DIGITAL DATA TRANSMISSION OVER AN HF CHANNEL

by

ZIAD RIZK

A Doctoral Thesis submitted in partial fulfillment

of the requirements for the award

of Doctor of philosophy

of the Loughborough University of Technology

August 1996

Supervisor : Dr. W. G. Marshall

Department of Electronic and Electrical Engineering

Loughborough University of Technology

CONTENTS

ABSTRACT

The thesis is concerned with detection, estimation techniques and a method of the adaptive adjustment of the equaliser, for use in a *4800bit/sec* synchronous digital transmission system operating over a voice-band time-varying HF channel. Two main impairments are additive Gaussian noise and inter-symbol interference (ISI), which can be very severe at times. All techniques considered here are algorithms or processes that operate on sequences of sample values. Modern digital modems normally operate in this way, and the techniques described are of direct application to practical systems, and could be implemented using the new technology of high speed real-time digital signal processing (DSP). The performance of the various systems that employ the above techniques are obtained using the computer simulated model of three types of HF channels.

The ionospheric propagation medium, the characteristics of HF channel and the signal distortion introduced by the channel are first described. The thesis then presents a suitable base-band model of the HF channel for computer simulation of quadrature amplitude modulation systems. A suitable method for the adjustment of the receiver is described next. This method is suitable both for the adjustment of a conventional decision feedback equaliser (DFE), and also for the adjustment of a linear feedforward filter that is employed ahead of a near-maximum likelihood (NML) detector. This method uses a minimum phase (root-finding) algorithm (MPA) to convert the channel response from being non-minimum phase to at least approximately minimum phase. The results of computer simulation tests of this algorithm are then presented over different types of HF channel models. The results demonstrate the algorithm's capability to make the channel response minimum (or near-minimum) phase.

Various NML detectors, derived from the Viterbi detector, are discussed. Each detector is here preceded by an adaptive linear filter that is adjusted adaptively using an MPA. The performance of these detectors is compared with the conventional DFE, whose tap-gains are adjusted adaptively using an MPA, and the detector which gives the best compromise between performance and complexity is selected for combined receivers. These results are obtained assuming perfect estimation of the channel response.

The estimation techniques studied in this thesis include both new and conventional estimators, which are based on the least-mean-square (LMS) algorithm or recursive-least-square (RLS) algorithm. The estimator provides an estimate of the sampled impulse

response (SIR) of the channel, necessary for the NML detector or MPA. The performances of these estimators are compared using computer simulation tests. The results also demonstrate that the simpler LMS algorithm with adaptive step size gives a comparable level of accuracy with the more complex RLS algorithm.

Finally the most promising of the detectors and estimators are connected with an adaptive equaliser, using an MPA, to form a new combined receiver. The details of the combined system structure with its computational complexity are given. Extensive computer simulation tests have been carried out on the different arrangements of the combined system including DFE, when all the functions of detection, estimation and MPA are present, in order to find the most cost effective system in terms of performance and complexity. A considerable reduction in the equipment complexity can be achieved by allowing a long period between successive adjustment of the adaptive filter and estimator.

ACKNOWLEDGEMENTS

I would like to express my deep gratitude to my supervisor, Dr. W.G. Marshall for his help and guidance throughout the research.

I would like also to thank the Syrian government (Scientific Studies and Research Centre) for the financial support during my study in UK.

I would like to thank Dr. M. Wainakh for his help and supervision of part of the research in Syria.

Thanks are also due to various friends and colleagues for their kind encouragement throughout the duration of research. The help provided by the staff of computer centre and general office of electronic engineering department is fully appreciated.

Special thanks go to my wife and my parents for their patience, encouragement and help.

LIST OF PRINCIPAL SYMBOLS

a(t) and A(f)	Impulse response and transfer function of a filter.
|A(f)|	Absolute value of A(f).
$A_i(z)$	z-Transform of one-tap feedback filter in Fig.4.4.3.
$B_i(z)$	z-Transform of two-taps feedforward filter in Fig.4.4.3.
b	step size of LMS algorithm.
a(t)*b(t)	Convolution between a(t) and b(t).
C_i	Adaptive step size vector of LMS algorithm.
e_i	Error in the estimated value of received sample r_i.
E[.]	Expectation operator.
D(z)	z-Transform of pre-detection filter.
B_d, f_{sp}	Frequency spread or Doppler spread.
f_{rms}, f_e	Root-mean square frequency, and fade rate.
$\Im[.]$	Imaginary part of a complex number.
j	When used as a subscript is an integer, otherwise it is $\sqrt{-1}$.
K_i	Kalman gain vector.
(L+1)	Number of samples in the SIR of linear baseband channel.
n(t)	Additive white Gaussian noise with zero mean and two-sided power spectral density $N_0/2$.
$N_0/2$	Power spectral of n(t).
$\Re[.]$	Real part of a complex number.
r(t)	Received signal.
$\{r_i\}$	Sequence of received signal samples.
r_i'	Estimated received samples.
p(t)	Impulse response of linear baseband channel.
P_i	Sampled impulse response of linear baseband channel (SIR).
P'_i	Estimate of P_i at time t=iT.
$P'_{i+1,i}$	One-step prediction of P_{i+1} at time t=iT.
$P''_{i+1,i}$	Prediction of the rate of change with respect to i of P_{i+1}.
P(z)	z-Transforme of the SIR of the channel.
$P_1(z)$	Factor of P(z) with all its roots inside the unite circle.
$P_2(z)$	Factor of P(z) with all its roots outside the unite circle.
$q_n(t)$	Statistically independent random processes.
$\{q_{h,i}\}$	Sequence obtained by sampling $q_h(t)$.
s_i	Data symbol.
s'_i	Detected data symbol.
T	Sampling interval.

T_M	Multipath spread.
v(t)	Gaaussian random process with zero mean.
$\{v_i\}$	Resultant noise component in the received signal r_i.
$w_B = 1/T_B$	Bandwidth of transmitted signal.
Y_i	Minimum phased scaled version of the SIR of linear baseband channel P_i.
$\lambda,\lambda_1,\lambda_2$	Small positive constant in the range (0-1).
ζ_1	Mean-square error in the estimate (prediction) of P_i.
ζ_2	Mean-square normalised error in the estimate (prediction) of P_i.
σ^2_v	Variance of v(t) (or $\{v_i\}$).
$(\Delta t)_c$	Coherence time.
$(\Delta f)_c$	Coherence frequency.
Δ_i	Correction matrix of LMS algorithm.
ψ	Signal-to-noise ratio(SNR).
MPA	Minimum phase (root-finding) algorithm.
NML	Near-maximum likelihood.
SIR	Sampled impulse response of the linear base-band channel.
Estimator 1	Least-mean square "LMS" estimator.
Estimator 2	Estimator 1 with degree one prediction.
Estimator 3	Adaptive Least-mean square "LMS" estimator.
Estimator 4	Square-root Kalman "SRK" estimator.
Estimator 5	Estimator 4 with prediction.
Detector 3	Modified version of NML detector.
System 1	(*FLMS:D-0*) Combined detector 3, estimator 1 and adaptive filter.
System 2	(*FLMS:D-1*) Combined detector 3, estimator 2 and adaptive filter.
System 3	(*ALMS:D-0*) Combined detector 3, estimator 3 without prediction and adaptive filter.
System 4	(*ALMS:D-1*) Combined detector 3, estimator 3 with prediction and adaptive filter.
System 5	(*SRK:D-0*) Combined detector 3, estimator 4 and adaptive filter.
System 6	(*SRK:D-1*) Combined detector 3, estimator 5 and adaptive filter.
System 7	(*CNLEQ*) Combined non-linear (decision feedback) equaliser.

CHAPTER 1

INTRODUCTION

1.1 BACKGROUND

The HF channel has a particular and important role to play in long-distance communication, even after the introduction of several other kinds of transmission media such as coaxial cable, optical fiber, satellite links, etc. Due to multipath fading in the HF channel, the problem of reliable transmission of digital information over this channel has always presented the communicator with a significant challenge. Therefore satellite systems have received considerable attention for beyond line-of-sight reliable application over the last two decades. For the reasons of security, cost effectiveness and lack of vulnerability to attack, much research has been conducted into methods of extending the capability, reliability and useability of the HF fading channel for long-distance digital data transmission [6,107-109]. More recently, however a new wave of interest is in evidence and efficient signal processing techniques combined with high speed processing capabilities promise to provide the means to achieve high reliability HF data communication [25,31,107-114]. Therefore HF radio links are still very important for many users (countries) for point-to-point communication, military operations, commercial shipping, aircraft communication, etc. [6,107-109].

HF radio uses frequencies in the range of *2* to *30MHz*. At these frequencies, long-distance communication is achieved through refractive bending of the radio waves in the ionosphere from ionised layers at different elevations [6,4]. In most cases, more than one ionospheric "layer" causes the return of a refracted radio-wave to the receiver, thus the received signal is dispersed in time [1,6,4]. The principal problem in HF communication is that the supportive layer (or layers) changes with frequency of transmission, angle of incidence of the transmitted signal, the time of day (or night) and with the sunspot activity. Thus the characteristics of HF radio channels vary with time in quite an unpredictable manner, and this results in signals transmitted over these channels undergoing fading and multipath propagation [1,6]. Therefore, the HF channel is characterised as a multipath time-varying channel with time and frequency spreading. These characteristics can produce a complete loss of the transmitted signal. The detection of such a signal is thus adversely affected and as a result HF channels exhibit high error rates [1,6,23,26,107].

Due to the fading, time-dispersive characteristics of the HF radio channel, the transmission of digital data has been confined in the past to very low data rates (typically 50-75 bit/s). At these rates, the signal-element duration (T_B) with serial transmission greatly exceeds the multipath spread (T_M) of the channel, such that :

CHAPTER 1

INTRODUCTION

1.1 BACKGROUND

The HF channel has a particular and important role to play in long-distance communication, even after the introduction of several other kinds of transmission media such as coaxial cable, optical fiber, satellite links, etc. Due to multipath fading in the HF channel, the problem of reliable transmission of digital information over this channel has always presented the communicator with a significant challenge. Therefore satellite systems have received considerable attention for beyond line-of-sight reliable application over the last two decades. For the reasons of security, cost effectiveness and lack of vulnerability to attack, much research has been conducted into methods of extending the capability, reliability and useability of the HF fading channel for long-distance digital data transmission [6,107-109]. More recently, however a new wave of interest is in evidence and efficient signal processing techniques combined with high speed processing capabilities promise to provide the means to achieve high reliability HF data communication [25,31,107-114]. Therefore HF radio links are still very important for many users (countries) for point-to-point communication, military operations, commercial shipping, aircraft communication, etc. [6,107-109].

HF radio uses frequencies in the range of *2* to *30MHz*. At these frequencies, long-distance communication is achieved through refractive bending of the radio waves in the ionosphere from ionised layers at different elevations [6,4]. In most cases, more than one ionospheric "layer" causes the return of a refracted radio-wave to the receiver, thus the received signal is dispersed in time [1,6,4]. The principal problem in HF communication is that the supportive layer (or layers) changes with frequency of transmission, angle of incidence of the transmitted signal, the time of day (or night) and with the sunspot activity. Thus the characteristics of HF radio channels vary with time in quite an unpredictable manner, and this results in signals transmitted over these channels undergoing fading and multipath propagation [1,6]. Therefore, the HF channel is characterised as a multipath time-varying channel with time and frequency spreading. These characteristics can produce a complete loss of the transmitted signal. The detection of such a signal is thus adversely affected and as a result HF channels exhibit high error rates [1,6,23,26,107].

Due to the fading, time-dispersive characteristics of the HF radio channel, the transmission of digital data has been confined in the past to very low data rates (typically 50-75 bit/s). At these rates, the signal-element duration (T_B) with serial transmission greatly exceeds the multipath spread (T_M) of the channel, such that :

$$T_B >> T_M \qquad 1.1$$

In this case the signal is subjected to flat fading. With the appropriate signal design and detection process, the inter-symbol interference effects can now be reduced to negligible levels. However, with the increasing demand for HF communication, it has become necessary to increase the data transmission rate (greater than *300 bit/s*), and as a result the multipath spread can extend over the duration of several signal-elements. The system is now subject to inter-symbol interference (ISI) [1,3,6,20,29,107]. This is the condition caused by the spreading in time of an individual signal-element by the HF radio link resulting in the overlapping of an adjacent signal-element. The multipath delay spread can commonly be several milliseconds, resulting in the simultaneous reception of several separate signal-elements at the receiver. This pulse "smearing" is known as ISI and is the principle reason for the poor performance of simple high data rate transmission/reception methods over HF radio links [1,6,29]. As the speed of transmission increases, so do the bit errors, affecting both modem performance and reliability. This problem was initially circumvented by the use of parallel modems, where transmitting took place over a number of sub-channels within a *3KHz* band, at a fairly low baud rate (e.g. *75baud*) [115].

Parallel modems operate by splitting the *3KHz* channel into a number of sub-channel (e.g. *16* parallel sub-channels), with the data then being multiplexed on to these channels. The direct effects of ISI are avoided because the sub-channel pulse lengths are long (usually *10* to *30msec*) compared with typical channel dispersion times, such that eqn.1.1 holds, and :

$$w_B << (\Delta f)_c \quad \text{where} \qquad (\Delta f)_c \approx 1/T_M \qquad 1.2$$

where w_B is the bandwidth of the transmitted signal, i.e. ($w_B \approx 1/T_B$ in most voice band applications), and $(\Delta f)_c$ denotes the coherence bandwidth of the channel, which is a measure of the frequency difference required between two sinusoids for them to be affected differently by the channel [1]. Thus each sub-channel is frequency-non-selective, and the receiver effectively sees only one sky wave path in each sub-channel. Another insurance against significant ISI is to employ time-guard at the end of each baud interval, the duration of which should be $\approx T_M$. The best known example of an HF parallel sub-channel modem is KINEPLEX [114]. There are several practical variants of this type of modem, but they all essentially employ multiple sub-channels (typically *16*), each with quarternary differential phase-shift-keying (DQPSK) and orthogonal sub-channel frequency spacing. A typical modem accepts data at a rate of *2400 bits/s* and formats this into 16 parallel sub-channels, each operating at a rate of *150bit/s*. These *16* data sub-channels are then differentially phase-key modulated on to *16* corresponding base-band sub-carriers; quarternary modulation allows the transmission of *2bits/phase* and thus the baud rate for each sub-channel is *75*, i.e. a frame interval of *13.3msec*. Of this frame

interval, *4.2msec* is employed as a "guard time" to allow the effects of ISI to be rejected. The performance of such a system is influenced by the deep fades that occur during transmission. Therefore another parallel system was developed to overcome this problem by using error correcting code and diversity techniques. in which redundancy is introduced into the system in an attempt to lessen the chance of severe signal loss. This refers generally to methods which send the wanted transmit signal two (or more) times in some manner to provide the receiver with multiple replicas of the same information signal, each of which have arrived under independently fading channel conditions. Such diversity techniques are :

. Multiple antenna method (space diversity) referred to as explicit diversity, which is an expensive method to use.

. Implicit diversity such as frequency and time diversity : when the channel itself provides the redundancy. For example if the channel is frequency-selective, i.e. $(B_s >\approx (\Delta f)_c)$ then, there is potential for implicit frequency diversity gain, because different portions of the frequency band fade independently of one another. B_s is the available bandwidth (in Hz), which is assumed to be an integer multiple (≥ 1) of the bandwidth w_B. Another kind of diversity is implicit time diversity, which is related to the coherence time $(\Delta T)_c$. $(\Delta T)_c$ is a measure of the minimum difference in time required between the transmission of two sinusoids, of the same frequency, for their respective attenuations suffered to be uncorrelated [1]. Assuming $B_d \approx 1/(\Delta T)_c$, where B_d is the Doppler spread of the channel which is a measure of the width of the received spectrum when a single sinusoid is transmitted through the channel. It is also a measure of the rate of change with time of the envelope of the sinusoid [1,6]. It can equally well be interpreted as the rate of fading of the channel with time. For a slow-varying channel, B_d is small, or equivalently, $(\Delta T)_c$ is large [1,6]. In fast-fading channel environments, where $(1/T_B >\approx B_d)$ redundant symbols in a coding scheme can be used to provide time diversity if the code word spans more than one fade period. In the case of slow-fading environments, where $1/T_B >> B_d$, this condition of spanning the fade period can be achieved by interleaving the code words so as to introduce time gaps $>\approx (\Delta T)_c$ between successive symbols in a particular code word. However, apart from reducing the net information rate of the system, this method means that the signal delay will be greater than $(\Delta T)_c$. In some applications, the required time delay is unsatisfactorily long for two-way communication [6]. Many systems have been developed using coding and diversity techniques such as CODEM, ANDVT and KATHRYN [109,110]. Unfortunately, the parallel sub-channel system suffers from the following [6,31,107-110,114,116] :

(i) Inefficient use of the available transmitted power : The amplitude of the composite signal varies according to the relative phases of the signal in the sub-channels. Because of the random nature of the process the composite signal will contain infrequent peaks of

magnitude several times the mean signal level. Since transmitter amplifiers are peak-power limited, the average transmitter power is likely to be will below that available.

(ii) It suffers from the effects of frequency-selective fading which can exceed the dynamic range of the individual sub-channels [1,6,116].

(iii) Maltipath effects cause signal dispersion which leads to some ISI, and it is necessary in the detection process to allow time-guard bands between bauds (sub-channels); these must be of a duration equal to the maximum expected dispersion time. This reduces the received signal energy available at the detector, and also reduces the number of available sub-carriers because of the need to preserve orthogonal spacing of the sub-carrier tones if cochannel interference is to be avoided in the detection process.

(iv) The performance of (uncoded) multitone transmitter schemes can be seriously degraded when spectral nulls in the channel fall on or near one of the tone frequencies used.

On account of the various difficulties referred to above an alternative to the parallel (multitone) approach and one which is considered in this thesis in great detail is to use a serial transmission system (single tone) and to employ some form of adaptive signal processing at the receiver. Comparisons of the two approaches at a speed of *2400bit/sec* have suggested that the single tone modem offers a better overall performance [31,111,114]. With the increase in the processing speed of digital hardware combined with an efficient signal processing techniques, serial modems are challenging the dominance of the parallel modem for high speed applications [23,26,30,31,48,107-112].

In high speed serial data transmission over the whole of the voice-band ($\approx 3KHz$), the symbol interval is short compared to HF multipath spread, such that :

$$1/T_B \approx w_B = B_s >\approx (\Delta f)_c \Rightarrow T_B <\approx T_M \qquad 1.3$$

One of the major problems facing the modem designer is that of eliminating the ISI. Receivers used to combat ISI can be classed into two separate groups [1,23,29,40]. In the first group, the receiver employs a device known as an equaliser to remove ISI from the received signal before passing it on to a detector, and the detection process is now a simple threshold comparator. The detector makes a decision on the value of a transmitted data symbol, by comparing the corresponding sample value with the appropriate threshold level (or levels). Equalisation techniques used are the linear equaliser and non-linear (decision feedback) equaliser [1,26,36,40,52,57,..]. The primary problem encountered in HF equalisation is the need to track a sometimes rapidly varying channel. This problem is accommodated by making the equaliser adaptive, that is, by using various adaptation algorithms to adjust the tap-gains of the equaliser at the appropriate time instants. Thus the tap-gains of a linear or decision feedback equaliser can be adjusted adaptively for a time-varying channel, using a gradient (LMS) algorithm [1,29,63,91,99] or Kalman algorithm (RLS) [1,23,48,69,81] or lattice algorithm [1,63,86] to minimise the mean square error in the equalised signal at its output. Alternatively an equaliser can be adjusted

from an estimate of the sampled impulse response (SIR) of the channel [1,23,27,28,61,62,67,73,92]. These methods require a special training sequence which is useful in maintaining performance, particularly when the tracking algorithm fails as happens in a deep fade or when the channel changes abruptly. Some of the proposed serial HF modems use equalisation techniques in the detection process [107,109-112,114].

The second group of detection techniques overcome the problem of ISI by modifying the detection process itself, to take account of the signal distortion caused by the channel. These detectors use a maximum likelihood sequence estimation (MLSE) algorithm and are known as the maximum likelihood detectors [1,29,37,38,42,65], and are the optimum detection processes for a sequence of data symbols transmitted over a non-ideal band-limited channel which introduces ISI and additive white Gaussian noise (AWGN), and where the transmitted symbols are equally likely to have any one of their possible values [29]. Unlike the equaliser, the MLSE makes no attempt to remove ISI so that the whole of the received signal energy is used in the detection of the transmitted data symbols, and it is optimum in the sense that, under the appropriate conditions, it minimises the probability of error in the detection of the whole message. The MLSE can be efficiently implemented via a recursive algorithm known as the Viterbi algorithm [1,23,38]. Unfortunately, when m-level data symbols are transmitted and at the same time the sampled impulse response of the channel has more than a few non-zero components, and the Viterbi detector becomes complex, since it requires an excessive amount of storage and computation. In the detection of a received message, it involves (m^{L+1}) operations per signal element ($L+1$ is the length of the channel SIR, which lies in the range *15* to *20* for HF channel).

A solution to this problem is to limit the number of vectors held in the receiver store at any time instant to a small value, regardless of the number of components in the SIR of the channel ($L+1$), but without reducing unduly the tolerance of the detector to noise. This has led to a relatively simple detection process and is known as near-maximum likelihood (NML) detection [26,36,42-46,58], in which a near optimum performance can be obtained with a far smaller number of operations per received signal element. It has been shown in [43,45,58] that NML detectors are not significantly inferior to the Viterbi detectors in terms of their tolerance to additive white Gaussian noise, especially when a binary or quarternary signal is transmitted. Various NML detectors have been developed from the Viterbi algorithm by reducing drastically the number of stored vectors, but without otherwise changing the basic method of operation[36,43-46,58,117]. Various NML detectors are tested in this thesis, operating over an HF channel model at *4800 bit/sec* employing 4-level quadrature amplitude modulation (QPSK).

A NML detector requires knowledge (as indeed does a Viterbi detector), not only of the possible transmitted data symbol values, but also of the SIR of the channel. Therefore the receiver must estimate continuously the SIR of the channel and appropriately update

the stored estimate that is used by the NML detector at every appropriate sampling instant [23,30,61,117], since the HF radio links vary considerably with time. The knowledge of the SIR of the channel can be obtained by using a channel estimator [27,61,67,117]. When the channel varies with time the tap coefficients of the estimator (SIR) must be adjusted adaptively at every sampling instant, according to a particular algorithm.

A reasonably satisfactory estimate of the channel SIR can be obtained by means of the conventional gradient algorithm [61,62,65,67,117], which is simple to implement and works adequately in a variety of applications, but suffers from the disadvantage of having a slow convergence. This algorithm is in fact, a recursive solution to the least-mean square estimation problems and also is termed the least-mean square (LMS) error algorithm [1,61-63,65-66,75,99,]. However the gradient algorithm is suitable for applications where the channel is time-invariant or it varies only very slowly with time. Furthermore, by suitably modifying the LMS algorithm itself a much more accurate estimate of very rapidly time-varying channels, such as HF channel, can be achieved by using an appropriate prediction technique [60,62,64,117].

Another approach is to use a Kalman (or recursive least-square : RLS) algorithm as an estimator [23,27,76,78]. It is well known that, when appropriately designed, a Kalman estimator gives the least-square weighted error in the SIR of a time-invariant channel [1,23,63,66,74,97]. This algorithm offers improved convergence but it is considerably more complex than the corresponding gradient algorithm. It has been found that the Kalman algorithm is sensitive to computer round-off errors, and a reduction in numerical accuracy due to round-off errors over time degrades the performance of the system [27,48]. In order to reduce the complexity of Kalman algorithm new fast RLS algorithms have been developed [23,28,63,84,98], but these exhibit numerical instability (worse than the conventional Kalman algorithm) in time-varying channel environments [48]. Therefore a square-root Kalman algorithm (SRKA) has been developed by F.M. Hsu [48] to improve the performance of the Kalman algorithm in terms of stability and numerical accuracy, when used to update the tap-gains of a DFE for HF channel applications. Different types of channel estimator based on the above strategies are examined in this thesis when operating over time-varying HF channels, and a novel estimator for use in the combined receiver (detector, estimator and adaptive equaliser) for operating over an HF channel at *4800bit/sec* is developed.

Another requirement in a NML detector, to achieve near-optimum performance, is that the magnitude of the earliest components of the SIR of the channel should be large, relative to the other components, which is not the case in the HF channel [23,39]. If the earliest components are the largest, then the SIR is referred to as near-minimum (or minimum) phase [40]. Therefore the SIR of the HF radio link is not minimum phase. Some sort of processing technique is needed to convert the SIR of the HF channel from

non-minimum phased to at least approximately minimum phased, and such a process is referred to as a minimum phase algorithm (or root-finding algorithm).

The SIR of the channel may be made minimum (or near-minimum) phase using a prefilter located at the input to the NML detector [40]. An adaptive linear feedforward transversal filter has been developed for the telephone channel [39]. The filter is placed just ahead of the NML detector to remove all or nearly all the phase distortion introduced by the channel. It should also adjust the SIR of the channel and filter, using a minimum phase (or root-finding) algorithm, into a form ideally suited to an NML detector of the type considered in this work. This algorithm requires only at it's input an estimate of the SIR of the channel which, of course, is provided by the channel estimator. This thesis examines the operation of this type of algorithm after some modification to operate over an HF channel model.

1.2 OUTLINE OF THE INVESTIGATION

The primary objective of the research in this thesis is to investigate and to develop the serial transmission of a *4800 bit/sec* QPSK system operating at *2400 baud* with an *1800Hz* carrier over a voice-band HF radio link. The receiver of such a system consists mainly of three parts which are the NML detector, channel estimator and adaptive equaliser. Thus the main concern has been the study, test and development of various types of these three parts, in order to select the most suitable detector, estimator and adaptive equaliser for an HF channel modem in terms of complexity and performance. This has led to a novel combined detector-estimator. The research has been carried out using computer simulation of the different arrangements of the system to test their performance over a model of an HF radio link.

Chapter 2 provides a detailed description of the HF radio channel. It contains a brief description of the ionospheric structure and the propagation mechanism in the ionosphere followed by the description of the possible signal impairments that could occur when a data signal is transmitted over such a channel. Finally a model of the HF channel is presented in a form suitable for computer simulation.

Chapter 3 describes a model of synchronous serial QAM digital data transmission, and gives an equivalent baseband software model of a three sky wave data transmission system. Finally, the generation of the SIR of an HF channel model is presented, which is used for testing the different parts contained in this thesis, by means of computer simulation.

Chapter 4 is a review of the linear and non-linear (decision feedback) equalisers, and describes the adaptive adjustment method of the receiver employing a feedforward transversal filter which uses a minimum phase (root-finding) algorithm (MPA) to make the SIR of the channel at least approximately minimum phased. The MPA is tested over

the HF channel model derived in chapter 3 to investigate the algorithm capabilities to convert the non-minimum phased channel to a near-minimum phased channel.

