 1997.

[80] E. Kuruoglu, C. Molina, and W. Fitzgerald, "Approximation of alpha-stable probability densities using finite Gaussian mixtures," in *Proceedings of EUSIPCO 98, Signal Processing IX: Theories and Applications*, vol. 2, pp. 989–992, Sep. 1998.

[81] R. Kozick, R. Blum, and B. Sadler, "Signal processing in non-Gaussian noise using mixture distributions and the EM algorithm," in *Thirty-First Asilomar Conference on Signals Systems and Computers*, vol. 1, pp. 438–442, Nov. 1997.

[82] C. Luschi and B. Mulgrew, "Non-parametric trellis equalization in the presence of non-Gaussian interference," *IEEE Transactions on Communications*, vol. 51, pp. 229–239, Feb. 2003.

[83] H. Zamiri-Jafarian and S. Pasupathy, "Recursive channel estimation for wireless communication via the EM algorithm," in *IEEE Conference on Personal Wireless Communications*, pp. 33–37, Dec. 1997.

[84] T. Hunziker, M. Hashiguchi, and T. Ohira, "Reception of coded OFDM signals in broad-band fading environments with strong co-channel interference," in *Proc. GLOBECOM'03*, vol. 4, pp. 2310–2314, Dec. 2003.

[85] D. Sengupta and S. M. Kay, "Efficient estimation of parameters for non-Gaussian autoregressive processes," *IEEE Acoustics, Speech, and Signal Processing*, vol. 37, pp. 785–794, Jun. 1989.

[86] Y. Zhang and R. S. Blum, "An adaptive receiver with an antenna array for channels with correlated non-Gaussian interference and noise using the SAGE algorithm," *IEEE Trans on Signal Processing*, vol. 48, pp. 2172–2175, Jul. 2000.

[87] V. Bhatia and B. Mulgrew, "A EM-based channel estimator in non-Gaussian noise," in *VTC-2004 (Fall), LA, US*, Sep. 2004.

[88] R. A. Redner and H. F. Walker, "Mixture densities, maximum likelihood and the EM algorithm," *SIAM Rev.*, vol. 26, pp. 195–239, Apr. 1984.

[89] J. A. Fessler and A. O. Hero, "Space-alternating generalized EM algorithm," *IEEE Trans. Signal Processing*, vol. 42, pp. 2664–2677, Oct. 1994.

[90] M. Feder and A. Catipovic, "Algorithms for joint channel estimation and data recovery–Application to equalization in underwater communications," *IEEE J. Oceanic Eng.*, vol. 16, pp. 42–55, Jan. 1991.

[91] D. Erdogmus, K. E. I. Hild, and J. C. Principe, "Online entropy manipulation: Stochastic information gradient," *IEEE Signal Processing Letters*, vol. 10, pp. 242–245, Aug. 2003.

[92] N. N. Schraudolph, "Gradient-based manipulation of non-parametric entropy estimates," *IEEE Transaction on Neural Networks*, vol. 15, Jul. 2004.

[93] D. Sengupta and S. M. Kay, "Parameter estimation and GLRT detection in colored non-Gaussian autoregressive processes," *IEEE Acoustics, Speech, and Signal Processing*, vol. 38, pp. 1661–1676, Oct. 1990.

[94] A. Swami, "Cramer-Rao bounds for deterministic signals in additive and multiplicative noise," *Signal Processing*, vol. 53, pp. 231–224, 1996.

[95] A. Spaulding and D. Middleton, "Optimum reception in an Impulsive Interference Environment-Part-I," *IEEE Trans. on Comm.*, vol. COM-25, pp. 910–923, Sep. 1977.

[96] L. A. Liporace, "Maximum likelihood estimation for multivariate observations of markov sources," *IEEE Trans. Inf. Thoery*, vol. 1T-28, pp. 729–734, Sep. 1982.

[97] E. Parzen, "On the estimation of a probability density function and mode," *Annals of Mathematical Statistics*, vol. 33, pp. 1065–1076, 1962.

[98] B. W. Silverman, ed., *Density Estimation for Statistics and Data Analysis*. London: Chapman Hall, 1986.

[99] C. E. Shannon, "A mathematical theory of communications," *Bell System Tech. J.*, vol. 27, pp. 379–423 and 623–656, 1948.

[100] D. Erdogmus and J. C. Principe, "An error-entropy minimization algorithm for supervised training of nonlinear adaptive systems," *IEEE Transactions on Signal Processing*, vol. 50, pp. 1780–1786, Jul. 2002.

[101] J. Beirlant, E. Dudewicz, L. Gyorfi, and E. van der Meulen, "Non-parametric entropy estimation: An overview," *Intl. Journal of Mathematics and Statistical Sciences*, vol. 7, pp. 17–39, Jun. 1997.

[102] P. Chen and H. Kobayashi, "Maximum likelihood channel estimation and signal detection for OFDM systems," in *IEEE International Conference on Communications (ICC)*, vol. 3, pp. 1640–1645, Apr./May. 2002.

[103] I. Bradaric and A. P. Petropulu, "Performance of training-based OFDM systems in the presence of time varying frequency-selective channels," in *Proc. IEEE ICASSP-2004*, May. 2004.

[104] Y. Li, "Pilot-symbol-aided channel estimation for OFDM in wireless systems," *IEEE Trans. Veh. Technol.*, vol. 49, pp. 1207–1215, Jul. 2000.