In chapter 5 a brief description of the Viterbi algorithm is given followed by the description of the operation of various types of the NML detectors, derived from the Viterbi algorithm. The performance of these detectors, when operating over an HF channel model are presented assuming perfect estimation, and the channel is made a minimum (or near-minimum) phase using the algorithm described in Chapter 4. Finally the performance of a non-linear equaliser, that uses a threshold detector, is also presented in this chapter for comparison purposes with the NML detector, assuming the same conditions. Most of the results obtained in this chapter do not assume the use of a channel estimator or pre-detection filter. Therefore these results give an upper-bound to the results obtained when a channel estimator is used to provide an estimate of the SIR of the channel.

Chapter 6 is concerned with channel estimation which is a necessary function of the NML receiver. It describes four types of channel estimators which are based on the LMS algorithm and RLS algorithm. A description of the simple gradient (LMS with fixed step-size) estimator is given first, where an appropriate prediction technique is incorporated with this estimator. A novel estimator, known here as an adaptive step-size LMS estimator (estimator 3), is described. This is a modified version of a gradient estimator making the step-size of the algorithm adjusted adaptively at every instant of time. A description is next presented of a modified form of RLS tracking algorithm, known as square-root Kalman algorithm (SRKA) [48], when used as an HF channel estimator. The final estimator considered in this chapter is actually the previous estimator (SRKA) but now the prediction technique is used. Finally the performance of the estimators is compared under the assumption of perfect detection.

The final work in the research project is contained in chapter 7 of this thesis, and concerns the operation and performance of various arrangements of combined estimator, NML detector and adaptive equaliser when no assumption has made (perfect estimation or detection). A large number of tests are performed on these arrangements using computer simulation, in order to determine the best combined system for *4800bit/sec* HF radio digital data transmission in terms of complexity and performance. This has led to a novel *4.8kbit/sec* combined receiver, which gives good results but is not unduly complex. The practical implementation of such system can be achieved using the new technology of high-speed digital signal processing (DSP).

Chapter 8 contains a summary of the project, the main conclusions drawn from the research, and suggestion for further work. Finally the thesis contains several appendices which are needed for the explanation of various algorithms. The appendices also include the *FORTRAN 77* code listings of the main computer programs that were developed by the author, during the course of the research.

CHAPTER 2

THE MULTIPATH HF RADIO FADING CHANNEL

2.1 INTRODUCTION

The HF channel has an important role to play in long distance communication, even after the introduction of several other kinds of transmission media such as coaxial cables, satellites, optical fibers, etc. Such latter channels cause little practical difficulty for medium-speed digital data transmission, but the problems of transmitting data reliably at this rate via a high frequency or HF ionospheric sky-wave paths are significant. The practical importance of the HF path is that it provides the only alternative to satellite for long-distance communication.

The frequency-band between 3 and 30MHz is the high frequency radio spectrum. A voice-band in HF channel normally occupies 3.4KHz within this spectrum. When a telephone channel is used as a transmission path, the properties of the transmission route can be quite accurately defined and reproduced independently of time factors. Such a channel is called a time-invariant channel, but for HF radio link this is not the case. For sky-wave propagation, the transmission conditions are constantly changing and this channel is called a time-varying channel. The radio waves propagate primarily due to reflection from the ionosphere, therefore to model an HF channel, it is necessary to understand the ionosphere structure.

This chapter starts with a brief description, in section 2.2 of the structure of the ionosphere. Section 2.3 and 2.4 explain the propagation of radio waves through the ionosphere. Section 2.5 describes the most common and significant signal distortions which occur on HF links. These are discussed in some detail since an HF channel model should be capable of simulating such impairments. In section 2.6 an explanation of the characteristics of fading multipath channels is given. In section 2.7 and 2.8 a description of the channel model of the system, and a classification of the HF radio channels used in the thesis, are given. The results of the computer simulation are presented in section 2.9.

2.2 THE IONOSPHERE STRUCTURE

The region between 50-500 Km above the earth's surface is called the ionosphere, and consists of molecules and atoms of Nitrogen and Oxygen. These are ionised principally, by the electromagnetic radiation from the sun into free electrons sufficiently to influence the propagation of radio waves [1-4]. The degree of ionisation is primarily a

function of the sun's ultraviolet (UV) radiation and may also be caused by particle radiation from sunspots, cosmic rays, and meteor activity [2,3]. Depending on the intensity of the UV radiation, more than one ionised layer may form. The existence of more than one ionised layer in the atmosphere is explained by the existence of different UV frequencies [2], in the sun's radiation. The lower frequencies produce the upper ionospheric layers, expending all their energy at high altitude. The higher frequency UV waves penetrate the atmosphere more deeply before producing appreciable ionisation [2]. Thus the level of ionisation would depend not only on the intensity of the solar radiation but also on the frequency of collisions between electrons and other particles. This level of ionisation is not uniform throughout the region from 50 to 450Km, the ionisation being distributed in layers having peak intensities at particular heights [4].

As a result, the peak electron densities in the upper layers are greater than those in the lower layers. Beyond an altitude of about 450Km, however, the electron density actually decreases with height, aided, no doubt, by a relative increase in the collision frequency and other complex phenomena [2,4,7]. For all practical purposes four layers of the ionosphere have been identified namely, D, E, F1 and F2. The salient parameters of these layers are their virtual heights, electron densities and critical frequencies, where the latter is defined as the highest frequency of a vertically incident ray that can be reflected by the layer. Due to the volatile nature of the ionosphere, the value of these parameters change both temporally and spatially, and as such, any values given should be taken as typical and not absolute [4] (see Table 2.2.1). The temporal variations occur daily, seasonally and according to the geomagnetic latitude. Fig 2.2.1 shows a profile of the three regions and depicts the electron density, and Fig.2.2.2 shows the ionised layers of the atmosphere as a function of nominal height above the earth's surface.

D-Region or D-Layer : not always present, but when it does exist, it is a day-time phenomenon, and it is the lowest of the four layers. When it exists, it occupies an area between 50 and 90Km above the earth.

The D-region, because of its relatively higher concentration of neutral particles and heavy ions, extracts energy from a passing wave as a result of collisions with electrons excited by the wave. As far as HF propagation is concerned, this region is viewed as an attenuation band. The level of ionisation is approximately 10^2 electrons/cc at 70Km, and 10^4 electrons/cc at 90Km (see Fig.2.2.1). For various reasons, including its high absorption and low electron density, the D-region has not been fully examined because of instrumentation difficulties. However it is known that the D-layer electron density varies with the 11-year solar sun-spot cycle [4], and with the sun's zenith distance. The electron density in this region is at a maximum at noon and during the summer.

The critical frequency for the D-region, defined as the highest carrier frequency of a vertically incident ray which can be reflected by the layer is of the order of 100 to 700 *KHz*.

E Region: A day-light phenomenon, existing between 90 and 140*Km* above the earth with a maximum region ionisation at about 110*Km*. The electron density at this height is in the order of 10^5 electrons/cc during day-light hours. At night-time there is still some ionisation, but it is much weaker. The E-layer depends directly on the sun's UV radiation and hence it is most dense directly under the sun. Layer density varies with seasons owing to variations in the sun's zenith angle with seasons. The E-layer has a critical frequency of about 4*MHz* [4]. In addition to the normal E-layer ionisation, there appears occasionally, patches of denser ionisation at E-layer heights, that seem to travel as ionisation clouds. This unpredictable phenomenon is called Sporadic E (Es). Their properties and frequency of occurrence differ significantly with geomagnetic latitude. An Es-layer is capable of reflecting very high frequencies (>10*MHz*), but is generally regarded as a nuisance for HF radio wave propagation due to its 'sporadic' occurrence and highly volatile and unreliable nature. The E region is useful for propagation support for distances up to 2000*Km*, using frequencies as high as 20*MHz*.

The region above 160*Km* is known as the F region which is the most important region for HF radio communication. Due to the different variations observed in the lower and upper parts of this region, it has been further subdivided into two layers (F1, F2). However, at night, these two layers merge to form a single F-layer which can support propagation up to distances of 4000*Km*, and is the only part of the ionosphere which could be used for reliable night-time communication. Frequencies as high as 50*MHz* (when the ionisation level is high) can be utilised for this mode.

F1 layer : A daylight phenomenon existing between 140 and 250*Km* above the earth. Its behaviour is similar to that of the E-layer in that it tends to follow the sun, i.e. most dense under the sun. At sunset the F1 layer rises, merging with the next higher layer (F2-layer).

F2-layer : This layer exists day and night between 150 and 250*Km* (night) and 250-300*Km* above the earth (day). During the day time in winter, it extends from 250 to 300*Km* above the earth. Variations in height are due to solar heat. The electron density in this region is in the order of 10^6 electrons/cc. The critical frequency for the F2 layer is between 5 to 10*MHz* [2,4]. The F2 layer is an important part of the ionosphere for HF radio communication both during day and night time. Since the F2 layer is at a considerable height, it can support single hop propagation over a long distance.

2.3 THE PROPAGATION MECHANISM IN THE IONOSPHERE

The ionospheric mechanism provides propagation support in the range 3-30MHz by means of specular reflection, refraction, or by scatter within the ionised medium. The ability of the ionosphere to provide propagation support is related simply to the condition that its refractive index at radio frequencies is different from that of free space.

HF radio waves propagated through the ionosphere and returned to earth through a phenomenon known as refractive bending. The refractive index of the ionospheric layer changes continuously with its height as it is a function of the electron density in the ionised medium [2,4]. Fig.2.3.1 and Fig.2.3.2 show the refractive bending of a radio wave. The total internal refraction for a given angle of incidence of a radio wave is given by [2,4] :

$$\eta = \sin\theta_i = \sqrt{1 - \frac{81 \cdot N}{F^2}} \qquad (2.3.1)$$

where : η is the refractive index .

$F = w/2\pi$ frequency of the radio wave in hertz (Hz).

N is electron density in electrons per cubic meter (m^3).

θ_i the angle of incidence of the electromagnetic wave with the ionised layer.

An important application of the above relations is in their use in obtaining what is known as the critical frequency for the case where the electromagnetic wave is vertically incident, i.e. ($\theta_i = 0, \sin\theta_i = 0,$ and $\eta = 0$). Such a wave will reach a height determined by N and then be returned to earth.

From eqn. 2.3.1 at $\theta_i = 0 \Rightarrow 1 - \frac{81}{F_c^2} \cdot N_{\max} = 0 \Rightarrow 1 = \frac{81}{F_c^2} \cdot N_{\max}$

where F_c is the critical frequency in Hz, then :

$$F_c = 9\sqrt{N_{\max}} \qquad (2.3.2)$$

F_c is the highest frequency which can be reflected as a result of vertical incidence. It is obviously only dependent upon $N_{\max}$, the maximum electron density in electrons per cc. For a given angle of incidence, θ_1, the maximum frequency at which reflection happen is called the maximum usable frequency (MUF) and is given as a function of F_c by [2,4] :

$$MUF = F_c \cdot \sec\theta_1 \qquad (2.3.3)$$

where: ($\sec\theta_1 = \frac{1}{\cos\theta_1}$), from eqn.2.3.1 and 2.3.2

$$\Rightarrow \sin^2\theta_1 = 1 - \frac{F_c^2}{F^2} \Rightarrow \frac{F_c^2}{F^2} = 1 - \sin^2\theta_1 = \cos^2\theta_1 \Rightarrow F = F_c \cdot \cos^{-1}\theta_1$$

Due to relationship between F_c and MUF, it is clear that for a given angle of incidence, the MUF'S of higher layers are greater than those of the lower layers. From Fig.2.3.2, at the (MUF), the radio wave takes the critical path which is the shortest distant back to earth, this is called the skip distance [2]. The refraction processes through a flat ionised region at some height "b" above the earth's surface is equivalent to a mirror like reflection from a reflector located at a height "a" above the earth (see Fig.2.3.3) [2], where the height "a" is called the virtual height. Thus the actual ray path can be replaced by the virtual ray path in a medium of unit refraction index and reflected from a plane located at the virtual height.

The operating frequency of an HF radio link should be chosen such that it is close to the MUF of the layer via which propagation is desired. This frequency is called the optimum working frequency (FOT), which is taken to be approximately 85% of the MUF [2,4]. The usable frequency has also a lower bound, the lowest usable frequency (LUF). It is due to it having a minimum signal-to-noise-ratio at the receiver.

Note that the above discussion has been based on ray theory, assuming a flat earth and neglecting the effect of the earth's magnetic field. However, it's adequate for the purpose intended, which is to give the reader a broad indication of the manner in which HF radio waves propagate through the ionosphere .

The presence of the earth's magnetic field gives rise to a phenomenon known as ray splitting, or magneto-ionic splitting [14]. When a radio wave enters the ionosphere, it interacts with the earth's magnetic field and is split into two differently polarised waves called the ordinary and extra ordinary rays as shown in Fig.2.3.4.

These waves are reflected by a different electron density and as such, at slightly different hights from the ionosphere.

2.4 BASIC HF PROPAGATION

An HF wave emitted from an antenna is characterised by a ground-wave and a sky-wave component. The ground-wave follows the surface of the earth and can provide useful communication from $40Km$ to $160Km$ or more, depending on RF power, antenna type and height off the ground, atmospheric noise, man-made noise and ground conductivity. Well designed ground wave links, with their shorter range will achieve a better time availability than sky-wave links. Sky-wave links are used for long circuits from about $160Km$ to $12800Km$.

The sky wave transmission phenomenon of HF depends on ionospheric refraction. Transmitted radio waves hitting the ionosphere are bent or refracted. When they are bent sufficiently, the waves are returned to earth at a distant location. Often at the distant location they are reflected back to the sky again, only to be returned to earth again even further from the transmitter.

2.4.1 PROPAGATION MODES

There are three basic modes of HF propagation :

1. Ground wave.
2. Sky wave (Oblique incidence).
3. Near vertical incidence (NVI) This is a distinct subset of the sky wave mode (2).

1.Basic Ground-wave Propagation : The space wave (not sky-wave) intensity decreases with the inverse of the distance where, the ground-wave decreases with the inverse of distance squared. Therefore, at long distances and non-zero elevation angles the intensity of the space wave exceeds that of the ground wave. The ground wave is diffracted somewhat to follow the curvature of the earth. The diffraction increases as frequency decreases. Diffraction is also influenced by the imperfect conductivity of the ground. Energy is absorbed by currents induced in the earth so that energy flow takes place from the wave downward. The loss of energy dissipated in the earth leads to attenuation dependent on conductivity and dielectric constant. With horizontal polarisation the wave attenuation is greater than with vertical polarisation due to the different behaviour of Fresnel reflection coefficients for both polarisations [14].

To summarise, ground wave is an excellent form of HF propagation where we can during day-light hours, achieve 99% [14] or better time availability. Ground wave propagation decreases with increasing frequency and with decreasing ground conductivity. As we go down in frequency, atmospheric noise starts to limit performance in day time, and sky wave interference at night is a basic limiter of ground wave performance on whichever the lower frequencies we wish to use, providing we are not operating above the MUF.

2. Sky Wave Propagation : A wave that has been reflected from the ionosphere is commonly called a sky wave. The reflection can take place at the E region and the F1 and/or F2 regions. In some circumstances RF energy can be reflected back from any two or all three regions at once.

HF sky wave communication can be one-hop, two-hop or three-hop, depending on path length and ionospheric conditions. Fig.2.4.1 shows eight possible sky wave modes of propagation . On somewhat longer paths (= > 1000Km), we can receive RF energy from two or more modes at once, giving rise to multipath reception which causes signal dispersion. Dispersion results in inter-symbol interference (ISI) on digital circuits.

In general, multipath propagation can arise from :

- Multi-hop especially when transmit and receive antennas have low gain and low take-off angles.
- Low and high angle paths (the low ray and the high ray).
- multi-layer propagation.

. ordinary (O) and extra-ordinary (X) rays from one or more paths (ray splitting).

Typical one-hop ranges are 2000, 3400 and 4000*Km* for E-, F1- and F2- layer reflections, respectively [14]. These limits depend on the layer height of maximum electron density for rays launched at grazing incidence. Distances beyond the values given above can be achieved by utilising consecutive reflections between the ionosphere and the earth's surface (see Fig.2.4.1 b-h). For each ground reflection the signal must pass through the absorptive D-layer twice, adding significantly to signal attenuation. The ground reflection itself is absorptive. It should be noted that the elevation angle increases as a function of the hop number, which results in lowering the path MUF.

The sky wave propagation based on F-layer modes is identified by a three character notation such as 1F1, 1F2, 2F2, and 3F2. The first digit is the hop number and the second two characters identify the dominant mode, i.e. F1 or F2 reflection. Accordingly, 2F2 means two hops where the dominant mode is F2 reflection.

An HF receiving installation will commonly receive multiple modes simultaneously, typically 1F2 and 2F2 and at greater distances 2F2 and 3F2. The strongest mode on a long path is usually the lowest order F2 mode unless the antenna discriminates against this. It can be appreciated that higher order F2 modes suffer greater attenuation due to absorption by D-layer passage and ground reflection. The result is a lower level signal than the lower order F2 propagation modes. In other words, a 3F2 mode is considerably more attenuated at a certain location than a 2F2 mode if it can be received at the same location (assuming isotropic antennas), [14]. It has travelled through the D-layer two more times than its 2F2 counterpart and been absorbed one more time by ground reflection. E-layer propagation is rarely of importance beyond one hop, reflections from the F1 layer occur only under restricted condition, and the 1F1 mode is less common than the 1E and 1F2 modes. The 1F1 mode is more common at high latitudes at ranges of 2000-2500*Km*. Multiple-hop F1 modes are very rare [14].

Figure 2.4.1 illustrates single-mode paths (a and b) and mixed mode paths (d-f). Figure 2.4.1-c shows a path with asymmetry. This occurs when a wave frequency exceeds the E-layer MUF only slightly, so that the wave does not penetrate the layer along a rectilinear path but will be bent downward resulting in the asymmetry.

Ray Splitting : a ray (wave) entering the ionosphere is split into two separate waves owing to the influence of the earth's magnetic field [2,14].

One ray of the split rays is called the ordinary wave (O-wave) and the other the extraordinary wave (X-wave). the O-wave will have a lower critical frequency, hence a lower MUF [2,14]. Both waves experience different amounts of refraction and thus travel independently along different ray paths displaced in time at the receiver, typically from 1 to 10 μsec. The O-wave suffers less absorption and therefore the more important (see Fig.2.3.4).

3. Near Vertical Incidence (NVI) Propagation : NVI propagation is used for short range HF communication. It can fill the so-called skip zone or zone of silence : here ground wave propagation is no longer effective, [2,14] to the point where one-hop sky wave, using oblique incidence propagation, may be used. NVI utilises the same sky wave principles of propagation discussed above. The key factor in NVI mode is the antenna . For effective HF communication using the NVI mode, the antenna must radiate its main beam energy at a very high angle, near vertical [2,14].

NVI circuits suffer the same impairments as oblique sky wave circuits, but in the case of NVI the fading is more severe [14]. Fig.2.4.2 shows diagrammatically the operation of NVI propagation. The letter *A* in the figure shows the extent of useful communication by means of ground wave component if it desired to transmit with a low-elevation angle antenna with vertical polarisation, such as a whip. In general, it can be used lower frequencies from 2 to 7*MHz* for NVI operation [14].

2.5 HF COMMUNICATION IMPAIRMENTS

2.5.1 Introduction

Signals transmitted over the HF channel arrive at the receiver suffering from severe time-varying amplitude distortion. This distortion introduces severe degradation into the quality of analogue communication and it has an even worse effect on digital data communication. The parameters that characterise an HF channel are based on these types of distortion, thus it is essential to understand them. There are a number of important impairments on an HF channel that affect received signal quality.

2.5.2 Fading

HF sky wave signals suffer from fading which is a random variation of the signal strength at the receiver. The amplitude and phase of sky wave signals fluctuate with reference to time, space and frequency. Fading may be caused by several different ionospheric phenomena. For example, movements of the ionosphere causing interference fading, rotation of the axes of the polarisation ellipses, time varying ionospheric absorption and skipping of the signal due to the maximum usable frequency MUF failure.

The period of a fading cycle depends largely on the cause of the fading. Thus the period of interference and polarisation fading may vary from a fraction of a second to a few seconds. In general, fading is faster on high frequencies than on low frequencies because a given movement in the ionosphere produces a greater phase shift on the shorter wave length [14.2]. We consider six types of fading :

a- Interference Fading : This is the most common type of fading encountered on HF circuits. It is caused by the mixing of two or more signal components propagating along

different paths. This is multipath fading which may arise from multiple-mode and multiple-layer propagated rays, high and low angle modes, ground and sky waves. Also a beam of radio waves incident on the ionosphere is not reflected from a point but from an extended region, small irregularities in electron density near the level of reflection give rise to individual reflected wave-lets and the received signal is the vector sum of the individual signals at the receiving antenna. Movements of ionospheric irregularities give rise to variation in the relative phase of the individual wave-lets and thus produce interference fading [2, 4, 14].

b-Polarising Fading : Whereas interference fading is regarded as characteristic of the incident wave, additional variation in the field intensity affecting the receiving antenna, called polarisation fading, occurs as a result of changes in the state of polarisation of the incident wave relative to the orientation of the receiving antenna. In general, the state of polarisation of the down-coming sky wave is constantly changed. This due mainly to the combination, with random amplitudes and phases, of the two positively polarised magneto-ionic components, the O-wave and X-wave. The state of polarisation of the down coming sky wave is in general elliptical, with either direction of rotation, and with random and constantly changing values of the dimension and orientation of the ellipse with respect to the receiving antenna [14].

c-Absorption Fading : Is caused by solar Flare activity. This type of fading particularly affects the lower frequencies, and fades may last from minutes to more than an hour [14]. Absorption fading is caused by short-time variations in the amount of energy lost from the wave because of absorption in the ionosphere [5]. The attenuation characteristic of the D-layer slowly changes and can last longer than an hour and is usually the greatest during sunrise and sunset [2].

d-Skip Fading : Skip fading is observed at places near the skip distance, and is caused by the waves alternately skipping and returning to earth near sunrise and sunset, when the ionisation density of the atmosphere is changing. It may happen that the MUF for a given transmission path oscillates about the virtual (optimum working) frequency [5].

e-Selective Fading : A modulated carrier has, within its bandwidth, a large number of frequency components that are exposed to randomly varying multipath propagation conditions. In other words, the fact that fading is frequency dependent means that different sidebands in a modulated wave fade differently. This gives rise to a distortion of the modulation envelope which is called selective fading [2,5].

f- Flat Fading : The term flat fading or (multiplicative) fading arises when all frequency components in the signal are affected in a similar manner.

Of interest to the communication engineer are fading depth, duration, and frequency (fade rate). For short-term fading, the fading depth is the difference in decibels (dB) between the signal levels exceeded for 10% and 90% of the time. Measurements have confirmed that we may expect about $14dB$ for Rayleigh distribution short-term fading, which is the most common form of such fading. This value is valid for paths (1500-6000)Km long and does not vary much with the time of day or season [14]. For long-term variations in signal level (i.e., variation of hourly median signal values of a month), the long-normal distribution provides a best fit [14].

2.5.3 Time Dispersion

On sky wave paths the primary cause of time dispersion is multipath propagation, which derives from differences in transit time between different propagation paths. The multipath spread causes amplitude and phase variations in signal spectrum owing to interference of the multipath wave components. When these fluctuations are correlated within the signal bandwidth and all the spectral components behave more or less in the same manner, we then call this flat-fading. When these fluctuations have little correlation, the fading is called frequency selective fading. Time dispersion is characterised by a delay power spectrum and is measured as multipath delay spread in micro-seconds or milli-seconds. Time dispersion is an especially serious and destructive impairment to digital communication signals in HF. One rule of thumb that is useful, is that if the delay spread exceeds half the time width (period) of a signal element, the error rate becomes intolerable. This is one rationale for extending the width of a signal element (e.g., by lowering the baud rate), to combat time dispersion. For instance, if a serial bit stream is transmitted at *100* bauds, a baud period is *0.01sec* (10*msec*). In this case, the circuit will remain operational although with a degraded BER, if the time dispersion remains under 5*msec*, i.e. half the period of a signal element or bit. At 200 baud the half-baud period value drops to 2.5*msec* and at *50* baud it is 10*msec*. At *2400* baud it is 0.2*msec*.
Multipath has been shown to be a function of the operating frequency relative to the MUF. Multipath delay tends to approach zero as the operating frequency approaches the MUF value [14,6].

Different modes of propagation have different group delays and this difference in the group delay also results in time dispersion. Time dispersion gives rise to inter-symbol interference, when the data transmission rate becomes comparable to the relative multipath delay. It is, thus a function of frequency, path length, local time, season and also geographical location [2,4,14].

2.5.4 Frequency Dispersion

Frequency dispersion arises on a single propagation path due to the Doppler effect introduced by the change in the altitude of the ionospheric layers, and nearly always is

present when there is time dispersion. But the converse does not necessarily hold true [14]. The upper regions of the atmosphere at high altitude are ionised first when the sun rises. The height of this ionisation level reduces as the sun rises further and further, until noon when it reaches a maximum. Thus the path traversed by a particular radio wave keeps decreasing and the emitted frequency appears to have increased. Exactly the opposite phenomenon takes place when the sun sets whereby, now the radio wave takes a longer path to reach the receiver and the emitted frequency now appears to have decreased. During night-time when the ionosphere is calm there is no Doppler effect [2,3]. In general frequency dispersion (Doppler shifts) range from 0.1-1.0Hz, and become quite small during night-time.

2.5.5 Noise in HF System

In declining importance, we categorise this noise as follows:

1. Interference from other emitters.
2. Atmospheric noise.
3. Man-made and galactic noise.
4. Receiver thermal noise.

1. The HF band has tens of thousands of users who are assigned operating frequencies by national authorities. We must also take into consideration noise from ionospheric sounder transmitters, harmonics and spurious emission from licensed emitters. To overcome this problem :

- Find a clear frequency (with no interference at the distant-end receiver) with sub-optimum propagation.
- Increase transmitted power to achieve the desired signal-to-noise ratio at the distant end.
- Use directional antennas where the interferer is in a side lobe and hence attenuated compared to the desired signal.
- Use antenna nulling. This is a form of electronic beam steering that creates a null in the direction of the interferer.
- Use sharp front-end preselection on receiver(s).

2. Lightning discharges cause the atmospheric noise which occupies a frequency band from very low frequencies to 30MHz at the input of the radio receiver. These electrical disturbances are transmitted long distances via the ionosphere in the same manner as HF sky waves. Atmospheric noise level is related directly to weather conditions.
The amplitude of lightning disturbances varies approximately inversely with the frequency squared [14], and is propagated in all directions both for ground and sky waves.

3. Man-Made noise can be generated by many sources, such as electrical machinery, automobile ignitions, all types of electronic processors/computers, high power electric transmission lines, and certain types of lighting. Impulsive noise in HF radio links is usually man-made interference and therefore only becomes really important in built up areas or where the radio receiving equipment is close to a source of electrical interference.

4. Only under very special circumstances does receiver thermal noise become a consideration in normal HF operation. If we consider that atmospheric, man-made and cosmic noise have a non-uniform distribution around an antenna, (i.e. it tends to be directional), then the antenna gain at a certain frequency will generally favour signal and discriminate against noise. As the result a band-limited received signal would thus contain noise components originating from several sources, each of which could be a statistically independent source. Central Limit Theorem arguments lead to the assumption that this noise has a Gaussian probability density function, at least near the mean, which is zero. This assumption has held the HF radio modem designer in good stead, since it has been found out that a modem having a better tolerance to additive Gaussian noise would almost certainly have a better tolerance to atmospheric noise [15]. Thus, performance comparisons of modems are carried out on the basis of their tolerance to additive Gaussian noise in this work.

Finally HF radio links also introduce time modulation effects which are known as Doppler shifts [3,15]. Random variation in the effective height of the ionosphere produce the corresponding random variation in the rate of arrival of the received signal wave form, which is therefore modulated in time. Since the transmitted HF radio signal has a band width of $3KHz$ and a carrier frequency in the range (3-$30MHz$), it is a narrow-band signal. It can be shown that the Doppler shifts are now approximately equivalent to frequency modulation effects. The combination of these with small frequency offset introduced by the radio equipment, results in a frequency offset of the receiver voice-frequency signal, which varies slowly with time and has a value in the range of $\pm 30Hz$ [3,15].

2.6 CHARACTERISATION OF FADING MULTIPATH CHANNELS

When an impulse is transmitted over a time-varying multipath channel the received signal might appear as a train of pulses. Hence one characteristic of a multipath medium is the time spread introduced in the signal which is transmitted through the channel [1,3]. A second characteristic is due to the time variation in the structure of the medium. As a result of such time variations appear to be unpredictable to the user of the channel, therefore it is reasonable to characterise the time-variant multipath channel statistically [1]. The usual line-of-sight propagation channel is characterised simply as a non-

distorting two-port filter described by a gain A (or attenuation) and by a propagation time delay. This is summarised as a delta-function impulse response of the form :

$$h(\tau) = A.\delta(\tau - \tau_0) \qquad 2.6.1$$

with a transfer function :

$$H(f) = A \cdot \exp(-j2\pi f \tau_0) \qquad 2.6.2$$

This represents an all-pass filter over an infinite domain of frequencies, obviously an over- idealisation of any physical channel, but one that is a perfectly adequate and correct description over the more limited band occupied by a communication signal.

A multipath channel is simply one where energy arrives via several such paths, usually as a result of reflections, or of inhomogeneities in the physical medium that produce ray-splitting or scattering effects. Continuous physical changes in the channel (motions, internal turbulence) cause small changes in the individual path lengths.

The variations between constructive and destructive interference resulting from the random phase changes comprise the effect called multipath fading, defined strictly speaking for unmodulated transmission (pure single frequency sine wave).

The transmitted signal, in general is given by [1] :

$$S(t) = \mathrm{Re}\{u(t).\exp(j2\pi f_c t)\} \qquad 2.6.3$$

where $Re\{.\}$ is the real part of the complex-valued quantity in brackets, f_c is the carrier frequency, and $u(t)$ the complex envelope of transmitted signal.

The observed received phasor is a vector sum of several phasors with the phase of each varying individually and randomly over a $(0,2\pi)$ range. When there are a several paths, the central limit theorem arguments lead to the conclusion that the received wave-form has all the characteristics of a very narrow band of stationary Gaussian noise [1], i.e. that it consists of Gaussian quadrature components characterised by a power spectral density of non-zero width, and with a corresponding Rayleigh distribution of the received envelope.

The received bandpass signal may be expressed in the form [1] :

$$x(t) = \sum_n \alpha_n(t).s[t - \tau_n(t)] \qquad 2.6.4$$

where $\alpha_n(t)$ is the attenuation factor for the signal received on the n^{th} path and $\tau_n(t)$ is the propagation delay for the n^{th} path. Substitution for $s(t)$ from eqn.2.6.3 into eqn.2.6.4 yields the result :

$$x(t) = \mathrm{Re}(\{\sum_n \alpha_n(t).\exp(-j2\pi f_c \tau_n(t)).u(t-\tau_n(t))\}.e^{j2\pi f_c t}) \quad 2.6.5$$

The equivalent low pass received signal can be derived from eqn.2.6.5

$$r(t) = \sum_n \alpha_n(t).\exp(-j2\pi.f_c.\tau_n(t)).u[t-\tau_n(t)] \quad 2.6.6$$

since *r(t)* is the response of an equivalent low pass channel to the equivalent low pass signal *u(t)*, it follows that the equivalent low pass channel is described by the time-variant impulse response :

$$c(\tau, t) = \sum_n \alpha_n(t).\exp(-j2\pi.f_c.\tau_n(t)).\delta(\tau - \tau_n(t)) \quad 2.6.7$$

When an unmodulated carrier, at frequency f_C, is transmitted, then $u(t)=1$ for all t, hence the received signal for the case of discrete multipath given by eqn.2.6.6 reduces to

$$\begin{aligned} r(t) &= \sum_n \alpha_n(t).\exp(-j2\pi.f_c.\tau_n(t)) \\ &= \sum_n \alpha_n(t).\exp(-j\phi_n(t)) \end{aligned} \quad 2.6.8$$

where $\phi_n(t) = 2\pi.f_c.\tau_n(t)$. Thus, the received signal is the sum of a number of time-variant vectors having amplitudes $\alpha_n(t)$ and phases $\phi_n(t)$. Large dynamic changes in the medium are required for $\alpha_n(t)$ to change sufficiently to cause a significant change in the received signal. On the other hand, $\phi_n(t)$ will change by 2π radians whenever τ_n changes by $(1/f_c)$. But $(1/f_c)$ is a small number, and hence, ϕ_n can change by 2π. with relatively small motion of the medium. Also the delay $\tau_n(t)$ associated with different signals changes at different rates and in an unpredictable (random) manner. This implies that the received signal *r(t)* in eqn.2.6.8 can be modelled as a random process [1]. When there is a large number of paths, then *r(t)* can be modelled as a complex valued Gaussian random process [1].

The multipath propagation model for the channel, in eqn. 2.6.8 leads to signal fading. The fading phenomenon is primarily a result of the time-variant phase $\{\phi_n(t)\}$ [1,3]. The statistical model for the short-term fading assumes a stationary statistic. That is, it assumes that the long-term variability is sufficiently slow that the channel statistics can be regarded to be sensibly fixed over some interval of engineering interest (minutes or longer) [3,14,15]. The amplitude variations in the received signal, termed signal fading are due to the time-variant multipath characteristics of the channel.

When the impulse response $c(\tau ; t)$ is modelled as a zero mean complex valued Gaussian process [1,3], the envelope $|c(\tau ; t)|$ at any instant *t* is Rayleigh-distributed. In this case the

channel is a Rayleigh fading channel. It is a continuous random variable, derived from two independently Gaussian random variables x, y and the first-order probability density function (PDF) of a stationary complex zero mean Gaussian process (complex envelope)

$z = x + j\,y$ is :

$$P(x,y) = \frac{1}{2\pi.\ s} \cdot \exp(-\frac{x^2 + y^2}{2.s}) \qquad 2.6.9$$

where the expected value $s = E[0.5|z|^2]$ 2.6.10

is the mean power in the random wave form. Its envelope and phase are :

$$R = \sqrt{x^2 + y^2} \quad \text{and} \quad \theta = \tan^{-1}\frac{y}{x} \qquad 2.6.11$$

whose joint first-order pdf is immediately observed to decompose into independent pdf's [1,3]:

uniform distribution $P(\theta) = \frac{1}{2\pi} \qquad 0 \le \theta < 2\pi$

and Rayleigh distribution $P(R) = \frac{R}{S} \cdot \exp(-\frac{R^2}{2S}) \qquad 0 \le R < \infty$ 2.6.12

where the mean value of x and y are zero and their variance is
$S = \sigma^2$ such that $\sigma_x^2 = \sigma_y^2 = \sigma^2$.

Since R can not be negative, then it must have a non-zero mean value, even though x and y have zero mean.

Associated with the distribution of the envelope is the PDF for its mean-squared value or "instantaneous power " as defined by $u=R^2/2$

$\Rightarrow P(u) = \frac{1}{S} \cdot \exp(-u/S)$ for $0 \le u < \infty$ (exponential distribution). The earlier statement about a strong approach to the central limit corresponds to a statement that a Rayleigh law for the envelope is likely to be most correct for the low values of R, near R=0. As it happens, this also characterises the domain of most concern for signal and receiver design, describing the signal envelope when it fades deeply below its mean or median. For this region, a valid approximation to the Rayleigh pdf is :

$$P(R) \approx R/s \qquad 0 \le R << \sqrt{s}$$

Fig.2.6.1 shows the plot of probability density function as a function of r and at $r = \sigma$ we obtain maximum value of the curve which is :

$$P(r) = \frac{1}{\sigma} \cdot e^{-0.5} = 1/\sigma \cdot \sqrt{e}$$

The cumulative distribution function of the Rayleigh distribution, is given by

$$f(r) = \int_0^r \frac{u}{\sigma^2} \cdot \exp(-\frac{u^2}{2\sigma^2}) \cdot du$$
$$= 1 - e^{-r^2/2\sigma^2} \quad \text{for} \quad r \geq 0 \qquad 2.6.13$$

Fig.2.6.2 shows the plot of the cumulative function and the mean value of R is given by

$$\bar{r} = \int_0^\infty r \cdot p(r) \cdot dr \Rightarrow$$
$$\bar{r} = 1/\sigma^2 \cdot \int_0^\infty r^2 \cdot (e^{-\frac{r^2}{2\sigma^2}}) \cdot dr = (\sqrt{\pi/2}) \cdot \sigma \qquad 2.6.14$$

The mean-square value of R is :

$$E[R^2] = E[x^2 + y^2] = E[x^2] + E[y^2] \qquad 2.6.15$$

where: $E[X^2] = \sigma^2 - \bar{x}^2$ and $E[Y^2] = \sigma^2 - \bar{y}^2$

but $\bar{x}, \bar{y}$ the mean values of x and y respectively and equal to zero then, eqn.2.6.15 it can be written :

$$E[R^2] = \bar{r}^2 = \sigma^2 + \sigma^2 = 2\sigma^2 \qquad 2.6.16$$

where $\bar{r}^2$ mean-square value of R. The variance of R is given by :

$$\sigma_R^2 = E[R^2] - (\bar{r})^2 = 2\sigma^2 - \frac{\pi}{2} \cdot \sigma^2 = \sigma^2[2 - \pi/2] \qquad 2.6.17$$

The median value of the Rayleigh distribution occurs, at $r = r_m$, at the point where the CDF (cumulative distribution function) $f(r)$ in eqn.2.6.13 is equal to (0.5). Therefore :

$$f(r_m) = 0.5 = 1 - \exp(-\frac{r^2_m}{2\sigma^2}) \Rightarrow \exp(-\frac{r^2_m}{2\sigma^2}) = 0.5 \Rightarrow \ln(0.5) = -\frac{r^2_m}{2\sigma^2} \Rightarrow$$
$$r_m^2 = -2\sigma^2 \cdot \ln(0.5) \Rightarrow r_m = \sigma \cdot (\sqrt{2 \cdot \ln(2)}) = 1.17741 \cdot \sigma$$

where σ^2 is the variance of Gaussian random variables used in the derivation of Rayleigh fading.

If $\sigma^2 = 1 \Rightarrow \sigma = 1 \Rightarrow r_m = \sqrt{2 \cdot \ln(2)} \Rightarrow r_m = 10 \cdot \log(\sqrt{2 \cdot \ln(2)})$ (dB) 2.6.18

Now a completely analogous characterisation of the time-variant multipath channel begins in the frequency domain by taking the Fourier transform of $c(\tau; t)$, we obtain the time-variant transfer function $C(f; t)$ where f is the frequency variable. Thus :

$$C(f,t) = \int_{-\infty}^{\infty} c(\tau\ ;t)\cdot e^{-j2\pi f\tau}\cdot d\tau \qquad 2.6.19$$

Since $c(\tau; t)$ is a complex-valued zero mean Gaussian random process in the t variable, it follows that $C(f\ ;\ t)$ also has the same statistics. Thus, under the assumption that the channel is wide-sense-stationary, we define the auto correlation function:

$$\Phi_c(f_1,f_2;\Delta t) = 0.5\cdot E[C^*(f_1;t)\cdot C(f_2;t+\Delta t)] \qquad 2.6.20$$

Since $C(f\ ;\ t)$ is the Fourier transform of $c(\tau; t)$, it is not surprising to find that $\Phi_c(f_1,f_2,\Delta t)$ is related to $\phi_c(\tau;\Delta t)$ by the Fourier transform. The relationship is easily established by substituting eqn.2.6.19 into eqn.2.6.20, thus :

$$\Phi_c(f_1,f_2;\Delta t) = \frac{1}{2}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \mathrm{E}[c^*(\tau_1;t)\cdot c\ (\tau_2;t+\Delta t)]\cdot\exp(j2\pi(f_1\cdot\tau_1 - f_2\cdot\tau_2))\cdot d\tau_1\cdot d\tau_2$$

Or:

$$\Phi_c(f_1,f_2;\Delta t) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\phi_c(\tau_1;\Delta t)\cdot\delta\ (\tau_1-\tau_2)\cdot\exp(j2\pi\ (f_1\cdot\tau_1 - f_2\cdot\tau_2))\cdot d\tau_1\cdot d\tau_2$$

then,

$$\Phi_c(f_1,f_2;\Delta t) = \int_{-\infty}^{\infty}\phi_c(\tau_1;\Delta t)\cdot\exp(j2\pi\ (f_1-f_2)\cdot\tau_1\ )\cdot d\tau_1$$

where: $\int_{-\infty}^{\infty}\delta\ (\tau_1-\tau_2)\cdot\exp(j2\pi\ (f_1\cdot\tau_1 - f_2\cdot\tau_2))\cdot d\tau_2 = \exp(j2\pi\ (f_1-f_2)\tau_1\ )$

and $\quad 0.5E[c^*(\tau_1;t)\cdot c\ (\tau_2;t+\Delta t)] = \phi_c(\tau_1;\Delta t)\delta\ (\tau_1-\tau_2)$

Thus: $\quad \Delta f\ = f_2 - f_1 \Rightarrow$

$$\Phi_c(f_1,f_2;\Delta t) = \int_{-\infty}^{\infty}\phi_c\ (\tau_1;\Delta t)\cdot\exp(-j2\pi\,\Delta f\cdot\tau_1\ )\cdot d\tau_1\ \equiv \Phi_c(\Delta f;\Delta t) \qquad 2.6.21$$

From eqn.2.6.21, we observe that $\Phi_c\ (\Delta f;\Delta t)$ is the Fourier transform of the multipath intensity profile. Furthermore, the assumption of uncorrelated scattering implies that the auto-correlation function of $C(f\ ;\ t)$ in frequency is a function of only the frequency difference $\Delta f = f_2 - f_1$ [1]. Therefore it is appropriate to call $\Phi_c\ (\Delta f;\Delta t)$ the spaced-frequency spaced-time correlation function of the channel. It can be measured in practice by transmitting a pair of sinusoids separated by Δf and cross-correlating the two separately received signals with a relative delay Δt.

From qn.2.6.21 at $\Delta t = 0$ then, with $\Phi_c(\Delta f : 0) \equiv \Phi_c(\Delta f)$ and $\phi_c(\tau : 0) \equiv \phi_c(\tau)$ the transform relation is simply:

$$\Phi_c(\Delta f) = \int_{-\infty}^{\infty} \phi_c(\tau) \cdot e^{-j2\pi \Delta f \cdot \tau} \cdot d\tau \qquad 2.6.22$$

Fig.2.6.3 shows this relationship. The value $\phi_c(\tau\ ;0)$ is the mean strength (intensity) of the channel versus delay time, "multipath profile". The notion that fading is independent at each delay is physically reasonable for most channels. The transform relation between $\Phi_c(\Delta f)$ and $\phi_c(\tau)$ involving the variables τ and Δf, shows that the selectivity of the channel as measured by the width of $\Phi_c(\Delta f)$ versus Δf is inversely reciprocal to the width of the channel response, the value described by the multipath spread T_M. The reciprocal of the multipath spread is a measure of the coherence bandwidth of the channel. That is:

$$(\Delta f)_c \approx 1/T_M \qquad 2.6.23$$

where $(\Delta f)_c$ denotes the coherence bandwidth. Thus two sinusoids with frequency separation greater than $(\Delta f)_c$ are affected differently by the channel. When an information bearing signal is transmitted through the channel, if $(\Delta f)_c$ is small in comparison to the bandwidth of the transmitted signal ($w_B = 1/T_B$), i.e. $(\Delta f_c) << w_B$ or $T_B << T_M$, the channel is said to be frequency-selective. In this case the signal is severely distorted by the channel. On the other hand if $(\Delta f)_c$ is large in comparison to the bandwidth of the transmitted signal, i.e. $(\Delta f_c) >> w_B$ or $T_B >> T_M$, the channel is said to be frequency-non selective fading.

We now focus our attention on the time variations of the channel as measured by the parameter Δt in $\Phi_c(\Delta f ; \Delta t)$. The time variations in the channel are evidenced as a Doppler broadening and, perhaps, in addition as a Doppler shift of a spectral line. In order to relate the Doppler effects to the time variations of the channel, we define the Fourier transform of $\Phi_c(\Delta f ; \Delta t)$ with respect to the variable Δt to be function of $S_c(\Delta f ; \lambda)$, that is :

$$S_c(\Delta f ; \lambda) = \int_{-\infty}^{\infty} \Phi_c(\Delta f ; \Delta t) \cdot e^{-j2\pi \lambda \Delta t} \cdot d(\Delta t) \qquad 2.6.24$$

with $\Delta f = 0$ and $S_c(0;\lambda) \equiv S_c(\lambda)$, then:

$$S_c(0\ ;\lambda) = S_c(\lambda) = \int_{-\infty}^{\infty} \Phi_c(\Delta t) \cdot e^{-j2\pi \lambda \Delta t} \cdot d(\Delta t) \qquad 2.6.25$$

The function $S_c(\lambda)$ is a power spectrum that gives the signal intensity as a function of the Doppler frequency (λ). Hence $S_c(\lambda)$ is called the Doppler power spectrum of the

channel. From eqn.2.6.25 we observe that if the channel is time-invariant, $\Phi_c(\Delta t) = 1$ and $S_c(\lambda)$ becomes equal to the delta function $\delta(\lambda)$. Therefore when there are no time variations in the channel, there is no spectral broadening observed in the transmission of a pure frequency tone. The range of values of (λ) over which $S_c(\lambda)$ is essentially non-zero is called the Doppler spread B_D of the channel. Since $S_c(\lambda)$ is related to $\Phi_c(\Delta t)$ by the Fourier transform, the reciprocal of B_D is a measure of the coherence time of the channel. That is :

$$(\Delta t)_c \approx 1/B_D \qquad 2.6.26$$

where $(\Delta t)_c$, denotes the coherence time. Clearly a slowly changing channel has a large coherence time or, equivalently a small Doppler spread. Fig.2.6.4 illustrates the relationship between $\Phi_c(\Delta t)$ and $S_c(\lambda)$. we have now established a Fourier transform relationship between $\Phi_c(\Delta f;\Delta t)$ and $\phi_c(\tau;\Delta t)$ involving the variables $(\tau;\Delta f)$, and a Fourier transform relationship between $\Phi_c(\Delta f;\Delta t)$ and $S_c(\Delta f;\lambda)$ involving the variables $(\Delta t;\lambda)$. There are two additional Fourier transform relationships that we can define which serve to relate $\phi_c(\tau;\Delta t)$ to $S_c(\Delta f;\lambda)$ and, thus close the loop. The desired relationships is obtained by defining a new function, denoted as $S(\tau;\lambda)$, to be the Fourier transform of $\phi_c(\tau;\Delta t)$ in the Δt variable. That is :

$$S(\tau;\lambda) = \int_{-\infty}^{\infty} \phi_c(\tau;\Delta t)\cdot e^{-j2\pi\lambda\Delta t}\cdot d(\Delta t) \qquad 2.6.27$$

It follows that $s(\tau;\lambda)$ and $S_c(\Delta f;\lambda)$ are a Fourier transform pair. That is :

$$S(\tau;\lambda) = \int_{-\infty}^{\infty} S_c(\Delta f;\lambda)\cdot e^{j2\pi\tau\Delta f}\cdot d(\Delta f)$$

Furthermore, $s(\tau;\lambda)$ and $\Phi_c(\Delta f;\Delta t)$ are related by the double Fourier transform :

$$S(\tau;\lambda) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \Phi_c(\Delta f;\Delta t)\cdot e^{-j2\pi\lambda\Delta t}\cdot e^{-j2\pi\tau\Delta f}\cdot d(\Delta t)\cdot d(\Delta f) \qquad 2.6.28$$

This new function $s(\tau;\lambda)$ is called the scattering function of the channel [1]. It provides us with a measure of the average power output of the channel as a function of the delay τ and the Doppler frequency λ. For an unmodulated sine wave transmitted at relative frequency f, for which the received complex envelope is $C(f\ ;\ t)$, the complex auto correlation of the wave form received at frequency f is $\Phi_f(0;\Delta t)$, and its power spectrum is

$$S_f(\lambda) = \int_{-\infty}^{\infty} \Phi_f(0;\Delta t)\cdot e^{-j2\pi\lambda\Delta t}\cdot d(\Delta t) \qquad \Rightarrow$$

$$S_f(\lambda) = \int_{-\infty}^{\infty} \phi_T(\tau\ ; \Delta t) \cdot e^{-j\,2\pi\lambda\,\Delta t} \cdot d\tau \cdot d(\Delta t) = \int S(\tau\ ; \lambda) \cdot d\tau \qquad 2.6.29$$

That is, the observed power spectrum is a sum of the spectra contributed by all the delays. The rate of Rayleigh envelope fading across the median envelope level is actually proportional to a specific definition of "rms bandwidth" f_{rms} of $S_f(\lambda)$ [1,3] :

" fade rate = $1.475 f_{rms}$ (average rate of down going crossing of median level), [17] ".

Note that the fade rate can easily be visually extracted from chart recordings. In order to process a symbol waveform with a filter matched to the waveform as transmitted, two separated criteria must be satisfied :

1. That there be essentially no loss of coherence over the symbol as received.

2. That there be essentially no frequency selective distortion of the symbol. This leads to a double inequality :

$$T_M << T << \frac{1}{B_D} \qquad 2.6.30$$

where T is the duration of symbol wave form and B_D Doppler spread.

In turn, both inequalities can be satisfied simultaneously only if the "spread factor" L defined by [3] :

$$L = T_M \cdot B_D << 1 \qquad 2.6.31$$

The characterisation given above describes the degree of distortion of an isolated symbol. Continuous communication involves sequences of symbols and side-by-side channel frequency allocations, so that we must also be concerned respectively with time smear that may cause inter-symbol interference or the frequency smear that may cause adjacent channel interference.

Finally for each received symbol, there is an interval of duration $(T \cdots T_M)$ which contains most of that symbol's energy and which is free of inter-symbol interference. The simple expedient adopted in a modem design is to use filters, prior to the decision processes, that have an impulse response selected to be shorter than $(T \cdots T_M)$; with proper synchronisation, this assures decision voltages free of inter symbol interference.

On the other hand, designs that can cope with the inter-symbol interference that occurs when $(T \cdots T_M)$ are extremely important in fading channel modem design, motivated by considerable other benefits that can be attributed to using single high-speed serial streams to carry the desired traffic rather than multiple parallel slower speed streams [3,17,23,60].

2.6.1 The Effect Of Signal Characteristics On The Choice Of A Channel Model

The equivalent lowpass received signal, exclusive of additive noise, may be expressed either in terms of the time domain variables $c(\tau\ ; t)$ and $u(t)$ as :

$$r(t) = \int_{-\infty}^{\infty} c(\tau\,; t) \cdot u(t - \tau\,) \cdot d\tau \qquad 2.6.32$$

or in terms of the frequency by functions $C(f\,;\,t)$ and $U(f\,)$ as :

$$r(t) = \int_{-\infty}^{\infty} C(f\,;t) \cdot U(f\,) \cdot e^{j2\pi\, f\, t} \cdot df \qquad 2.6.33$$

where $u(t)$ is the equivalent lowpass signal transmitted over the channel and $U(f)$ denote its frequency content. Suppose we are transmitting digital information over the channel by modulating (either in amplitude, or in phase, or both) the basic pulse $u(t)$ at a rate $1/T$, where T is the signalling interval. It is apparent from eqn.2.6.33 that the time-variant channel characterised by the transfer function $C(f\,;\,t)$ distorts the signal $U(f)$. If $U(f)$ has a bandwidth W greater than the coherence bandwidth $(\Delta f\,)_c$ of the channel, $U(f)$ is subjected to different gains and phase shifts across the band. In such a case the channel is said to be frequency selective. Additional distortion is caused by the time variations in $C(f\,;\,t)$. This type of distortion is evidenced as a variation in the received signal strength and has been termed fading. The effect of the channel on the transmitted signal $u(t)$ is a function of our choice of signal band-width and signal duration. For example, if we selected the signalling interval T to satisfy the condition $T >> T_M$, the channel introduces a negligible amount of inter-symbol interference. If the band-width of the signal pulse $u(t)$ is $W \approx 1/T$, the condition $T >> T_M$ implies that :

$$W << \frac{1}{T_M} \approx (\Delta f\,)_c \qquad 2.6.34$$

that is, the signal band-width W is much smaller than the coherence bandwidth of the channel. Hence the channel is frequency-non-selective. In other words, all of the frequency components in $U(f)$ undergo the same attenuation and phase shift in transmission through the channel. But this implies that, within the bandwidth occupied by $U(f)$, the time-variant transfer function $C(f\,;\,t)$ of the channel is a complex-valued constant in the frequency variable. Since $U(f)$ has its frequency content concentrated in the vicinity of $f=0$, then $C(f\,;\,t) = C(0\,;\,t)$. Consequently eqn.2.6.33 reduces to :

$$r(t) = C(0\,;t) \cdot \int_{-\infty}^{\infty} U(f\,) \cdot e^{j2\pi\, f\, t} \cdot df = C(0\,;\,t) \cdot u(t) \qquad 2.6.35$$

Thus, when the signal band-width W is much smaller than the coherence bandwidth $(\Delta f\,)_c$ of the channel, the received signal is simply the transmitted signal multiplied by a complex-valued Gaussian process $C(0\,;\,t)$, which represent the time-variant characteristics

of the channel. In this case, we say that the multipath components in the received signal are not resolvable because $W << (\Delta f)_c$. The transfer function *C(0 ; t)* for a frequency non-selective channel may be expressed in the form

$$C(0;t) = \alpha(t) \cdot e^{-j\phi(t)} \qquad 2.6.36$$

where $\alpha(t)$ represents the envelope and $\phi(t)$ represents the phase of the equivalent lowpass channel. When *C(0 ; t)* is modelled as a zero mean complex-valued Gaussian random process, the envelope $\alpha(t)$ is Rayleigh distributed for any fixed value of t and $\phi(t)$ is uniform distributed over the interval $(-\pi, \pi)$. The rapidity of the fading on the frequency non-selective channel is determined either from the correlation function $\phi_c(\Delta t)$ or from the Doppler power spectrum $S_c(\lambda)$. Alternatively, either of the channel parameters $(\Delta t)_c$ or B_D can be used to characterised the rapidity of the fading.