[105] J. Park, D. Kim, C. Kang, and D. Hong, "Effect of Bluetooth interference on OFDM-based WLAN," in *Proc. VTC (Fall)*, vol. 2, pp. 786–789, Oct. 2003.

[106] C. F. Chiasserini and R. R. Rao, "Performance of IEEE 802.11 WLANs in a Bluetooth environment," in *Proc. IEEE WCNC-2000*, vol. 1, pp. 94–99, Sep. 2000.

[107] A. Jeremic, T. A. Thomas, and A. Nehorai, "OFDM channel estimation in presence of asynchronous interference," in *Proc. ICASSP*, vol. 4, pp. 684–687, Apr. 2003.

[108] M. Speth, S. A. Fechtel, G. Fock, and H. Meyr, "Optimum receiver design for wireless broad-band systems using OFDM-part I," *IEEE Transactions on Communications*, vol. 47, pp. 1668–1677, Nov. 1999.

[109] Z. Jie, Z. Yun, S. Ling, and Z. Ping, "Effects of frequency-offset on the performance of OFDM systems," in *Proc. ICCT*, vol. 2, pp. 1029–1032, Apr. 2003.

[110] H. Harada and R. Prasad, eds., *Simulation and Software Radio for Mobile Communications*. London: Artech House, 2002.

[111] J.-J. van de Beek, O. Edfors, and M. Sandell, "On channel estimation in OFDM systems," in *VTC 1995*, vol. 2, pp. 815–819, Jul. 1995.

[112] O. Edfors, M. Sandell, J.-J. van de Beek, S. Wilson, and P. Borgesson, "OFDM channel estimation by singular value decomposition," *IEEE Transactions on Communications*, vol. 46, pp. 931–939, Jul. 1998.

[113] P. He, K.-U. Schmidt, C. K. Ho, and S. Sun, "Iterative channel estimator and equalizer for OFDM modulation systems," in *Proc. IEEE VTC-2003 (Spring)*, vol. 2, pp. 1313–1317, Apr. 2003.

[114] X. Ma, H. Kobayashi, and S. C. Schwartz, "EM-based channel estimation algorithms for OFDM," *Eurasip Journal on Applied Signal Processing*, vol. 2004, pp. 1460–1477, Aug. 2004.

[115] V. Bhatia and B. Mulgrew, "Non-parametric maximum likelihood channel estimator in the presence of non-Gaussian noise," *IEEE Trans. Signal Processing (submitted)*, 2005.

[116] A. Leke and J. M. Cioffi, "Impact of imperfect channel knowledge on the performance of multicarrier systems," in *Proc. GLOBECOM 98*, vol. 2, pp. 951–955, Nov. 1998.

[117] Multiband OFDM alliance SIG, "Multiband OFDM physical layer proposal for IEEE 802.15 task group 3a," (2004, Sep.) [Online] Available: http://www.multibandofdm.org/papers/MultiBand_OFDM_Physical_Layer_Proposal_for_IEEE_802.15.3a_Sept_04.pdf.

[118] C. Fleming, "A tutorial on convolutional coding with viterbi decoding," (2003, Jan.) [Online] Available:http://home.netcom.com/ chip.f/viterbi/tutorial.html.

[119] D. Gesbert (et al), "From theory to practice: An overview of MIMO space-time coded wireless systems," *IEEE JSAC*, vol. 21, pp. 281–301, Apr. 2003.

[120] G. J. Foschini and M. J. Gans, "On limits of wireless communications in a fading environment when using multiple antennas," *Wireless Pers. Commun.*, vol. 6, pp. 311–335, Mar. 1998.

[121] R. Duda, P. Hart, and D. Stork, eds., *Pattern Classification 2nd ed.* John Wiley and Sons, Inc, 2001.

[122] D. Erdogmus and J. C. Principe, "Convergence properties and data efficiency of the minimum error entropy criterion in Adaline training," *IEEE Transactions on Signal Processing*, vol. 51, pp. 1966–1978, Jul. 2003.

[123] D. Cox and D. Hinkley, eds., *Theoretical Statistics*. London: Chapman Hall, 1974.

[124] I. A. Rezek and S. J. Roberts, "Parametric model order estimation: a brief review," *IEE Colloquium on the Use of Model Based Digital Signal Processing Techniques in the Analysis of Biomedical Signals*, pp. 3/1–3/6, Apr. 1997.

[125] H. Akaike, "A new look at the statistical model identification," *IEEE Trans Automat. Contr*, vol. AC-19, pp. 716–723, Dec. 1974.

[126] J. Rissanen, "Modeling by shortest data description," *Automatica*, 1978.

[127] S. K. Katsikas, S. D. Likothanassis, and D. G. Lainiotis, "AR model identification with unknown process order," *IEEE Transactions on Acoustics Speech & Signal Processing*, 1990.

[128] Y. Gong and C. F. N. Cowan, "Structure adaptation of linear MMSE adaptive filters," *IEE Proceedings - Vision, Image and Signal Processing*, vol. 151, pp. 271–277, Aug. 2004.

[129] T. Matsuoka and T. J. Ulrych, "Information theory measures with application to model identification," *IEEE Trans ASSP*, vol. ASSP-34, pp. 511 – 517, Jun. 1986.