For example, suppose it is possible to select the signal bandwidth W to satisfy the condition $W << (\Delta f)_c$ and the signalling interval T to satisfy the condition $T << (\Delta t)_c$.

Since T is smaller than the coherence time of the channel, the channel attenuation and phase shift are essentially fixed for the duration of at least one signalling interval. When this condition holds, we call the channel a slowly fading channel. Furthermore, when $W \approx 1/T$, the conditions that the channel is frequency non-selective and slowly fading imply that the product of T_M and B_D must satisfy the condition $(T_M \cdot B_D < 1)$.

The product $(T_M \cdot B_D)$ is called the spread factor of the channel. If $(T_M \cdot B_D < 1)$, the channel is said to be under spread; otherwise it is over spread. The multipath spread, the Doppler spread and the spread factor are listed in Table 2.6.1 for several channels. We observe from this table that several radio channels, including the moon when used as a passive reflector, are under spread. Consequently it is possible to select the signal *u(t)* such that these channels are frequency non-selective and slowly fading. The slow-fading condition implies that the channel characteristics vary sufficiently slowly that they can be measured [1].

2.7 HF IONOSPHERIC CHANNEL CLASSIFICATION

Due to the number and variety of parameters that need to be considered (and indeed their relative importance) in HF channel classifications, there is at present, no standard classification for HF channels. The CCIR, for example, have classified HF channels according to their multipath delays and frequency spread as good, average, poor and flutter fading channels [11] as follows :

1. Good conditions :

differential time delay	0.5 msec.
frequency spread	0.1 Hz.

2. Moderate conditions :

differential time delay	1.0 msec.
frequency spread	0.5 Hz.

3. Poor conditions :

differential time delay	2.0 msec.
frequency spread	1.0 Hz.

4. Flutter fading :

differential time delay	5 msec.
frequency spread	10.0 Hz.

These channel models are based on two skywaves where each skywave is subjected to independent Rayleigh fading and has the same frequency spread. The channels used in this work are based on the CCIR model and references [11,12,13]. Four type of channel have been chosen with their parameters summarised in Table (2.7.1). All these channels could be the worst case for testing the performance of the detection and are probably suitable for testing the modem operating over a real HF radio channel.

2.8 HF CHANNEL MODEL

Testing of HF modems can be performed in two ways : First, live testing can be performed over real HF links. Second, testing can be performed using a channel simulator in a laboratory environment. The use of on-the-air testing is unsatisfactory for several reasons :

1. It is high cost and involves uncontrolled channel parameters.
2. It is virtually impossible to compare the performance of different modems accurately since, due to the time varying nature of the HF radio medium, they cannot be tested over the same channel.
3. It is also difficult to test modems for 'worst case' conditions since these could occur at random, making scheduled testing impossible.

The flexibilities needed are achieved instead by the use of laboratory simulators of fading channel conditions. In addition to testing engineered equipment, such simulators allow experiments to test new concepts under controlled statistical conditions. But this method requires the development and implementation of an accurate theoretical model.
Laboratory simulators exist in software or hardware [12,13]. However, care should be taken to ensure that the model on which the simulator is based on a valid representation of an HF radio channel.

2.8.1 Gaussian-Scatter HF Channel Model

The most commonly used channel simulator and the one recommended by the international radio consultative committee CCIR is that proposed by Watterson [16]. The simulators built to date have been aimed at the narrow-band (<10*KHz*) channel, and short transmission times (< 10 minutes) are considered. But in this work the channels of interest are band-limited voice-band channels (=3*KHz*), and transmission times are in the order of tens of seconds. The objective here is to present the channel model in its given form, ensuring that the assumptions made are both reasonable and valid for the research being carried out.

The sampling theorem representation of the channel in the form of a tapped delay line with uniform tap spacings is used. This is a finite impulse response (FIR) filter in signal processing terms, as indicated in Fig.2.8.1, a block diagram of the stationary Gaussian-scatter HF channel ionospheric model. The input transmitted signal *E(t)* is fed to an ideal delay line with several adjustable taps, numbered 1,2,....., i , ..., n, one for each ionospheric propagation mode or path. The delayed signal is *E(t-iT)*. This signal is modulated in amplitude and phase by an appropriate complex random function $G_i(t)$. $G_i(t)$ is composed of $B_{ia}(t)$ and $B_{ib}(t)$, which are sampled functions of two independent complex gaussian random processes. Each tap-gain function is defined by [16,11]:

$$G_i(t) = B_{ia}(t)\cdot e^{[j(w-w_{ia})t]} + B_{ib}(t)\cdot e^{[j(w-w_{ib})t]} \qquad 2.8.1$$

where w_{ia} and w_{ib} are the frequency shifts (Doppler effect), and the *a* and *b* subscripts identify two magnetoinic components that are generally present in each mode or path. $B_{ia}(t)$ and $B_{ib}(t)$ are sample functions of two independent complex Gaussian ergodic random processes. The exponential factors in eqn.2.8.1 are included to provide the desired frequency-shifts (Doppler), for the magnetonic components in the tap-gain spectrum.
Each tap-gain function has a spectrum *Q(w)*, that in general consists of the sum of two magnetoionic components, each of which is a Gaussian function of frequency, and given by [11]:

$$Q_i(w) = \frac{1}{A_{ia}\sqrt{2\pi}\cdot\sigma_{ia}}\exp[-\frac{(w-w_{ia})^2}{2\sigma_{ia}^2}] + \frac{1}{A_{ib}\sqrt{2\pi}\cdot\sigma_{ib}}\exp[-\frac{(w-w_{ib})^2}{2\sigma_{ib}^2}] \quad 2.8.2$$

where A_{ia} and A_{ib} the component attenuations, and the frequency spread on each component is determined by $2\sigma_{ia}$ and $2\sigma_{ib}$.

The tap-gain function described by eqns.2.8.1 and 2.8.2 in general is that, it applies when the spectra of the two magnetoionic components are significantly different and only one of the two terms in eqns. 2.8.1 and 2.8.2 is required in the following two cases :

- For some low-rays, the frequency-shift and frequency spread of the two magneto-ionic components are nearly equal, their spectra nearly match, and a single term can be used with the tap-gain spectrum in Fig.2.8.2.
- The two magneto-ionic components in high rays often have a significantly large difference in delay. In this case, separate delay-line taps with appropriate spacing should be used, with each of the two corresponding tap-gain functions and spectra consisting of a single term, as in Fig.2.8.2.

The received signal $R_j(t)$ for the j[th] path is :

$$R_j(t) = G_j(t) \cdot E(t - T_j) \qquad 2.8.3$$

As many propagation modes are possible, the total received signal *R(t)* is the sum of the received signal from each path and an additive noise *n(t)*, [11]

$$R(t) = \sum_j G_j(t) \cdot E(t - T_j) + n(t) \qquad 2.8.4$$

The time-variant tap weights $\{G_n(t)\}$ are assumed to have the following properties [1,11]

1. Each tap-gain function is a zero mean complex-valued stationary gaussian random process.
2. Each tap-gain function is independent of the other tap-gain functions.
3. Each tap-gain spectrum (generally) consists of the sum of two Gaussian functions of frequency as specified by eqn.2.8.2.

The first of these hypotheses gives rise to Rayleigh fading being imposed on each skywave where the signal envelope follows a Rayleigh distribution and the phase is uniformly distributed. The second hypothesis ensures that the fading on each skywave is uncorrelated, and as a consequence of the zero mean criterion, is also independent. The third hypothesis reflects the fact that, included in each skywave are the two magneto-ionic components, that is the ordinary and the extra-ordinary rays. However, if the frequency spreads of these two components are assumed to be equal and significantly larger than their frequency shifts (as on the day-time F-layer mode) [16], then a single Gaussian function is sufficient to represent the tap-gain spectrum (see Fig.2.8.2). The model used in this work assumes a single gaussian function for each tap-gain spectrum. Also, for further simplification, the average power and frequency spreads of the different sky-waves themselves are assumed to be equal, and the frequency spreads are also assumed to be large, compared to the frequency shifts (Doppler shifts). This simplified the model, although obviously not representing the exact nature of the HF channel, does reflect the 'worst case' type situations that could be encountered. For example, the assumption that

the average power in each sky wave is equal, leads to the worst fades, since these occur when two sky waves are of equal strength and are in phase opposition. The "equal frequency spread" assumption would not significantly affect the performance of the modem relative to its performance over a 'real' channel since the spreads being modelled are much larger than those observed on typical HF radio channels [10,17,60].

The frequency shifts could also be assumed to be taken care of by these large frequency spreads, and thus need not be included as a separate form of distortion. Moreover, even if there were an abnormally large shift in the mean carrier frequency, this would be detected and rectified by the carrier recovery circuit in the demodulator.

2.8.2 RAYLEIGH FADING MODEL

A single Rayleigh fading propagation path is modelled as it shown in Fig.2.8.3, this is achieved by phase shifting the input baseband signal to produce two equal-amplitude versions of the input signal in phase quadrature, subjecting each to the Gaussian modulation process, and then linearly summing the two modulated quadrature signals.

This 90^0 phase shift should be carried out over the entire band-width of the input signal, or equivalently a Hilbert transform of the input signal should be performed.

$q_1(t)$ and $q_2(t)$ in Fig.2.8.3 are two random processes. These random processes should be Gaussian with zero mean, the same variance and statistically independent. The shape of their power spectra must be Gaussian, having the same root mean square (rms) frequency f_{rms} Fig.2.8.4. Therefore the power spectrum of $q_1(t)$ and $q_2(t)$ are given by [10,17] :

$$\left|Q_1(f)\right|^2 = \left|Q_2(f)\right|^2 = \exp(-\frac{f^2}{2f_{rms}^2}) \qquad 2.8.5$$

The fading rate can be controlled by the bandwidth of the power spectra of the Gaussian variables $q_1(t)$ and $q_2(t)$. The frequency spread, f_{sp} introduced by $q_1(t)$ and $q_2(t)$ into an unmodulated carrier is defined as the width of the power spectrum and is given by :

$$f_{sp} = 2 \cdot f_{rms} \qquad 2.8.6$$

The fading rate is defined (for a single carrier) as the average number of downward crossings per unit time of the envelope through the median value. The fading rate (f_e) is related to the *rms* frequency by :

$$f_{rms} = \frac{f_e}{1.475} \Rightarrow f_e = 1.475 f_{rms} \qquad 2.8.7$$

From eqns. 2.8.6 and 2.8.7 we can write :

$$f_{sp} = \frac{2f_e}{1.475} \approx 1.356 f_e \Rightarrow f_e = \frac{f_{sp}}{1.356} \qquad 2.8.8$$

which mean that $f_{sp} = 1Hz \Rightarrow f_e = 1/1.356 \approx .737\, fade/\sec = 44.25$ fades per minute.

The two processes $q_1(t)$ and $q_2(t)$ are generated by filtering a zero mean white Gaussian noise signal $N_1(t)$, $N_2(t)$ as shown in Fig.2.8.5, both with zero mean and unity two-sided power spectral density. The filters should also have a Gaussian frequency response matching the power spectrum of the Gaussian variable $q_1(t)$ and $q_2(t)$. 5-pole Bessel filters have been chosen which meet the above requirement and their detailed design is given in Appendix A. The impulse response of the Bessel filter approaches Gaussian when the order of the filter is sufficiently large and the frequency response is given by [18,19]:

$$W(f) = \exp(-\frac{f^2}{4f_{rms}^2}) \qquad 2.8.9$$

Therefore, a bessel filter is used in the channel simulator to provide the necessary shaping to the random process $q(t)$. In the HF channel simulator model used in this work there are a 3-sky waves and therefore, it requires six random processes $q_i(t)$ for i=1,2,.....,6 and each of these are similarly generated and the variance of a six variables are equal to $1/6 \approx 0.167$. This value of variance for each individual process ensures that the total variance of the 3-sky wave channel is unity. A block diagram of the filter used is given in Fig.A.1 (Appendix A). From Nyquist theorem the sampling frequency of $q_i(t)$ in digital implementation of the bessel filter, should be more than twice the highest frequency contained in $q_i(t)$. $q_i(t)$ has a Gaussian spectrum and the maximum frequency contained in $q_i(t)$ theoretically is infinite. However, for the highest frequency spread of interest, i.e. $2Hz$, the cut-off frequency of the filter is $f_c = 1.17Hz$, and gives rise to a $+3dB$ band-width of $2.34Hz$. This implies that a sampling frequency of more than $10Hz$, is adequate for accurate representation of the filters in the digital domain. The sampling rate of the fading processes $q_i(t)$ is determined by the transmission bit rate of the modem. For testing a *2400 baud* digital data modem on the HF channel every signal sample should be modified by the fading, hence forcing the sampling frequency of the $q_i(t)$ to be $2400Hz$ or more. In order to be consistent with the sampling frequency, the roots of the digital filter must correspond to this sampling frequency and at the same time it is necessary to see these roots are not too close to the unit circle in Z-plane. Unfortunately at this high sampling frequency the pole locations of such filters in the Z-plane are very close to the unit circle. This problem is solved by having an intermediate sampling frequency for the $q_i(t)$ and obtaining the remaining samples by a process of linear interpolation. Thus $q_i(t)$ has been sampled at $100Hz$, that is a sampling interval of $10msec$. Where the interpolation considered here is of linear form, when the two adjacent points of $q_i(t)$, (such as

$q_{i,j}$ & $q_{i,j-1}$) are joined by a straight line, this line is then sampled to provide the sequence of the required q_i, such that :

$$q_i = q_{i,j-1} + i \cdot \delta_{i,j}$$

where $$\delta_{i,j} = -(q_{i,j-1} - q_{i,j})/\mu = (q_{i,j} - q_{i,j-1})/\mu$$

μ represents the interpolation step, a positive integer. Thus, when *100Hz* is employed as the sampling frequency for the bessel filter then by interpolating *48* ($\mu = 48$) between samples the required sampling rate of 4800 *samples/seconds* can be obtained.

The constant (G_0) at the input to the bessel filter (see Fig.A.1), is employed to change the variance of the $q_i(t)$ to their required values. The aim here is to ensure that the total mean power input to the model is equal to the total mean power output. Thus the total variance of all $q_i(t)$'s should be unity. Combined with the requirement that the variance of all the $q_i(t)$'s should be equal, this implies that in a 3-sky wave channel, each $q_i(t)$ should have a variance of *1/6*, where as in a 2-sky wave channel, the variance of each $q_i(t)$ should be *1/4*. This is because six sequences ($q_1(t) \cdots q_6(t)$) are required for a 3-sky wave channel and four sequences ($q_1(t) \cdots q_4(t)$) are required for 2-sky wave channel. The values which give the required variances in the $q_i(t)$ for channel 1, 2, 3 and 4 are, respectively, *19500, 15156, 308538* and *204933*.

Finally Fig.2.8.6 shows the model of a 3-sky wave HF channel simulator used in this work. The two signals $q_1(t)$ and $q_2(t)$ give the first sky wave, the two signals $q_3(t)$ and $q_4(t)$ give the second sky wave and the two signals $q_5(t)$ and $q_6(t)$ give the third sky wave where each of these signals are statistically independent of each other, but possess the same properties. If the signal at the input of Fig.2.8.6 is *x(t)* then the output signal *Z(t)* is :

$$z(t) = x(t) \cdot q_1(t) + \hat{x}(t) \cdot q_2(t) + x(t-\tau_1) \cdot q_3(t) + \hat{x}(t-\tau_1) \cdot q_4(t) + x(t-\tau_2) \cdot q_5(t) + \hat{x}(t-\tau_2) \cdot q_6(t)$$

where $\hat{x}(t)$ is the Hilbert-transform of x(t).τ_1 and τ_2 are time delay between the second sky wave and the third sky wave with respect to the first sky wave, respectively.

2.9 RESULT OF TESTING HF CHANNEL SIMULATOR

The results of the tests carried out on the simulated channel are summarised in Table 2.9 and Figs.2.9.1-2.9.8. Table 2.9 shows the measured mean value, variance and the number of fades relative to the median value (r_m) of the sequences q_i for the 4

channels model, with different values of the frequency spread, and different seed number. The median value is obtained as follows :

The cumulative distribution function of the Rayleigh distribution, is given by eqn.2.6.13. the median value is the point at which the CDF is 0.5, i.e. at $r=r_m$ then $r_m = \sigma \cdot \sqrt{2 \cdot \ln(2)}$, where σ^2 is the variance of Gaussian random variables used in the derivation of Rayleigh fading. For example, in channel 1 the variance of each of the six sequences q_i is $\sigma^2 = 1/6 = 0.167 \Rightarrow r_m = 0.40825 \cdot 1.1774 = 0.4811$. In the same way the r_m for channel 2, 3 and 4 are *0.5887, 0.5887* and *0.8326* respectively. The measurement of fades can be made relative to this median value over the period considered in these tests of (*25seconds*) of the fading channel sampled at *4800* samples/sec. The theoretical number of fades can be obtained from $f_e = (f_{sp}/1.356)$, when $f_{sp} = 2Hz \Rightarrow f_e \approx 1.475 \text{fades/sec.} \approx$ $\approx 25 \cdot 1.475 = 37 fades$. Therefore for a duration of 25 seconds the number of fades is about 37*fades* for frequency spread 2*Hz*, 18*fades* for a frequency spread of 1*Hz*. It can be shown from Table 2.9 that the mean value of the sequences q_i are very close to zero and the sum of their variances is equal to unity. Figs.2.9.1-2.9.4 show the magnitude variation of the sequences $q_i(t)$ over a duration of 25*seconds* period for the four channel model, which is given by $Z = 10 \cdot \log_{10}[\sum_{i=1}^{n} \bar{q}_i^2]$, $n = 1, 2, \ldots$. It can be seen from these figures, that the amplitude variation of $q_i(t)$ is random, also these figures give some indication of the number of fades likely to be encountered in the respective channels. Where a fade is regarded as a downward crossing of the signal through its median value. In terms of the depths of the fades, there is at least one fade in excess of 30*dB*. These depths however, are expected to change when the other sky-waves are added to form the resultant HF channel model as can be seen from the previous figures. Figs.2.9.5-2.9.8 show the phase variations of the sequences $q_i(t)$, given by :

$$\theta = \tan^{-1}\left(\sum_{i=1}^{k} q_{2i} \Big/ \sum_{i=1}^{k} q_{2i-1}\right) \quad k = 1, \cdots, n/2$$

In all figures the phase varies between $-\pi$ &$+\pi$, which means that the phase variation is uniformly distributed. It can be seen that for channel 3 and 4 (the channel with $f_{sp} < 2Hz$), the phase variations are slower than in the other two channels. The random sequences q_i are obtained using the NAG routine G05DDF employing G05CBF for initialisation and shuffling random numbers. Toward the end of this work all measurements conducted over the previous channels model are based on the respective fading patterns in Figs.2.9.1-2.9.4.

Ionosphere Layer	Virtual Heigh (Km)	Electron Density per. cc	Critical Frequency (MHz)
D	50 ----- 90	$10^2 \cdots 10^4$	0.5
E	90 ----- 130	$10^4 \cdots 10^5$	4.0
F: F1	130 ---- 210	$10^5 \cdots 5 \cdot 10^5$	5.0
F2	250 ---- 350	$\tilde{}10^6$	10.0

TABLE 2.2.1 : Main Parameters Of Ionospheric Layers

Type Of Channel	Multipath Duration (sec.)	Doppler Spread (Hz)	Spread Factor
Short wave ionospheric propagation (HF)	$10^{-3} \cdots 10^{-2}$	$10^{-3} \cdots 1$	$10^{-4} \cdots 10^{-2}$
Ionospheric propagation under disturbed auroral conditions (HF)	$10^{-3} \cdots 10^{-2}$	$10 \cdots 100$	$10^{-2} \cdots 1$
Ionospheric forward scatter (VHF)	10^{-4}	10	10^{-3}
Tropospheric scatter (SHF)	10^{-6}	10	10^{-5}
Oribital scatter (X-Band)	10^{-4}	10^3	10^{-1}
Moon at max. libration (f_0=0.4Kc)	10^{-2}	10	10^{-1}

TABLTE 2.6.1 : Multipath Spread, Doppler Spread, and Spread Factor For Several Time-Variant Multipath Channels .

Type of Channel	Multipath Spread ($\tau 1, \tau 2$) (ms)	Frequency spread (Hz)	Number of Sky wave	DC-Gain	Seed number (ID)
CHANNEL 1	(1.1 & 3.0)	2.0	3	19500.0	75
CHANNEL 2	(3.0 & 0.0)	2.0	2	15156.0	57
CHANNEL 3	(2.0 & 0.0)	1.0	2	308538.0	57
CHANNEL 4	(0.0 & 0.0)	1.0	1	204933.0	57

TABLE 2.7.1 : Channel Used In The Tests

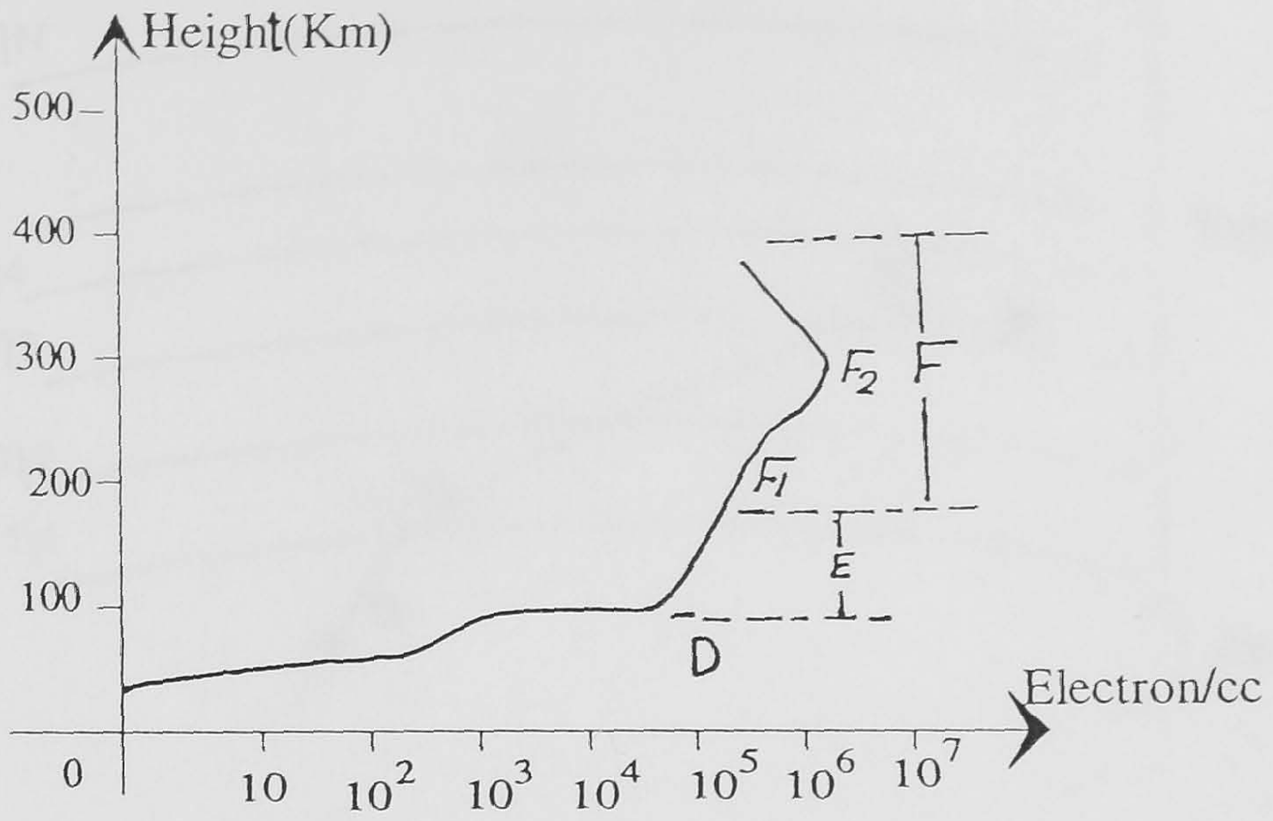


Fig. 2. 2.1 : Electron Density Vs. Height .

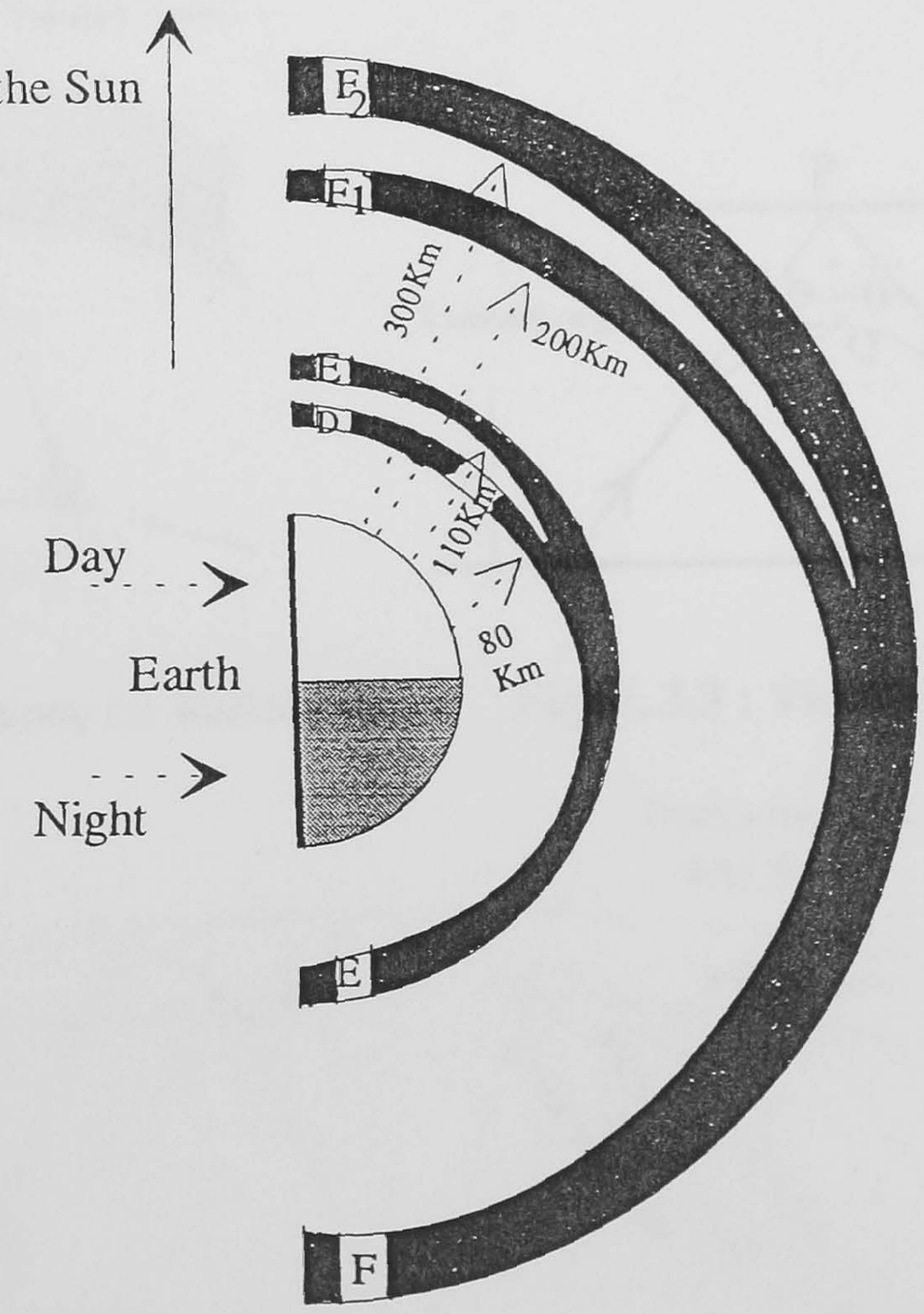


Fig. 2. 2.2 : Ionized Layers of Atmosphere as a Function of Nominal Height Above the Earth's Surface.

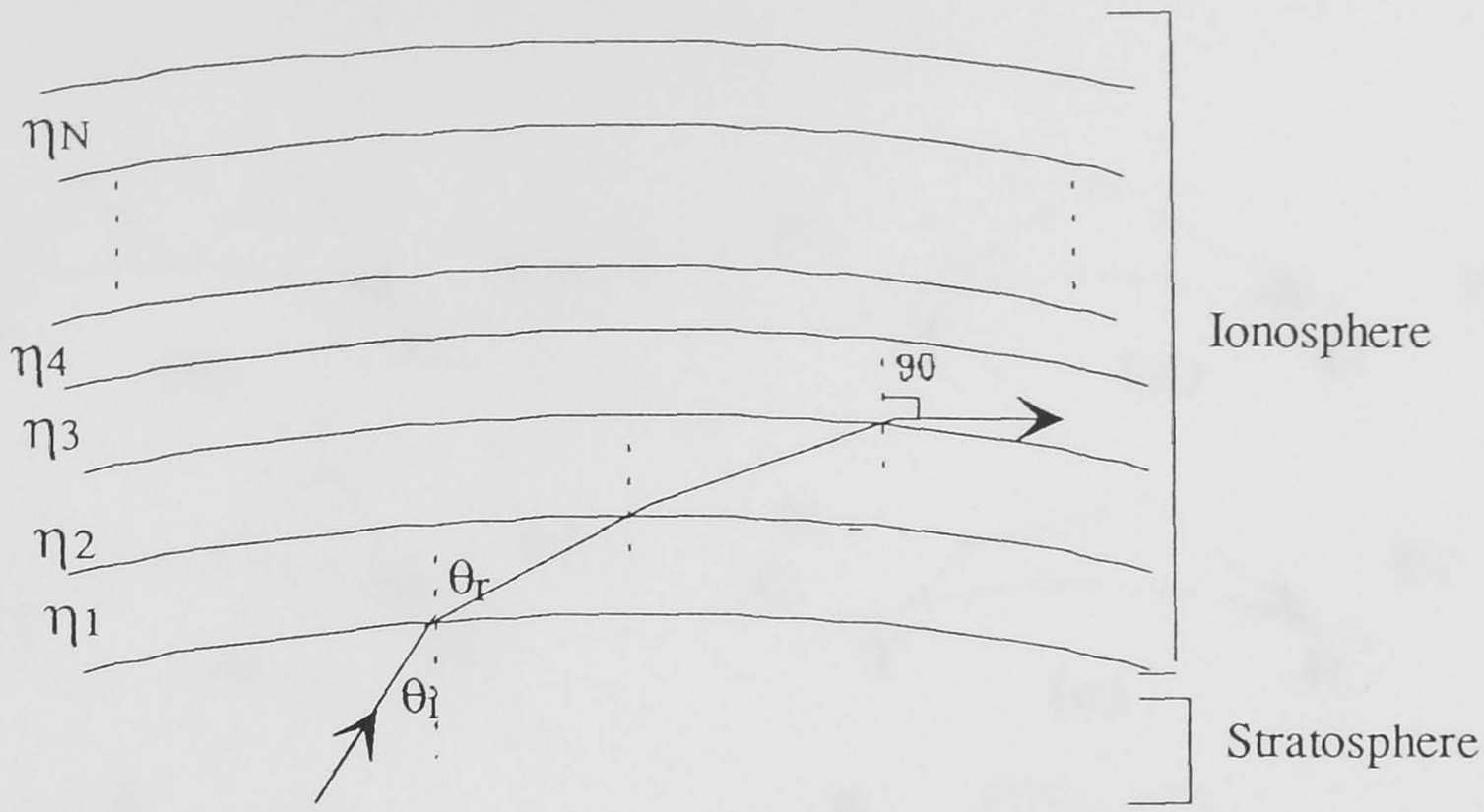


Fig. 2. 3.1 : Ionospheric Refraction Mechanism.

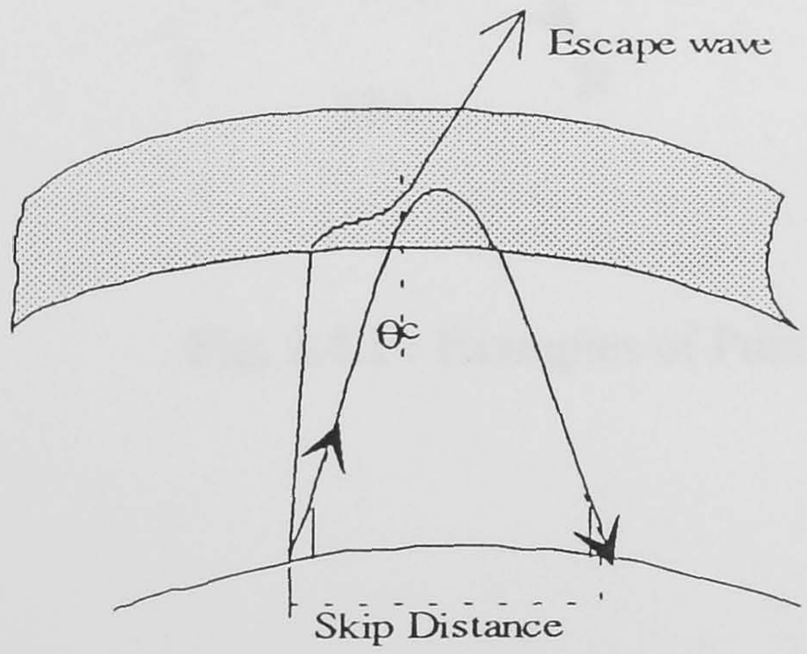


Fig. 2. 3.2 : Critical Conditions for Reflection.

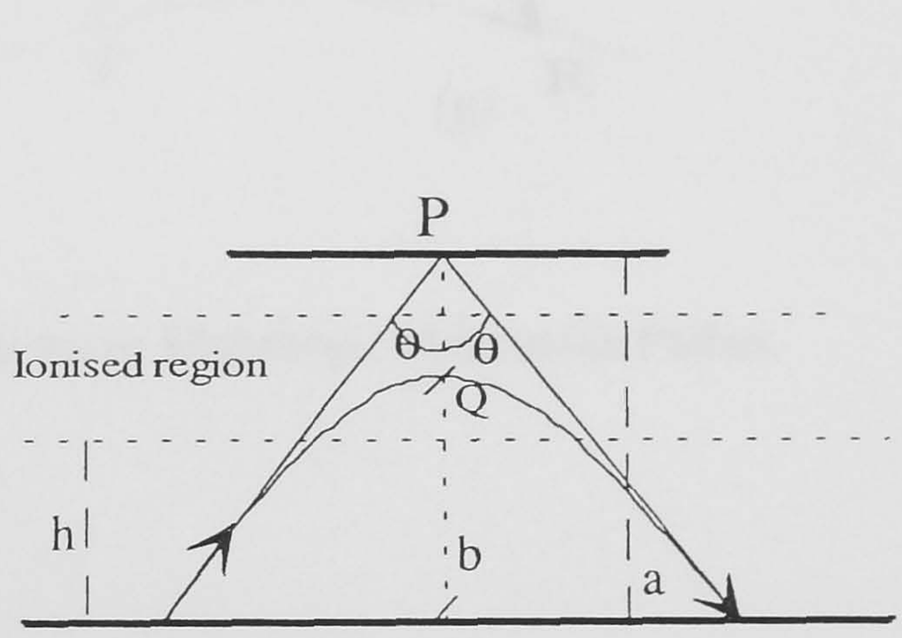


Fig. 2. 3.3 : Virtual Path.

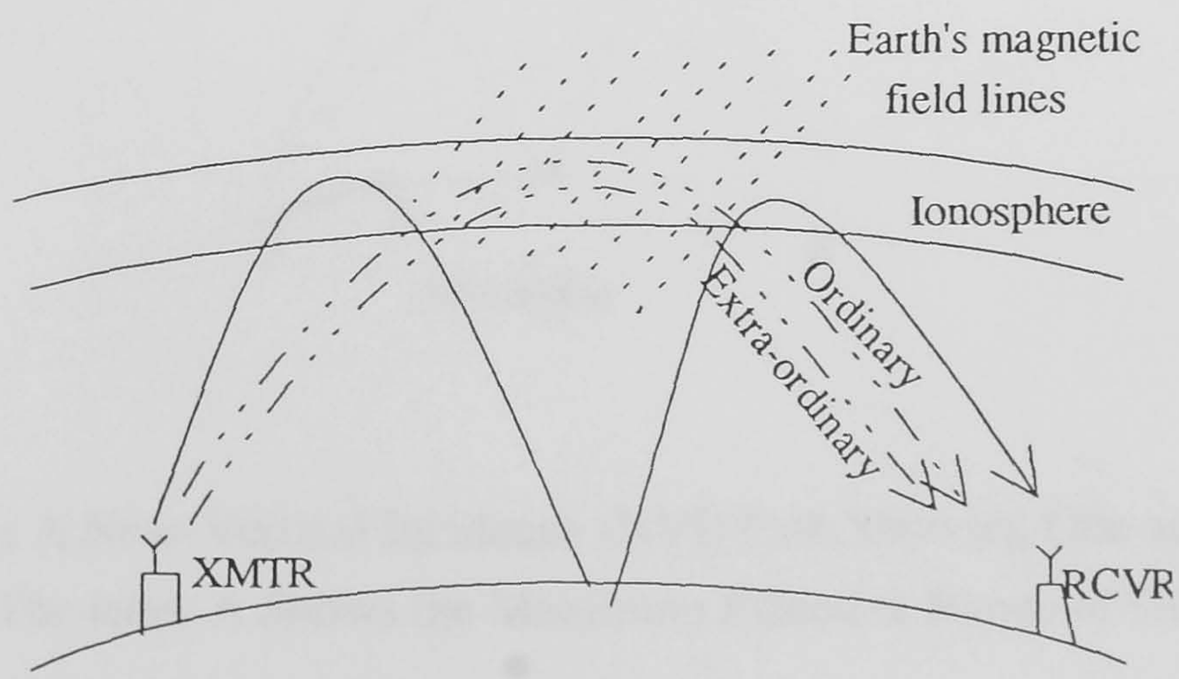


Fig. 2. 3.4 : The X- and O-waves , and Simultaneous of one Hop and Two Hops at Receiver.

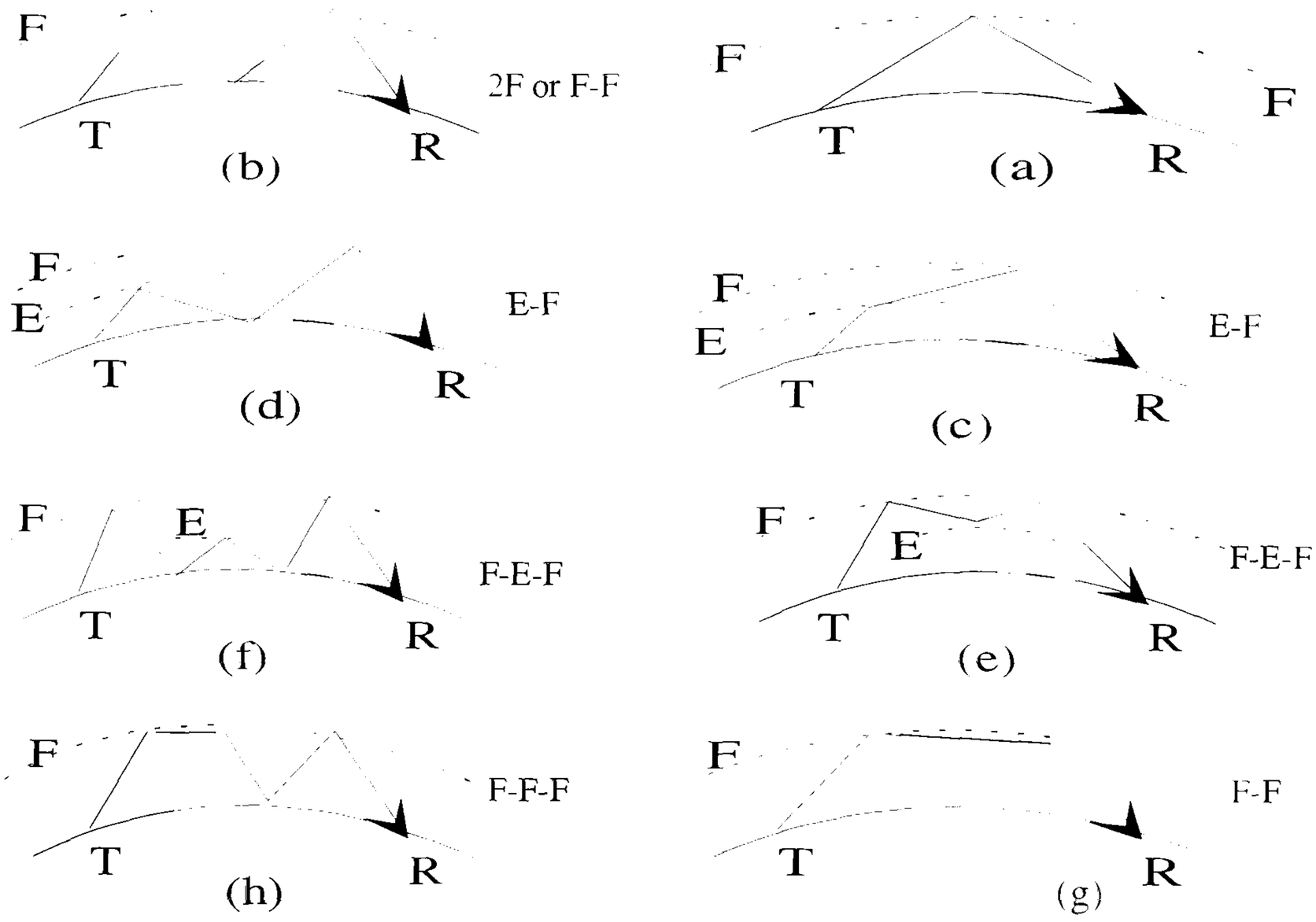


Fig. 2.4.1 : Examples of Possible Sky-wave Multihop / Multimode Pathes.

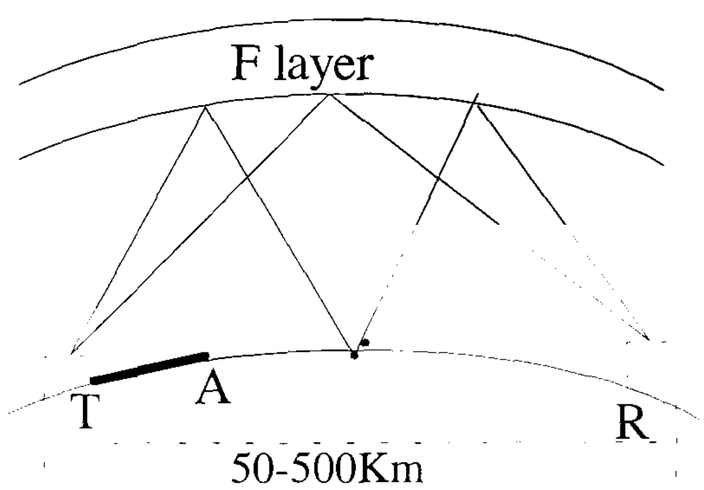


Fig. 2.4.2 : A Near-Vertical Incidence (NVI) Path Showing One and Two Hops. The letter A Shows the Maximum Effective Range of Groundwave.

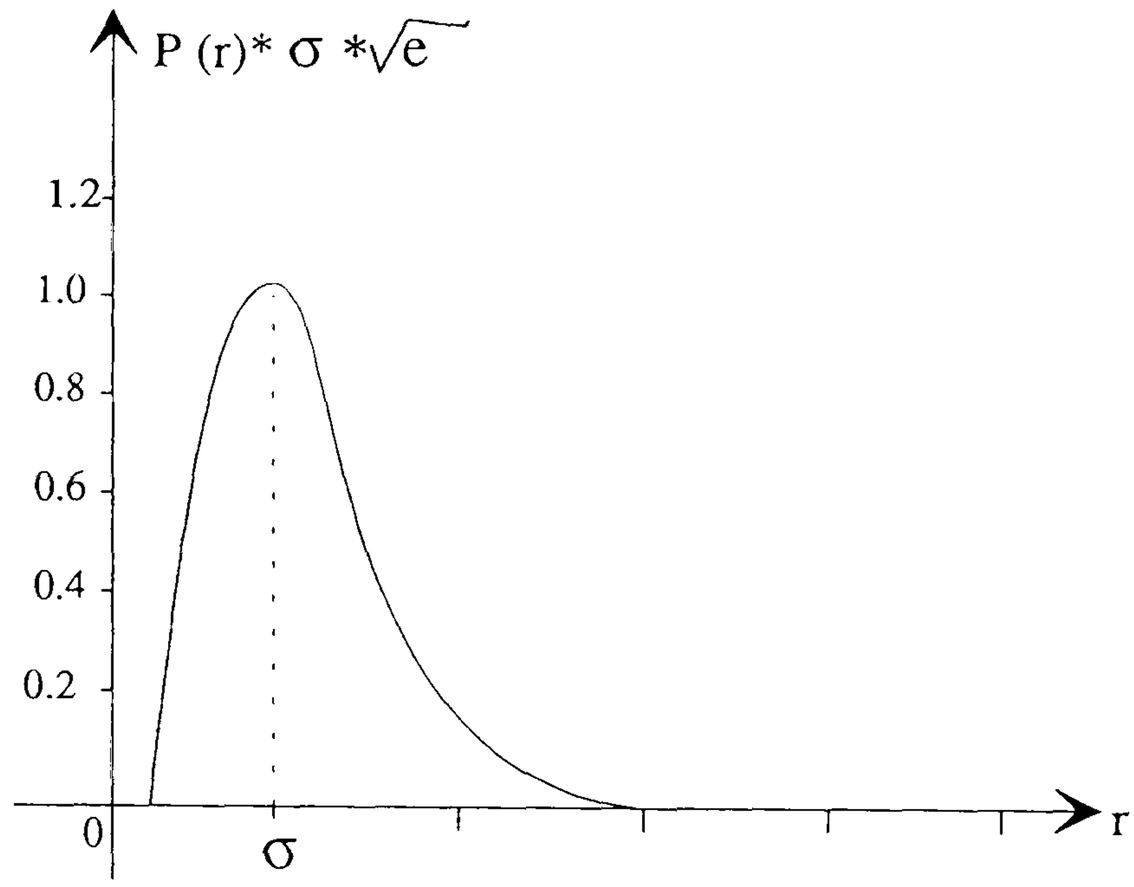


Fig. 2.6.1 : Rayleigh Probability Density Function.

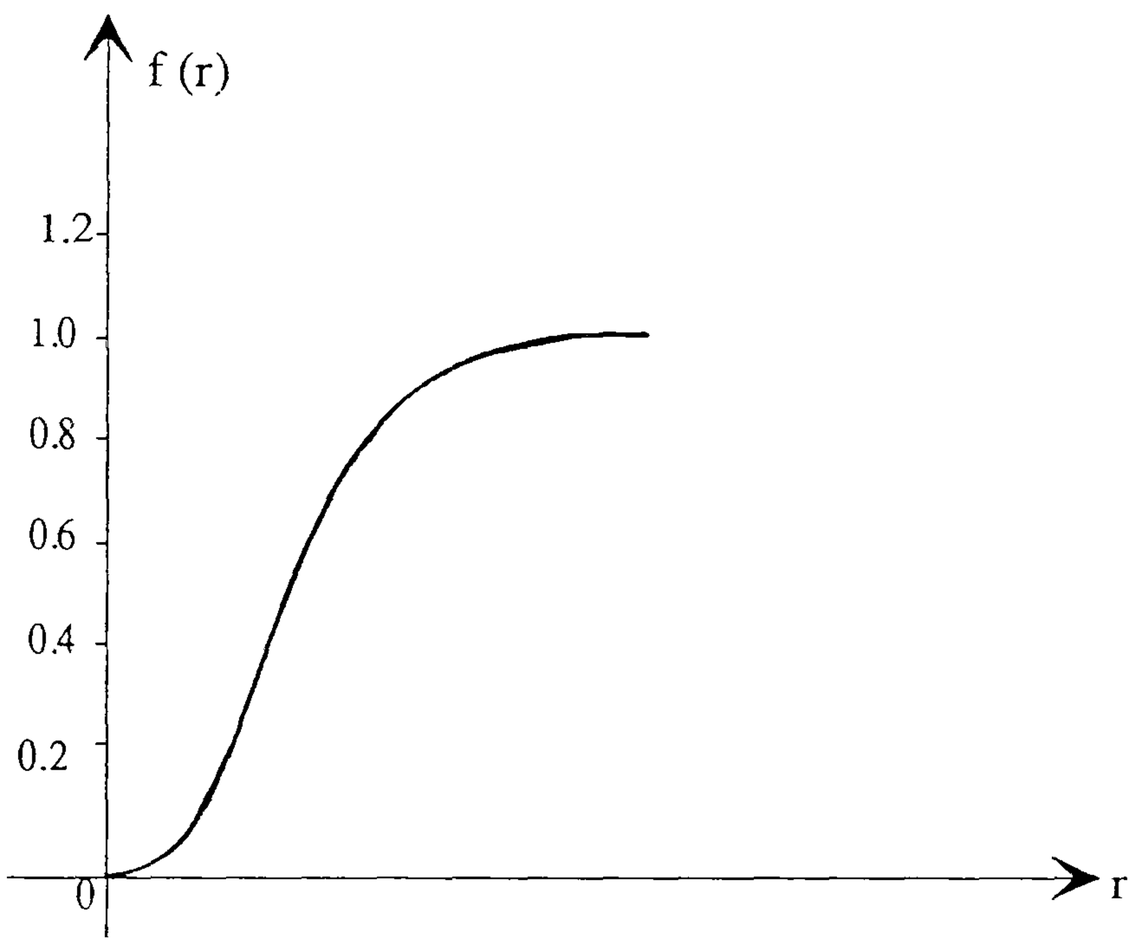
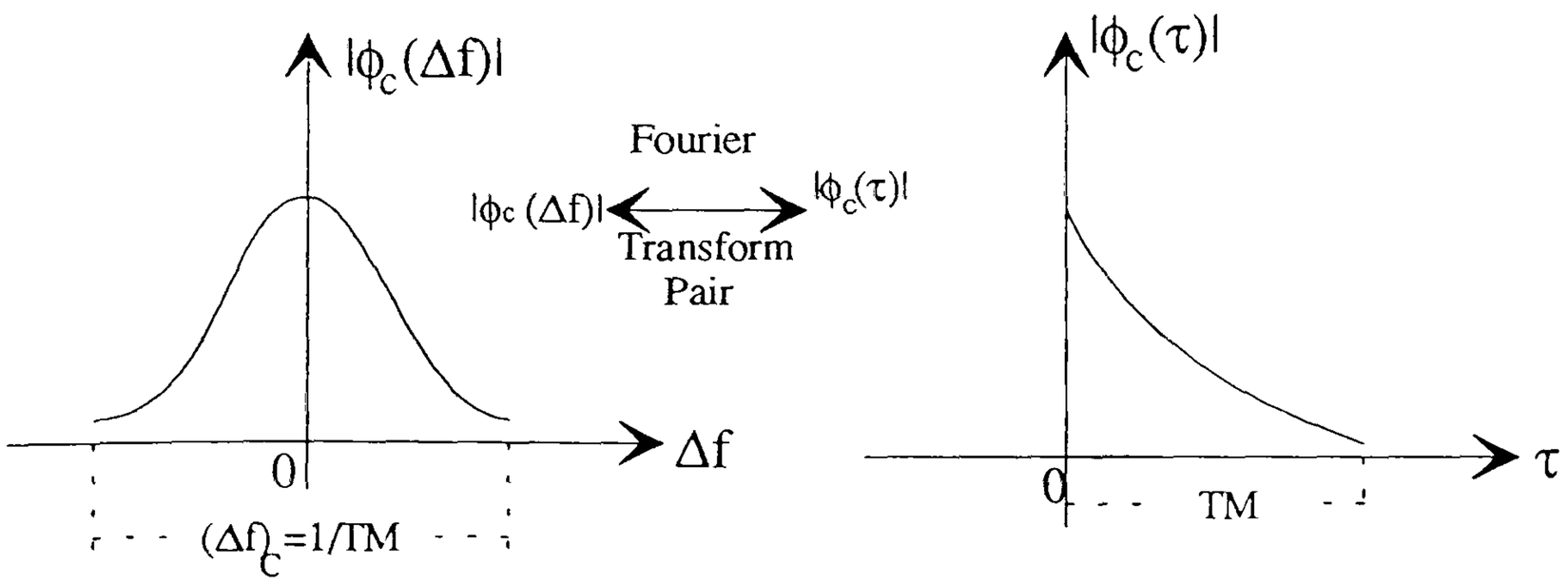
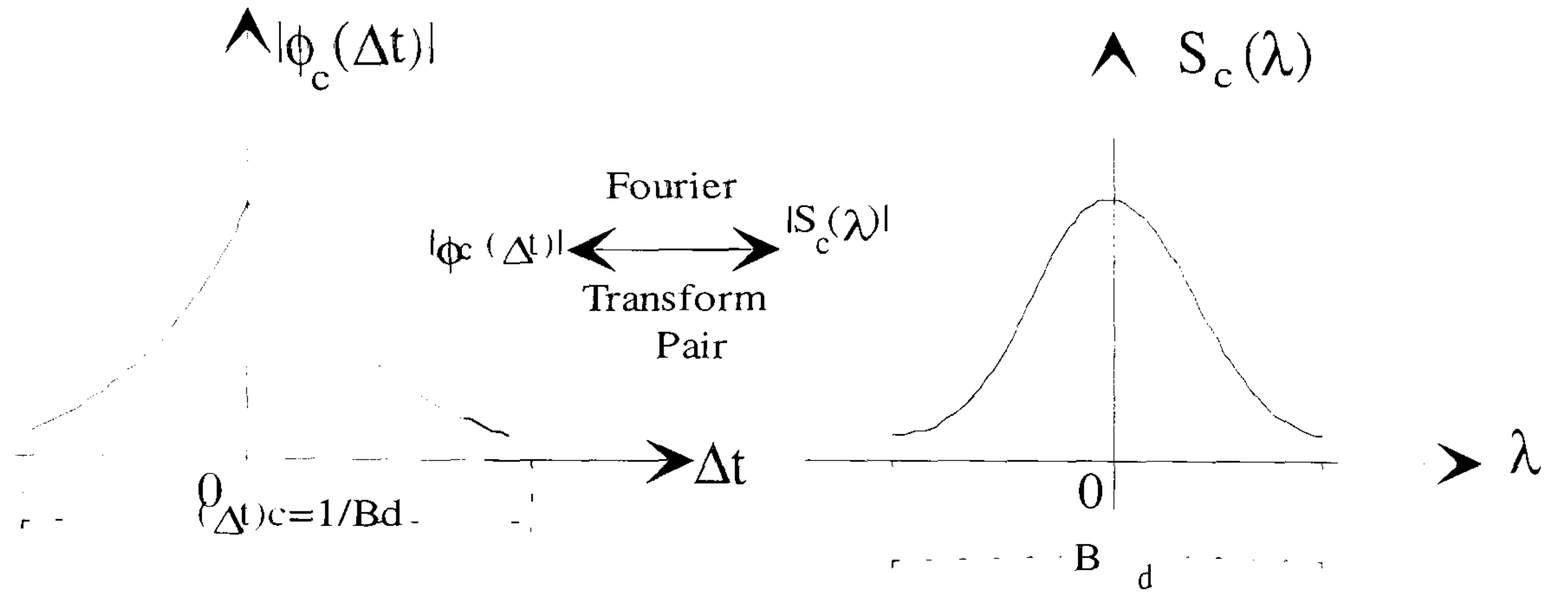


Fig. 2.6.2 : Rayleigh Comulative Distribution Function.



a- Spaced-frequency correlation function. b- Multipath intensity profile .

Fig. 2.6.3 : Relationship Between $\Phi_c(\Delta f)$ and $\phi_c(\tau)$.



a- spaced-time correlation function . b- Doppler power spectrum .

Fig. 2.6.4 : Relationship Between $\phi_c(\Delta t)$ and $S_c(\lambda)$.

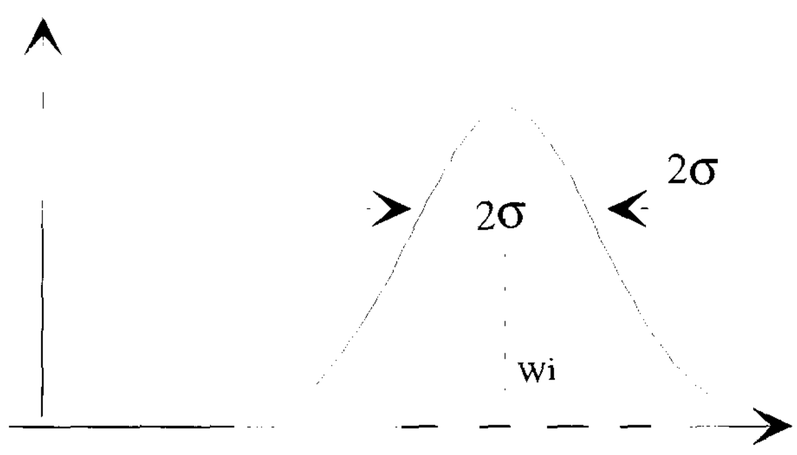


Fig. 2. 8.2 : One Gaussian Scatter Spectrum.

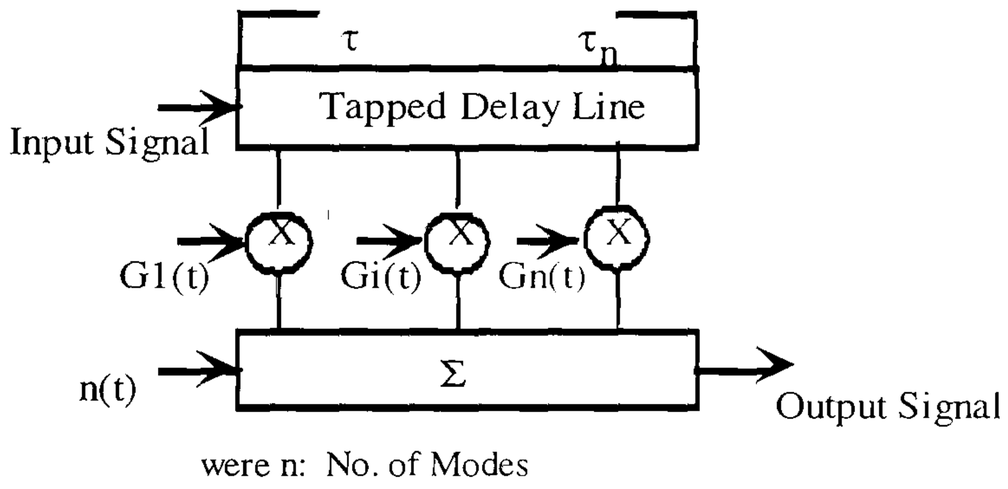


Fig. 2.8.1 : Block Diagram of HF Channel Model.

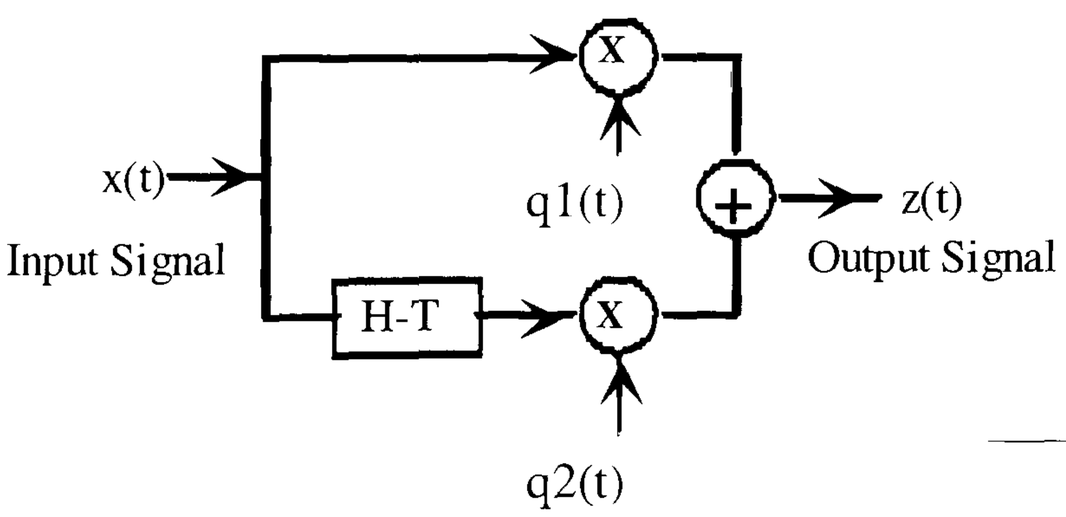


Fig.2.8.3 : Rayleigh Fading Introduced by One Sky-wave.

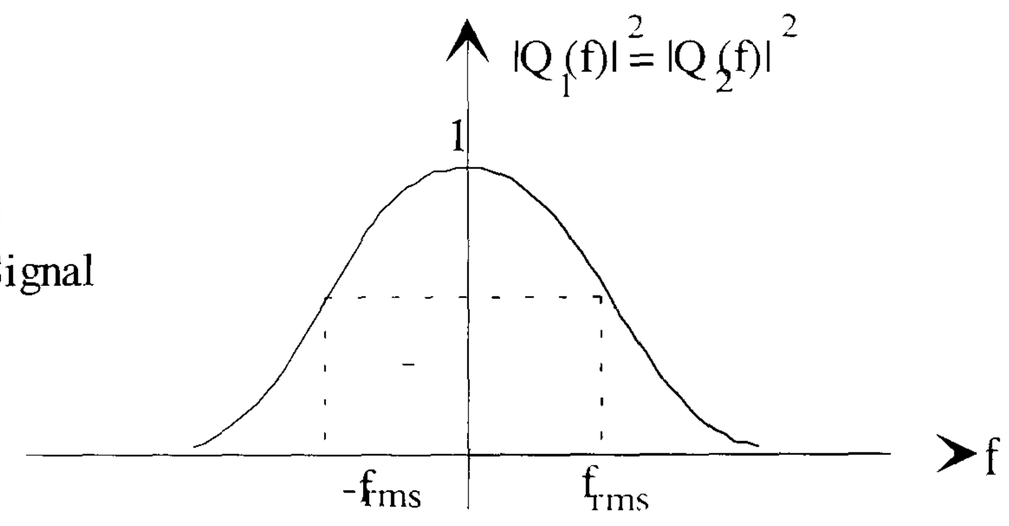


Fig. 2.8.4 : Power Spectrum of q1(t) and q2(t).

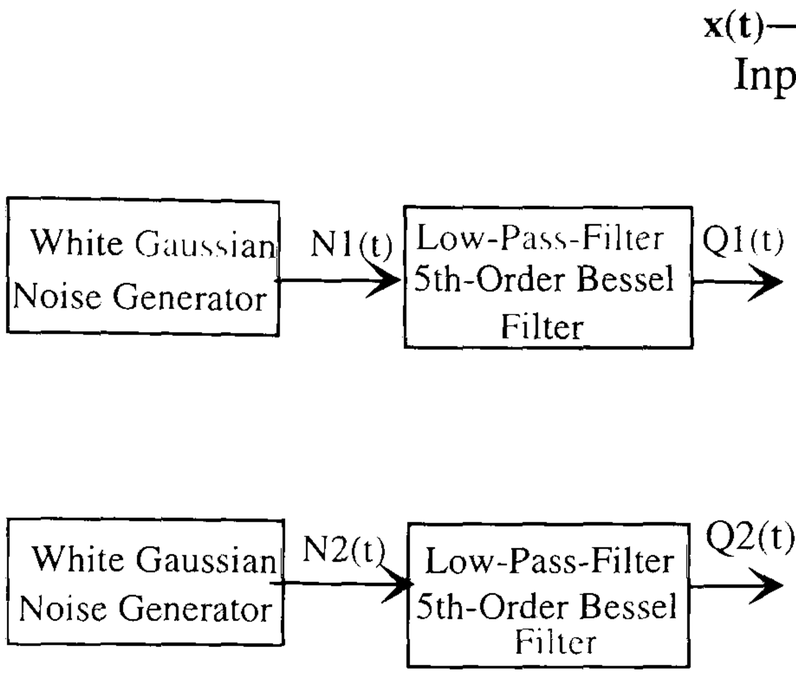


Fig. 2.8.5 : Method Of Generating Of Q1(t),Q2(t).

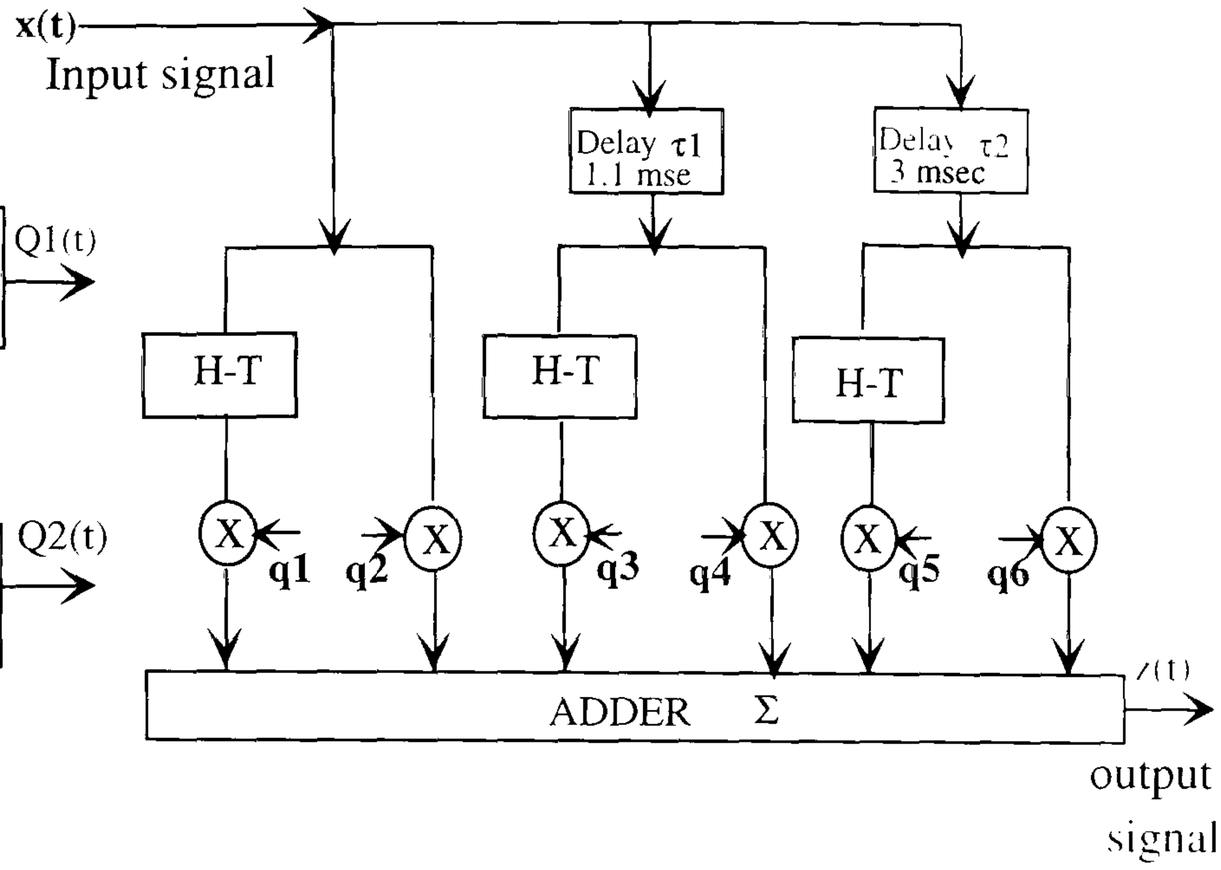


Fig.2.8.6 : HF TRANSMISSION PATH.

Channel Type	Seed Number (ID)	Mean value of q_i	Variance of q_i	Number of Fades fads/sec
Channel1 , Fsp=2Hz	75	-0.072583 -0.034652 0.073498 0.02961 -0.00679353 0.198304	0.163161471 0.165772228 0.140094846 0.125983859 0.188886771 0.216127844	35
Channel1 , Fsp=2Hz	13	-0.010012 -0.013244 0.01388 0.09614 -0.035755 -0.012377	0.15951934 0.19990402 0.166005087 0.182031123 0.133761497 0.15877883	35
Channel1 , Fsp=2Hz	177	0.0112 -0.001836 0.03474 -0.07514 0.00000865 0.03002	0.19078331 0.156384245 0.144983444 0.178360812 0.147407942 0.18208164	35

a- channel1 (3-sky waves)

Channel2 , Fsp=2Hz	57	0.090417 -0.0050908 0.00791 0.029504	0.244330539 0.231254674 0.332983251 0.191430132	35
Channel2 , Fsp=2Hz	13	-0.0119 -0.01575 0.0165 0.1143	0.225481676 0.28256568 0.23464931 0.25730222	35
Channel2 , Fsp=2Hz	177	0.0137 -0.00224 0.04243 -0.09176	0.284534921 0.233232033 0.216228838 0.266008	35

b- channel2 (2-sky waves)

Channel3 , Fsp=1Hz	57	0.122798 -0.01139 0.16093 0.04568252	0.228895682 0.21227541 0.3711399871 0.187688984	18
Channel3 , Fsp=1Hz	13	-0.01296 -0.0386 0.0478 0.1466	0.13660237 0.303061076 0.26752481 0.292811764	18
Channel3 , Fsp=1Hz	177	0.01445 -0.03237 0.06391 -0.1234	0.307395431 0.2092935 0.204333936 0.2789771	18

c- channel3 (2-sky waves)

Channel4 , Fsp=1Hz	57	0.184878 -0.017149	0.5188365478 0.48116347	18
Channel4 , Fsp=1Hz	13	0.01954 -0.05821	0.3106975 0.689303	18
Channel4 , Fsp=1Hz	177	0.0201 -0.045	0.5949335 0.405067	18

d- channel4 (1-sky wave)

Table 2.9 : Measured Characteristics of the Fading Sequences used to Model the HF Channel

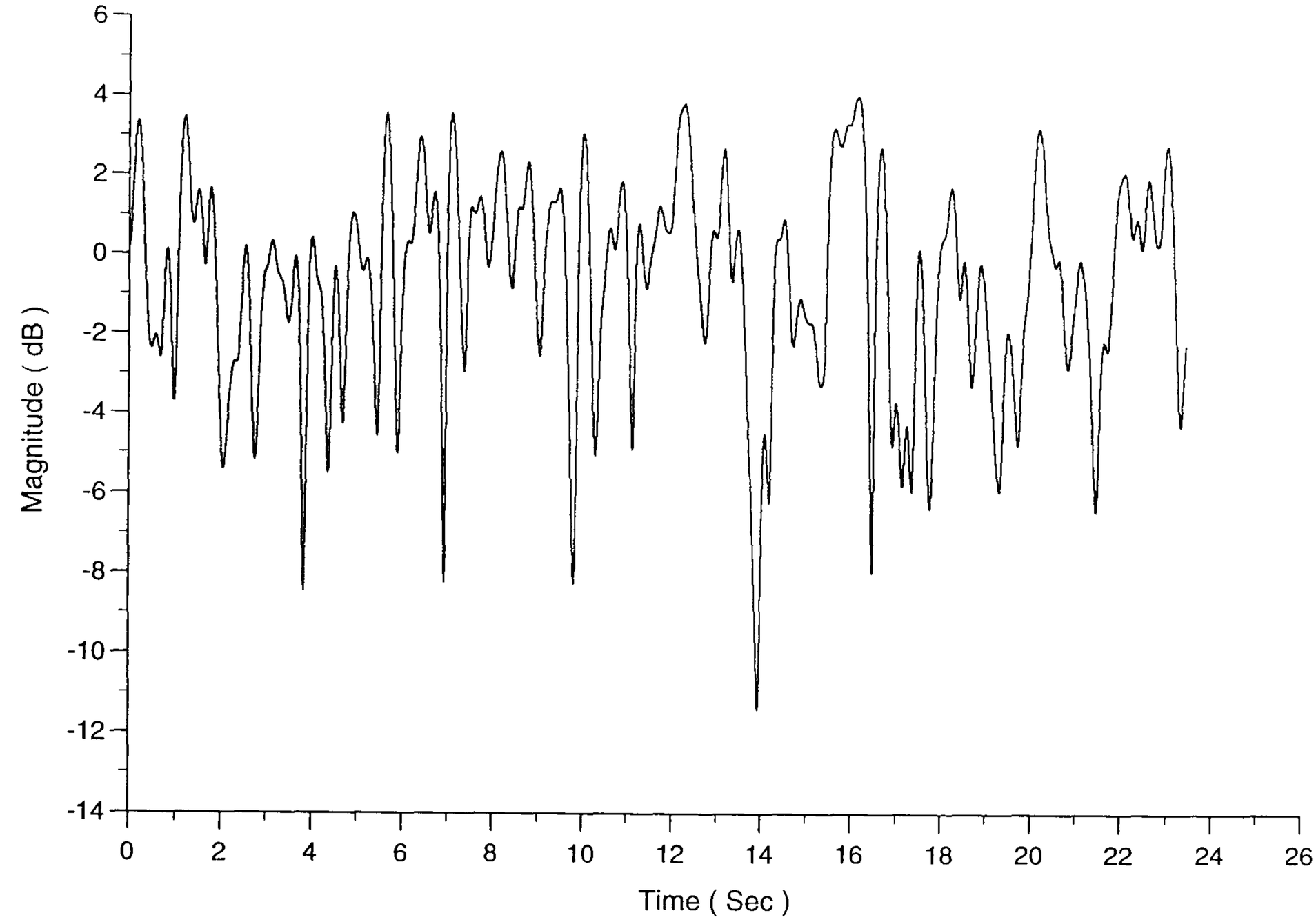


Fig. 2.9.1 Magnitude Variation of QQ Signal for Channel 1.

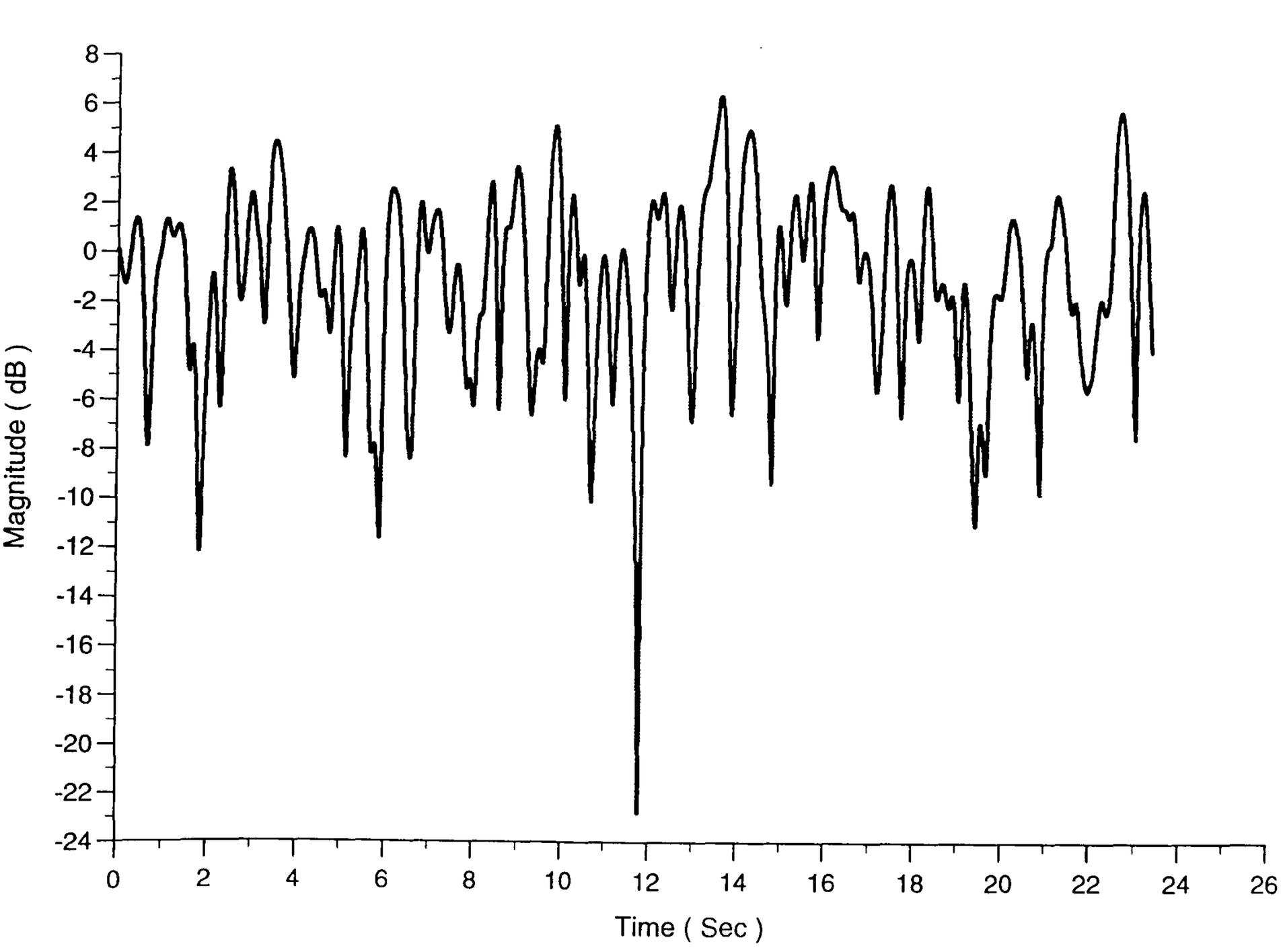


Fig. 2.9.2 Magnitude Variation of QQ Signal for Channel 2.

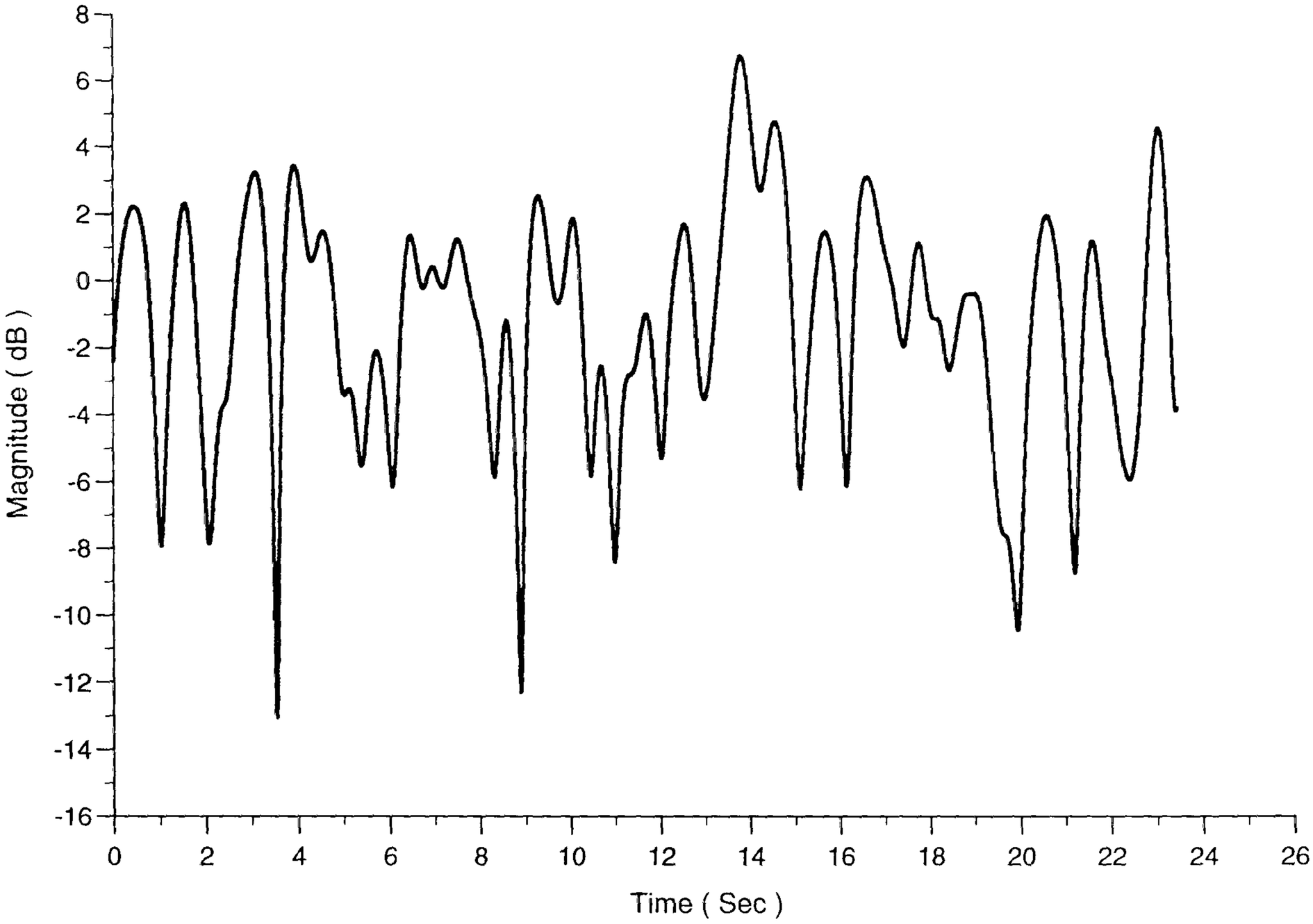


Fig. 2.9.3 Magnitude Variation of QQ Signal for Channel 3.

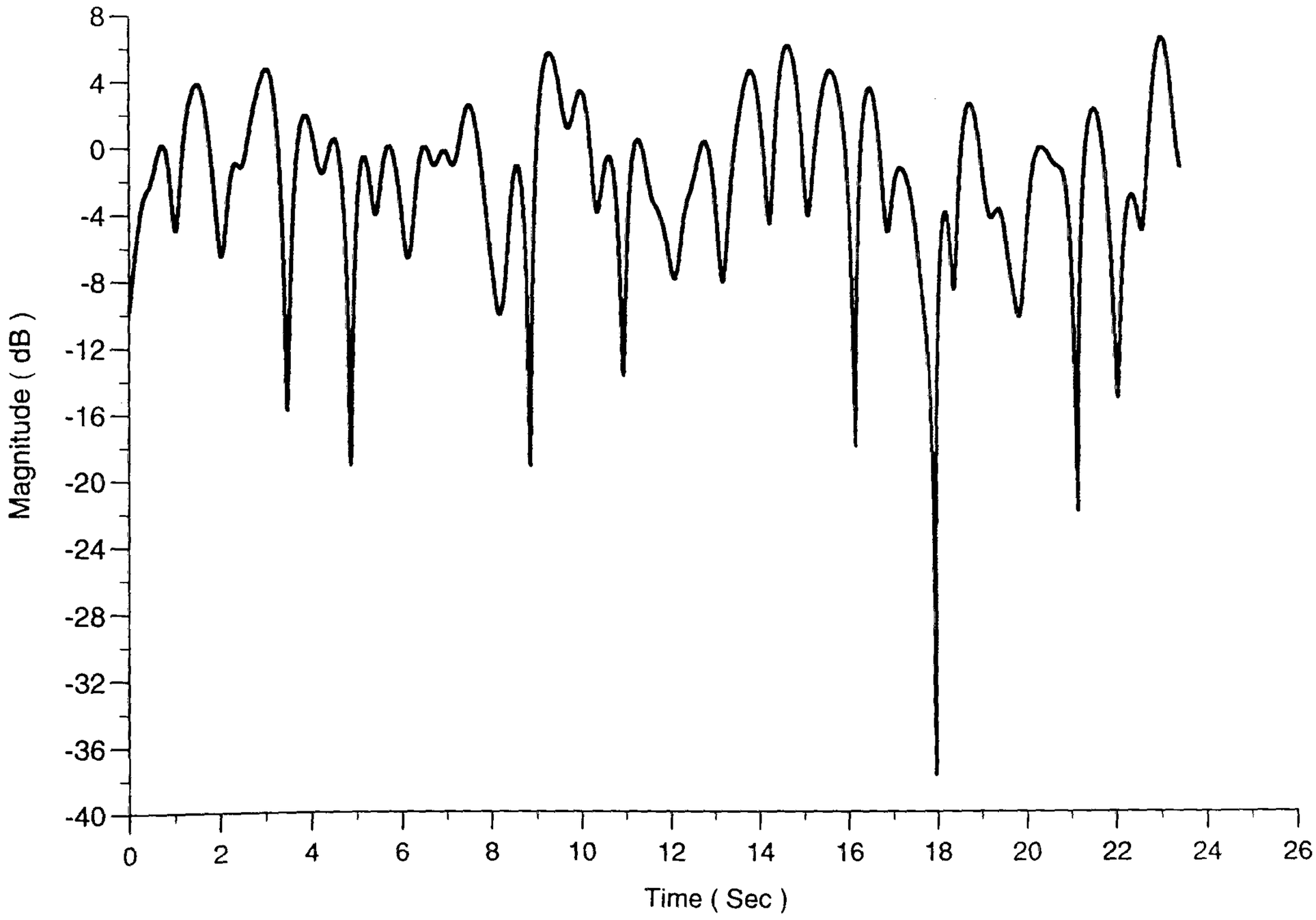


Fig. 2.9.4 Magnitude Variation of QQ Signal for Channel 4.

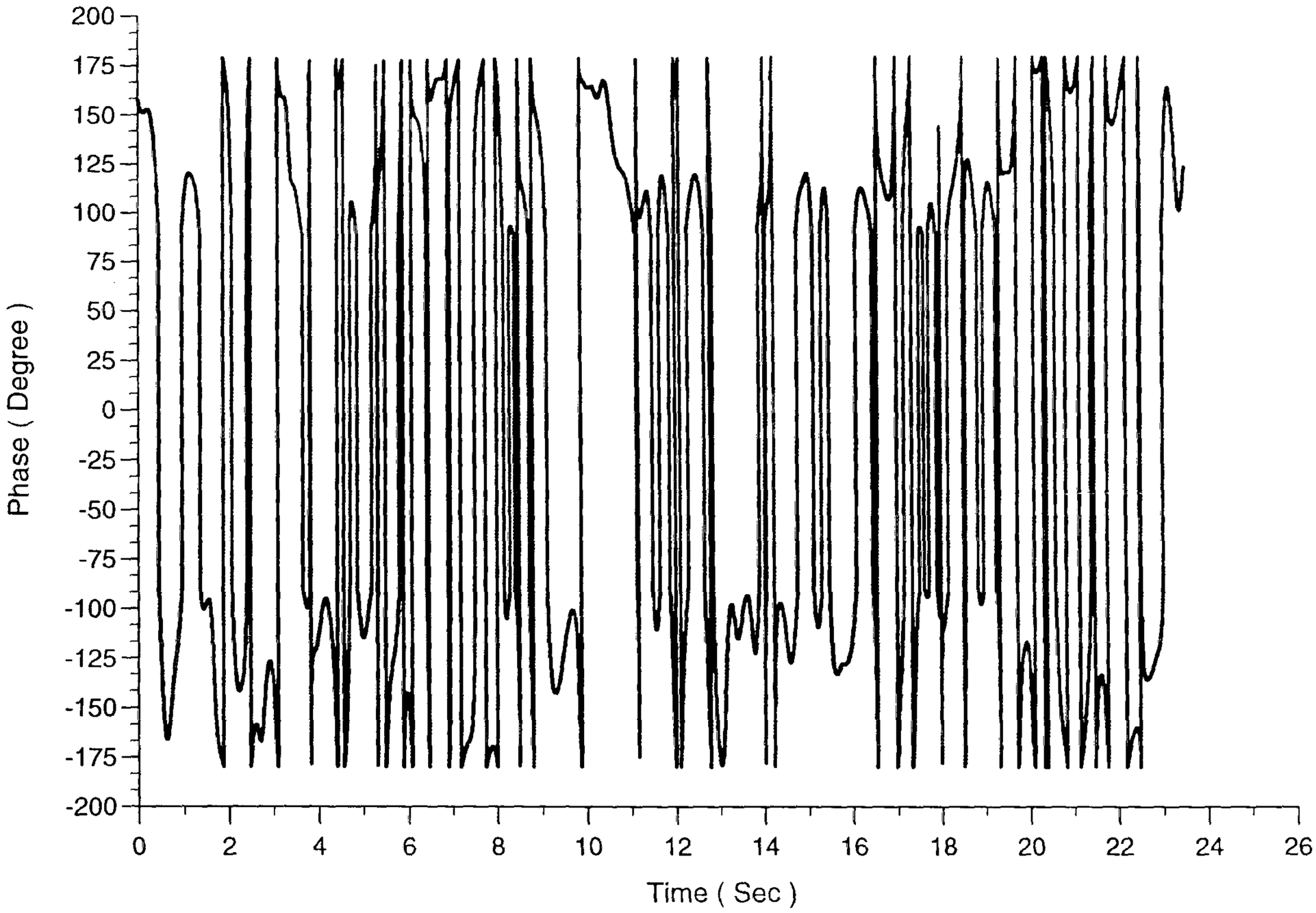


Fig. 2.9.5 Phase Variation of QQ Signal for Channel 1.

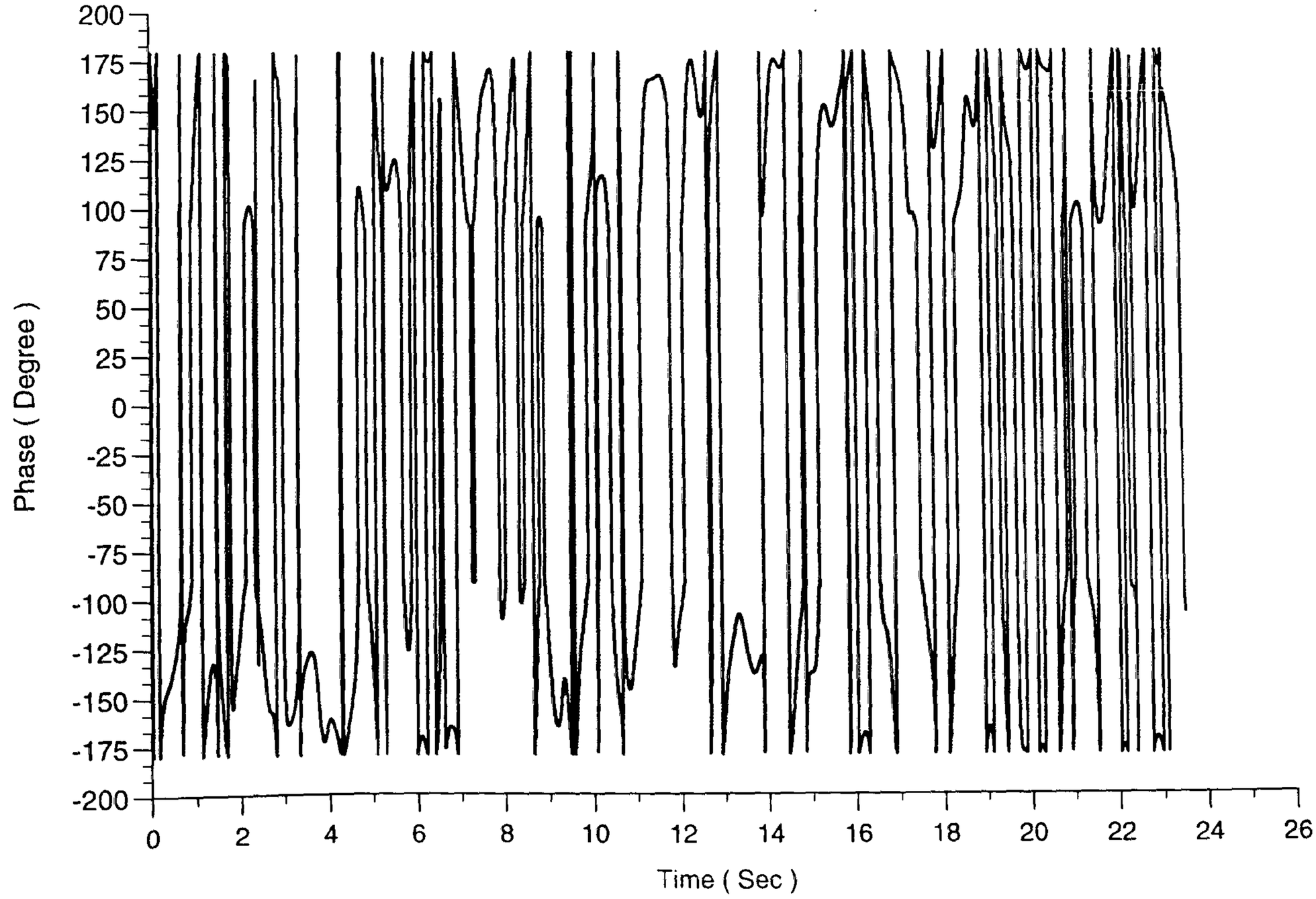


Fig. 2.9.6 Phase Variation of QQ Signal for Channel 2.

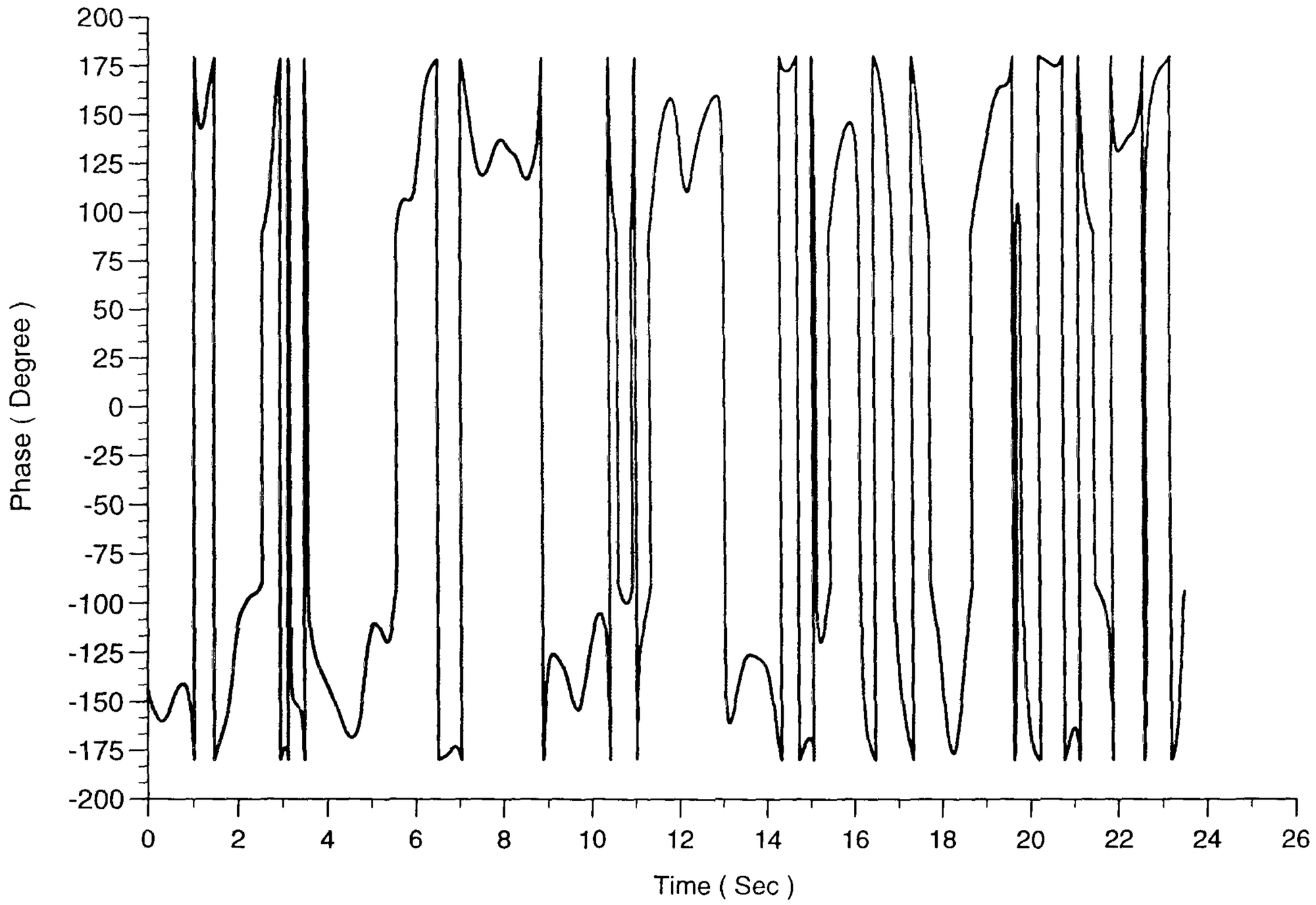


Fig. 2.9.7 Phase Variation of QQ Signal for Channel 3.

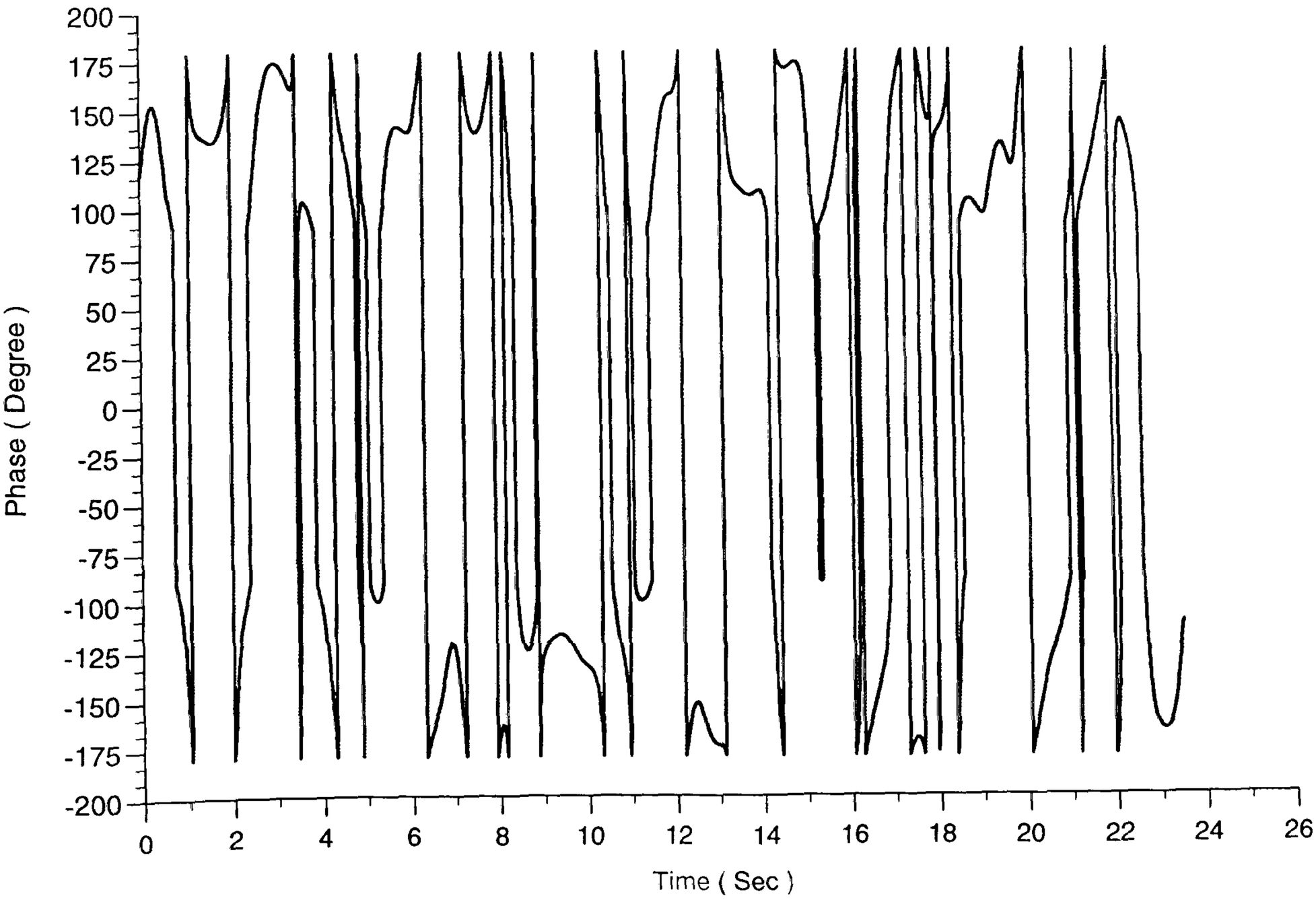


Fig. 2.9.8 Phase Variation of QQ Signal for Channel 4.

CHAPTER 3

MODEL OF DATA TRANSMISSION SYSTEM USING QUADRATURE AMPLITUDE MODULATION

3.1 INTRODUCTION

When it is required to transmit digital data over a voice frequency channel, it is necessary to modulate the incoming data onto a carrier wave, usually sinusoidal, with fixed frequency limits imposed by the channel. The channel may be a telephone channel, radio HF channel, satellite channel, or optical fibre. The digital information to be transmitted over the channel is assumed to be a sequence of binary digits occurring at a uniform rate of R bits per second. Each binary digit may be transmitted directly by sending either a wave form $s_1(t)$ corresponding to a binary digit *0* or a wave form $s_2(t)$ corresponding to a binary digit *1*. In order to increase the spectrum utilisation efficiency in a digital transmission system, the information sequence can be subdivided into groups with *k-bits* per group, there are $2^k = M$ distinct groups and hence, we require M different wave forms in order to transmit the *k-bit* groups unambiguously. Therefore multilevel modulation schemes, such 4-QAM have been developed for transmission of high-speed data over voice frequency channels. With QAM, the data signal is divided into two data signals at half the rate, each of which is transmitted as a double sideband on the same carrier frequency, but with the two modulating carriers in phase quadrature. Then, by using demodulating carriers in phase quadrature, the two data signals can be recovered separately at the receiver.

The advantages of quadrature amplitude modulation are :

- Highly bandwidth efficient modulation scheme which is also linear [20].
- A QAM signal has m levels where $m \geq 4$, and by increasing the number of levels, it is possible to increase the information transmission rate over a given bandwidth.
- No need to transmit any auxiliary pilot tone for tracking phase jitter.
- Symbol timing can be obtained directly from the line signal and therefore is independent of the carrier recovery system.
- The demodulation process is coherent and linear which greatly simplifies the theoretical analysis of the system by reducing it to that of a linear baseband channel.

These, and other factors combine to make QAM one of the most widely used modulation methods in digital transmission system over voice frequency channels.

3.2 MODEL OF QAM SYSTEM OVER HF CHANNEL

Fig.3.2.1 shows the model of a synchronous serial data transmission system. This model is widely used in digital transmission [17,26,29], and is related to the model of the QAM system explained in Appendix B. The model of QAM system over an HF channel can be described as follows :

The transmitted signal now is subjected to Rayleigh fading. From Fig.3.2.1 $\{s_i\}$ is a stream of complex data symbol values, and $\{s_i\}$ are statistically independent and equally likely to have any one of their possible values. $\sum_i s_i \cdot \delta(t - iT)$ is the input signal to the QAM transmission system, where $1/T$ is the signal element rate in baud (transmission baud rate). Filtering of the input data stream is necessary in order to band limit the resulting QAM signal to the typical *3KHz* bandwidth available on an HF channel. This is done by the shaping low-pass filter (A'), which has the real-valued impulse response *a'(t)* with transfer function *A'(f)* such that :

$$A'(f) = 0 \quad for \quad f_c - k f_{sp} < f < -f_c + k f_{sp} \qquad 3.2.1$$

where f_{sp} is the largest value of the frequency spread expected to be introduced by the HF channel into the modulated QAM signal, f_c is the carrier frequency in *Hz* and *k* is any positive integer [17]. An example of such a low-pass filter (A') is shown in Fig.3.2.2, which illustrates the amplitude spectrum.

The linear modulator converts the input data stream $\{s_i\}$ into a serial stream of four level QAM signal elements with a carrier frequency ($f_c = 1800 Hz$) and an element rate of *2400* baud. The voice band QAM signal is fed to the HF channel, where its spectrum is shifted into the HF band by a linear process of single sideband suppressed carrier amplitude modulation. A corresponding demodulator returns the spectrum of the received signal back to the voice band. The signal at the output of the lowpass filter is given by :

$$\sum_i s_i \cdot a'(t - iT) = \sum_i s_{1,i} \cdot a'(t - iT) + j \sum_i s_{2,i} \cdot a'(t - iT) \qquad 3.2.2$$

The signal at the output of multiplier is given by :

$$m_1(t) = \sqrt{2} \sum_i s_i \cdot a'(t - iT) e^{j2\pi f_c t} = \sqrt{2} \sum_i s_i \cdot a'(t - iT)[\cos 2\pi f_c t + j \sin 2\pi f_c t]$$

$$= \sqrt{2} \{[\sum_i s_{1,i} \cdot a'(t - iT) + j \sum_i s_{2,i} \cdot a'(t - iT)][\cos 2\pi f_c t + j \sin 2\pi f_c t]\} \qquad 3.2.3$$

By taking the real-part of eqn.3.2.3 the QAM signal $m_2(t)$ can be obtained as :

$$m_2(t)=\Re\{m_1(t)\}=\sqrt{2}\{\sum_i s_{1,i}\cdot a'(t-iT)\cos 2\pi f_c t-\sum_i s_{2,i}\cdot a'(t-iT)\sin 2\pi f_c t\} \quad 3.2.4$$

The derivation of eqn.3.2.4 is given in Appendix-B. The amplitude spectrum of $m_2(t)$ is $|M_2(f)|$ and is shown in Fig.3.2.3. Now compare eqn.3.2.4 with eqn.B-7 from Appendix-B, the QAM signal $m_2(t)$ can be written as :

$$m_2(t)=\frac{1}{\sqrt{2}}\{\sum_i s_i\cdot a'(t-iT)e^{j2\pi f_c t}+s_i^*\cdot a'(t-iT)e^{-j2\pi f_c t}\} \quad 3.2.5$$

where s_i^* is the complex conjugate of s_i .

The QAM signal $m_2(t)$ is transmitted via the HF radio link, whose model is as explained in Chapter 2 and as shown in Fig.3.2.4. For theoretical analysis the two-sky wave model was chosen, for this study. The model is based on the CCIR recommendations. This signal is fed first to the transmitter filter of the radio equipment which has the impulse response and transfer function *d(t), D(f)* respectively. The output of filter D is given by :

$$m(t)=m_2(t)*d(t) \quad 3.2.6$$

The output of the transmission path is given by :

$$z(t)=m(t)*h(t)=m_2(t)*h(t)*d(t) \quad 3.2.7$$

z(t) is fed through the radio receiver filter which has the impulse response and transfer function *g(t), G(f)* respectively.

The transmission path, filter D and filter G represent the voice band HF channel which introduces distortion into the QAM signal while the linear modulator and demodulator do not introduce any distortion. From eqn.3.2.5 and 3.2.6, then :

$$m(t)=\frac{1}{\sqrt{2}}\sum_i\{s_i\cdot a'(t-iT)e^{j2\pi f_c t}+s_i^*\cdot a'(t-iT)e^{-j2\pi f_c t}\}*d(t) \quad 3.2.8$$

$$=\frac{1}{\sqrt{2}}\sum_i\{s_i\cdot a'(t-iT)e^{j2\pi f_c t}*d(t)+s_i^*\cdot a'(t-iT)e^{-j2\pi f_c t}*d(t)\}$$

Using eqn.B-23 from Appendix-B, then eqn.3.2.8 becomes :

$$m(t)=\frac{1}{\sqrt{2}}\sum_i\{[s_i\cdot a'(t-iT)*d(t)e^{-j2\pi f_c t}]e^{j2\pi f_c t} \quad 3.2.9$$

$$+[s_i^*\cdot a'(t-iT)*d(t)e^{j2\pi f_c t}]e^{-j2\pi f_c t}\}$$

Let :

$$a(t-iT) = a'(t-iT) * d(t) e^{-j2\pi f_c t} \qquad 3.2.10$$

Then eqn.3.2.8 becomes :

$$m(t) = \frac{1}{\sqrt{2}} \sum_i \{[s_i \cdot a(t-iT) e^{j2\pi f_c t} + s_i^* \cdot a^*(t-iT) e^{-j2\pi f_c t}\} \qquad 3.2.11$$

where *a(t-iT)* represents the overall filtering carried out at the transmitter on the baseband signal. The signal *m(t)* is transmitted through a two-sky wave HF channel shown in Fig.3.2.4, where the delay in transmission over each path is constant and the relative delay for two paths is τ seconds. The output of this model is given by :

$$z(t) = m(t) \cdot q_1(t) + \hat{m}(t) q_2(t) + m(t-\tau) \cdot q_3(t) + \hat{m}(t-\tau) q_4(t) \qquad 3.2.12$$

Where $\hat{m}(t)$ is the Hilbert transform of *m(t)* and the Hilbert filter is an all pass filter which introduces a $(-90^o/90^o)$ phase shift to all positive/negative frequencies of its input signal [1]. The output of Hilbert transformer is given by :

$$\hat{m}(t) = m(t) * f(t) \qquad 3.2.13$$

Where *f(t)* is the impulse response of Hilbert transform, given by :

$$f(t) = 1/(\pi \cdot t) \qquad 3.2.14$$

With the transfer function given by :

$$\begin{aligned} F(f) &= -j \quad for \quad f > 0 \\ &+ j \quad for \quad f < 0 \end{aligned} \qquad 3.2.15$$

From eqn.3.2.11 and eqn.3.2.13 we get :

$$\hat{m}(t) = \frac{1}{\sqrt{2}} \{\sum_i \; s_i \cdot a(t-iT) e^{j2\pi f_c t} * f(t) + \sum_i \; s_i^* \cdot a^*(t-iT) e^{-j2\pi f_c t} * f(t)\} \qquad 3.2.16$$

Using the eqn.B-23 from Appendix-B, eqn.3.2.16 becomes :

$$\begin{aligned} \hat{m}(t) = \frac{1}{\sqrt{2}} \{&\sum_i [s_i \cdot a(t-iT) * f(t) e^{-j2\pi f_c t}] e^{j2\pi f_c t} \\ &+ [\sum_i \; s_i^* \cdot a^*(t-iT) * f(t) e^{j2\pi f_c t}] e^{-j2\pi f_c t}\} \end{aligned} \qquad 3.2.17$$

Now, according to eqn.3.2.10 and 3.2.1, the Fourier transform of *a(t)* is band-limited to the frequency band of *A'(f)*, i.e. $(-f_c$ to $f_c)$. The Fourier transforms of

$f(t)e^{-j2\pi f_c t}$ & $f(t)e^{j2\pi f_c t}$ are $F(f+f_c)$ & $F(f-f_c)$ respectively which takes the value $-j$ & $+j$ over the same frequency band. Consequently, by taking the Fourier transform of $\hat{m}(t)$ in eqn.3.2.17, and substituting the values $(-j,+j)$ for $F(f+f_c)$ & $F(f-f_c)$ over frequency band of $A(f)$, $(-f_c, f_c)$, and then taking the inverse Fourier transform, then eqn.3.2.17 can be written as :

$$\hat{m}(t)=\frac{1}{\sqrt{2}}\{\sum_i s_i\cdot a(t-iT)(-j)e^{j2\pi f_c t}+\sum_i s_i^*\cdot a^*(t-iT)(j)e^{-j2\pi f_c t}\}$$

Or,

$$\hat{m}(t)=\frac{1}{\sqrt{2}}\{\sum_i -j\,s_i\cdot a(t-iT)e^{j2\pi f_c t}+\sum_i j\,s_i^*\cdot a^*(t-iT)e^{-j2\pi f_c t}\} \quad 3.2.18$$

Considering two-sky wave HF channel and from eqn.3.2.12 & 3.2.18 yields :

$$\begin{aligned} z(t)=\frac{1}{\sqrt{2}}\sum_i\{&[s_i\cdot a(t-iT)e^{j2\pi f_c t}+s_i^*\cdot a^*(t-iT)e^{-j2\pi f_c t}]\cdot q_1(t)\\ &+q_2(t)[-j\,s_i\cdot a(t-iT)e^{j2\pi f_c t}+j\,s_i^*\cdot a^*(t-iT)e^{-j2\pi f_c t}]\\ &+q_3(t)[s_i\cdot a(t-\tau-iT)e^{j2\pi f_c(t-\tau)}+s_i^*\cdot a^*(t-\tau-iT)e^{-j2\pi f_c(t-\tau)}]\\ &+q_4(t)[-j\,s_i\cdot a(t-\tau-iT)e^{j2\pi f_c(t-\tau)}+s_i^*\cdot a^*(t-\tau-iT)e^{-j2\pi f_c(t-\tau)}]\} \end{aligned}$$

$\Rightarrow$

$$\begin{aligned} z(t)=\frac{1}{\sqrt{2}}\sum_i\{&s_i\cdot a(t-iT)[q_1(t)-jq_2(t)]e^{j2\pi f_c t}\\ &+s_i^*\cdot a^*(t-iT)[q_1(t)+jq_2(t)]e^{-j2\pi f_c t}\\ &+s_i\cdot a(t-\tau-iT)[q_3(t)-jq_4(t)]e^{j2\pi f_c(t-\tau)}\\ &+s_i^*\cdot a^*(t-\tau-iT)[q_3(t)+jq_4(t)]e^{-j2\pi f_c(t-\tau)}\} \end{aligned} \quad 3.2.19$$

Let :

$$h_i(t-iT)=a(t-iT)[q_1(t)-jq_2(t)]+a(t-\tau-iT)[q_3(t)-jq_4(t)]\cdot e^{-j2\pi f_c\tau} \quad 3.2.20$$

Then the complex-conjugate of $h_i(t-iT)$ is :

$$h_i^*(t-iT)=a^*(t-iT)[q_1(t)+jq_2(t)]+a^*(t-\tau-iT)[q_3(t)+jq_4(t)]\cdot e^{j2\pi f_c\tau} \quad 3.2.21$$

Substituting eqn.3.2.20 , 3.2.21 into 3.2.19, then :

$$z(t)=\frac{1}{\sqrt{2}}\sum_i\{s_i\cdot h_i(t-iT)e^{j2\pi f_c t}+s_i^*\cdot h_i^*(t-iT)e^{-j2\pi f_c t}\} \quad 3.2.22$$

If τ is assumed constant then ($e^{-j2\pi f_c\tau}$) is a fixed complex-valued scalar quantity with absolute value of unity, therefore it has no affect on the statistical properties of

$[(q_3(t)-jq_4(t))e^{-j2\pi f_c\tau}]$, bearing in mind that $q_3(t)$ & $q_4(t)$ are statistically independent with zero mean Gaussian processes. Then $h_i(t-iT)$ can be written as :

$$h_i(t-iT)=a(t-iT)[q_1(t)-jq_2(t)]+a(t-\tau-iT)[q_3(t)-jq_4(t)] \qquad 3.2.23$$

The signal at the input of filter C is given by :

$$r_1(t)=z(t)*g(t)+n(t) \qquad 3.2.24$$

Where $n(t)$ is the only additive noise and is assumed here to be white Gaussian noise with zero mean and two-sided power spectral density of $N_0/2$. The band-pass filter (Filter C) removes the noise frequencies outside the data signal band without excessively distorting the wanted signal. The signal at the output of receiver filter C is given by :

$$r_2(t)=r_1(t)*c(t)=z(t)*g(t)*c(t)+n(t)*c(t) \qquad 3.2.25$$

If the signal $r_2(t)$ is coherently demodulated assuming that the receiver is synchronised with the transmitter, and by neglecting any phase difference between the reference carrier and the received signal, i.e. $f_c=f'_c$, then the signal at the output of receiver lowpass filter (Filter B') is :

$$r(t)=\sqrt{2}\cdot r_2(t)\cdot e^{-j2\pi f'_c t}*b'(t) \qquad 3.2.26$$

From eqn.3.2.25, then :

$$r(t)=\sqrt{2}\{[z(t)*g(t)*c(t)]\cdot e^{-j2\pi f'_c t}+[n(t)*c(t)]e^{-j2\pi f'_c t}\}*b'(t)$$

Using the eqn.B-23 from Appendix-B, then r(t) can be written as :

$$r(t)=\sqrt{2}\{z(t)\cdot e^{-j2\pi f'_c t}*[g(t)*c(t)]\cdot e^{-j2\pi f'_c t}\}*b'(t) \qquad 3.2.27$$
$$+\sqrt{2}\{[n(t)*c(t)]e^{-j2\pi f'_c t}\}*b'(t)$$

Let :

$$b(t)=\{[g(t)*c(t)]\cdot e^{-j2\pi f'_c t}\}*b'(t) \qquad 3.2.28$$

and :

$$f'_c=f_c \qquad 3.2.29$$

Then :

$$r(t)=\sqrt{2}\{z(t)\cdot e^{-j2\pi f_c t}\}*b(t)+v(t) \qquad 3.2.30$$

where $b(t)$ represents the overall filtering carried out on the signal at the receiver, and :

$$v(t) = \sqrt{2}\{[n(t) * c(t)]e^{-j2\pi f_c t}\} * b'(t) \qquad 3.2.31$$

$v(t)$ is the resultant noise component in the receiver.

Now replacing the value of $z(t)$ from eqn.3.2.22 into eqn.3.2.30, then :

$$r(t) = \sum_i \{ s_i \cdot h_i(t - iT) + s_i^* \cdot h_i^*(t - iT)e^{-j4\pi f_c t}\} * b(t) + v(t) \qquad 3.2.32$$

Now consider the eqn.3.2.20 :

$$h_i(t - iT) = a(t - iT)[q_1(t) - j\,q_2(t)] + a(t - \tau - iT)[q_3(t) - j\,q_4(t)]$$

which consists of random components $q_i(t)$ and time-invariant impulse response $a(t)$. The frequency response of $h_i(t - iT)$ could be considered band limited and is dependent on A(f) and the frequency spreads introduced by the $q_i(t)$. The frequency spread is given by ($f_{sp} = 2 f_{rms}$), where f_{rms} is rms frequency in the order of a few Hz and expands, according to (eqn. $|Q_1(f)|^2 = e^{-f^2/2f_{rms}^2}$ Chapter 2), over all frequencies. If now, $A'(f)$ given by eqn.3.2.1 (assuming the value of k in eqn.3.2.1 is equal to (5) [17]) could be made such that after accommodating the highest frequency spread, it still lies within the frequency band ($-f_c$ *to* f_c), then the frequency response of $h_i(t - iT)$ could be limited to this band, such that :

$$|H(f)| = 0 \qquad |f| > f_c \qquad 3.2.33$$

Therefore, the Fourier transform of $h_i^*(t - iT)e^{-j4\pi f_c t}$ in eqn.3.2.32 will be outside the passband of filter B' according to eqn.3.2.1, assuming the filter B' at the receiver has the same characteristic as the transmitter filter A'. From previous discussion eqn.3.2.32 can be written as

$$r(t) = \sum_i s_i \cdot h_i(t - iT) * b(t) + v(t) \qquad 3.2.34$$

Or,

$$r(t) = \sum_i s_i \cdot p_i(t - iT) + v(t) \qquad 3.2.35$$

where $p_i(t - iT)$ here, is the time-varying impulse response of the linear base band channel, and is a cascade of the transmitter filter, receiver filters and voice band RF channel. $p_i(t - iT)$ is given by :

$$p_i(t-iT) = h_i(t-iT)*b(t) \qquad 3.2.36$$

or from eqn.3.2.20, then :

$$\begin{aligned} p_i(t-iT) = \{&a(t-iT)[q_1(t)-jq_2(t)]+ \\ &+a(t-\tau-iT)[q_3(t)-jq_4(t)]\}*b(t) \end{aligned} \qquad 3.2.37$$

By similarity it can be shown that $p_i(t-iT)$ for a three sky wave channel is :

$$\begin{aligned} p_i(t-iT) = \{&a(t-iT)[q_1(t)-jq_2(t)]+ \\ &+a(t-\tau_1-iT)[q_3(t)-jq_4(t)]+ \\ &+a(t-\tau_2-iT)[q_5(t)-jq_6(t)]\}*b(t) \end{aligned} \qquad 3.3.38$$

where τ_1 & τ_2 are the difference between the arrival times at the receiver. From eqn.3.2.33 and 3.2.37 an alternative baseband model of the QAM system can be represented as shown in Fig.3.2.5. It is now important to consider the statistics of transmitted signal and the complex-valued gaussian noise wave-form $v(t)$. First the average transmitted energy per signal element at the output of Filter A in Fig.3.2.5 is given by :

$$E_s = E\{\int_{-\infty}^{\infty} |s_i \cdot a(t-iT)|^2 \cdot d(t)\} \qquad 3.2.39$$

where $E[\cdot]$ denotes the expected value of the quantity within the square brackets. Using Parseval's theorem then :

$$E_s = \overline{s_i}^2 \cdot \int_{-\infty}^{\infty} |A(f)|^2 \cdot d(f) \qquad 3.2.40$$

where $\overline{s_i}^2 = E[|s_i|^2]$. The average transmitted energy per bit (E_B), is given by :

$$E_B = \frac{E_s}{\log_2 m} = \frac{\overline{s_i}^2}{\log_2 m} \cdot \int_{-\infty}^{\infty} |A(f)|^2 \cdot d(f) = \frac{\overline{s_i}^2}{2} \cdot \int_{-\infty}^{\infty} |A(f)|^2 \cdot d(f) \qquad 3.2.41$$

In eqn.3.2.41 m is equal to 4 for this case (QPSK). The average energy at the input of receiver filter B (Fig.3.2.5) is :

$$E_R = E\{\int_{-\infty}^{\infty}(s_i \cdot a(t-iT)[q_1(t)-jq_2(t)]+s_i \cdot a(t-\tau-iT)[q_3(t)-jq_4(t)])^2 \cdot dt\}$$

$$\Rightarrow E_R = \overline{s_i^2} \cdot [\overline{q_1(t)}^2 + \overline{q_2(t)}^2 + \overline{q_3(t)}^2 + \overline{q_4(t)}^2] \cdot \int_{-\infty}^{\infty} |A(f)|^2 \cdot df \qquad 3.2.42$$

Assuming $q_i(t)$ are independent, and the quantities $\overline{q_1}^2$, $\overline{q_2}^2$, $\overline{q_3}^2$, $\overline{q_4}^2$ are the variances of q_1, q_2, q_3, q_4 respectively. But from Chapter 2, the four variances are equal with the value of each is (1/4). In other words, the two sky waves channel does not introduce on average any gain or attenuation into the transmitted signal.
From eqn. 3.2.40 and 3.2.42, yields :

$$E_R = [\overline{q_1(t)}^2 + \overline{q_2(t)}^2 + \overline{q_3(t)}^2 + \overline{q_4(t)}^2] \cdot E_s \qquad 3.2.43$$

From eqn.3.2.43, and if the sum of the variances of the $q_i(t)$ is equal to unity, then the average energy at the output of transmitter filter is equal to the average energy at the input of the receiver filter. Thus the HF channel model on average does not affect the signal-to-noise ratio of the system. The noise component of *v(t)* is given by eqn.3.2.31, whose Fourier transform is :

$$V(f) = \sqrt{2} \cdot N(f) \cdot C(f+f_c) \cdot B'(f) \qquad 3.2.44$$

And the power spectral density of v(t) is given by :

$$|V(f)|^2 = \sqrt{2} \cdot \sqrt{2} \cdot \frac{N_0}{2} \cdot |C(f+f_c)|^2 \cdot |B'(f)|^2 \qquad 3.2.45$$

With the auto-correlation function $R_v(\tau)$ is given by :

$$R_v(\tau) = N_0 \int_{-f_c}^{f_c} |C(f+f_c)|^2 \cdot |B'(f)|^2 \cdot e^{j2\pi f\tau} \cdot df \qquad 3.2.46$$

The integration limits of eqn.3.2.46 related to the filter B'. From eqn.3.2.45 the variance of *v(t)* is given by :

$$\sigma_v = R_v(0) = N_0 \int_{-f_c}^{f_c} |C(f+f_c)|^2 \cdot |B'(f)|^2 \cdot df \qquad 3.2.47$$

C(f) has the same bandwidth as the spectrum of the QAM signal. If *C(f)* over positive frequencies is symmetric about f_c, then the auto-correlation functions of each the real and

imaginary parts of *v(t)* are equal to half Real $[R_v(\tau)]$ [17], and when $[R_v(\tau)]$ is real, then the real and imaginary parts of *v(t)* are uncorrelated [1,17]. Now, the signal to noise ratio per information bit is defined as :

$$SNR = \Psi = \frac{Transmitted\ \ energy\ \ per\ \ bit}{Noise\ power\ spectral\ \ density}$$

Or :

$$SNR = \Psi = 10 \cdot \log_{10}[\frac{E_B}{N_0/2}] \qquad 3.2.48$$

It is shown in Appendix B-1 that the Signal-To-Noise Ratio for a QPSK signal is :

$$SNR = \Psi = 10 \cdot \log_{10}[\frac{\overline{s_i}^2}{2\sigma^2}] \qquad 3.2.49$$

3.3 EQUIPMENT FILTERS USED IN THE TESTS

The impulse response of the linear baseband channel for a three-sky wave channel is given by eqn.3.2.38 as :

$$\begin{aligned} p_i(t-iT) = \{ & a(t-iT)[q_1(t) - j\,q_2(t)] + \\ & + a(t-\tau_1-iT)[q_3(t) - j\,q_4(t)] + \\ & + a(t-\tau_2-iT)[q_5(t) - j\,q_6(t)]\} * b(t) \end{aligned} \qquad 3.3.1$$

Where a(t) & b(t) is given by :

$$a(t) = a'(t) * (d(t) \cdot e^{-j2\pi f_c t}) \qquad 3.3.2$$

$$b(t) = \{[g(t) * c(t)] \cdot e^{-j2\pi f_c t}\} * b'(t) \qquad 3.3.3$$

Where *a(t) & b(t)* are the impulse responses of the over-all transmitter filter A and the impulse response of the over-all receiver filter B respectively in Fig.3.2.5, and *a'(t), d(t), g(t), c(t)* and *b'(t)* are the impulse responses of Filter A', D, G, C, and B' respectively in Fig.3.3.1. Filter A' is the baseband signal shaping filter. Filter B' is the low-pass filter at the output of the linear demodulator. Filters A' and B' are used to prevent aliasing and have an approximately sinusoidal roll-off in amplitude [24]. Filters D and G are the radio transmitter and radio receiver filters respectively. when the HF channel introduces no fading, attenuation or multipath propagation then, eqn.3.3.1 becomes :

$$p_i(t) = [a(t) * b(t)] \qquad 3.3.4$$

$p_i(t)$ in eqn.3.3.4 should be minimum phase for optimum performance of the detection process, such that $|A(f)| = |B(f)|$ [15].

The group delay and attenuation characteristics of the radio filters D & G in cascade over the positive frequencies is shown in Fig.3.3.1, where the radio filters used are the Clansman VCR321 type. Fig.3.3.2 shows the frequency characteristics of the modem transmitter and receiver filters in cascade and in the pass-band of the QAM signal. This characteristic corresponds to the impulse response :

$$\{a'(t)*[c(t)\cdot e^{-j2\pi f_c t}]*b'(t)\}e^{j2\pi f_c t} \qquad 3.3.5$$

Fig.3.3.3 shows the resultant attenuation and group delay characteristics corresponding to the impulse response :

$$\{a(t)*b(t)\}\cdot e^{j2\pi f_c t} \qquad 3.3.6$$

The attenuation and group delay characteristics corresponding to each *a(t)* and *b(t)* in eqn.3.3.1, are obtained by shifting those in Fig.3.3.3 by $f_c = 1800Hz$ to the left and dividing them by *2*. The characteristics of the equipment filter given in Fig.3.3.1 to Fig.3.3.3 operating on the voice-band signal are explained in detail in [17,26].

Table 3.3.2 gives the sampled impulse response of the minimum phase transmitter filters *a(t)*, *a(t-τ)*, for (*τ =1.1msec, 2msec* and *3msec*), and table 3.3.1 gives the sampled impulse response of the minimum phase receiver filter *b(t)*. The values given in tables 3.3.1 to 3.3.2 have been obtained by [28], and further details regarding these filters are available in refs. [17,26,28]. However it should be noted here, that each of the sampled versions of *a(t)*, *a(t-τ)* and *b(t)*, for (*τ =1.1msec, 2msec* and *3msec*), are minimum phase or (near minimum phase) [17,29], so that the cascade of any one transmitter and receiver filter would result in a minimum phase or (near minimum phase) sequence. Thus in the absence of any fading, the channel formed by the transmitter filter and receiver filter is minimum phase, allowing the detection operation to be optimum.
The sampled impulse response given in tables 3.3.1-3.3.2 assumes a sampling rate of *4800 samples/sec*, the reason for which will become clear in the next section.

3.4 GENERATION OF THE SIR OF THE LINEAR BASE-BAND CHANNEL

The sampled impulse response of the channel as given in eqn.3.3.1 is used to generate the SIR of the HF channel for testing different detectors and estimators. The demodulated signal *r(t)* at the output of the QAM system model is given by eqn.3.2.35 as

$$r(t) = \sum_i s_i \cdot p_i(t - iT) + v(t) \qquad 3.4.1$$

The wave form $r(t)$ is sampled once per data element s_i at the time instant iT. Assuming correct sampling at the receiver and the fact that delay in transmission is zero, (the first potentially non-zero sample of a received signal element arrives without delay), then the complex-value sample of $r(t)$ at $t=iT$ is given by :

$$r_i = \sum_{h=0}^{L} s_{i-h} \cdot p_{i,h} + V_i = P_i \cdot S_i^T + V_i \qquad 3.4.2$$

Where : $p_{i,h} = p_{i-h}(ht)$, $V_i = v(iT)$

$$\mathbf{S_i} = [s_i, s_{i-1}, ., ., ., ., ., ., ., ., s_{i-L}\ ] \qquad 3.4.3$$

$$P_\mathbf{i} = [p_{i,0}, p_{i,1}, ., ., ., ., ., ., ., ., p_{i,L}\ ] \qquad 3.4.4$$

Assuming $p_{i,h} = 0$, *for* $h<0$ & $h>L$; for practical purposes.

P_i & S_i are $(L+1)$-component row vectors, and S_i^T is the transpose of S_i .

The $\{r_i\}$, $\{p_i\}$, $\{v_i\}$ are complex-valued. P_i is taken to be the sampled impulse response of the linear baseband channel at time $t=iT$. P_i in eqn.3.4.4 is generated by sampling $P_i(t-iT)$ in eqn.3.3.1 at the rate of *2400 samples per sec* (i.e., *1/T*), which is the baud rate of the system.

However the convolution process itself in eqn.3.3.1 is carried out in the discrete time domain. In order to avoid any aliasing, when any one of the sequences $q_i(t)$ *for* $i = 1,2,3,.,.,.,6$ changes rapidly, the sampling rate of the convolution is set to *4800 samples/sec*, which is well above the Nyquist rate for filters A and B. $q_i(t)$ *for* $i = 1,2,3,.,.,.,6$ have also been obtained at the sampling rate of *4800 samples/sec* as described in chapter 2, and the corresponding resultant samples up to time $t=iT$ are represented by the following six sequences :

$$QQ_{1,i} = [q_{1,1}, q_{1,2}, ., ., ., ., q_{1,2i}]$$
$$QQ_{2,i} = [q_{2,1}, q_{2,2}, ., ., ., ., q_{2,2i}] \qquad 3.4.5$$
.
.
$$QQ_{6,i} = [q_{6,1}, q_{6,2}, ., ., ., ., ., q_{6,2i}]$$

where :

$q_{k,h} = q_k(h\frac{T}{2})$, *for* $k = 1,2,.,....6$.

Let the sequences A_1, A_2, A_3 *and* B represent the four impulse responses,

$a(t)$, $a(t-\tau_1)$, $a(t-\tau_2)$ *and* $b(t)$ respectively sampled at *4800 samples/sec*, i.e :

$$\begin{aligned} A_1 &= [\, a_{1,0}, a_{1,1}, .., ..., a_{1,\rho} \,] \\ A_2 &= [\, a_{2,0}, a_{2,1}, .., ..., a_{2,\rho}] \\ A_3 &= [\, a_{3,0}, a_{3,1}, .., ..., a_{3,\rho} \,] \\ B &= [\, b_0, b_1, ., ..., b_\rho \,] \end{aligned} \qquad 3.4.6$$

where :

$$a_{1,k} = a(k\frac{T}{2}) \;, a_{2,k} = a(k\frac{T}{2} - \tau_1) \;, a_{3,k} = a(k\frac{T}{2} - \tau_2) \;, b_k = b(k\frac{T}{2}) \qquad 3.4.7$$

and *1/T* is the data-symbol rate of *2400 symbols/sec*. For practical purposes, it is assumed that : $a(t) \approx b(t) \approx 0$ *for* $g < t < 0$.

ρ in eqn.3.4.6 is related to the maximum delay between sky waves (assumed to be τ_2) as [17,26]. g is the length of the SIR of filter A or B.

$$\rho = g + \tau_2 \cdot \frac{2}{T} \qquad 3.4.8$$

Now, from eqns.3.4.5 to 3.4.8 the components of the vector P_i in eqn.3.4.4 at time $t=iT$ are given in descrete form by [17] :

$$\begin{aligned} p_{i,h} = [\frac{T}{2}] \cdot \{ \sum_{k=0}^{2h} [& a_{1,k}(q_{1,2(i-h)+k} - jq_{2,2(i-h)+k}) + \\ & + a_{2,k}(q_{3,2(i-h)+k} - jq_{4,2(i-h)+k}) + \\ & + a_{3,k}(q_{5,2(i-h)+k} - jq_{6,2(i-h)+k})] \cdot b_{2h-k} \} \end{aligned} \qquad 3.4.9$$

for $h = 0, 1, 2, ., ., ., ..., L$.

Where (L) is dependent on the maximum delay τ_2, by the following relation [17,26] :

$$L = \frac{2g + \rho' + 1}{2} \qquad \text{where} \qquad \rho' = \frac{2\tau_2}{T} \qquad 3.4.10$$

and $(T/2)$ is a scaling factor and it is shown in [17], for exact representation of $p_{i,h}$ it must be multiplied by the inverse of the sampling rate of $\{a_k\}$ & $\{b_k\}$.
Thus, the values of $(L+1)$ for channel 1, 2 and 3 are respectively *22*, *22* and *20*.

(eg. $\tau_2 = 3m\text{sec}$ & $(g+1) = 16$ (table 3.3.1)) $\Rightarrow g = 15$, $\rho' = 0.0003 \cdot 4800 = 14.4 \Rightarrow$
$\Rightarrow L = \frac{30 + 14.4 + 1}{2} = \frac{45.4}{2} = 22$). " $L = g + \frac{\rho' + 1}{2}$ ".

$\{p_{i,h}\}$, *for* $h = 0,1,.,.,.,.,...L$ are obtained at a sampling rate of *2400 samples/sec*, by taking every alternative sample from the convolution process.

Appendix F gives the programme for generating the SIR of the linear base band channel $\{p_{i,h}\}$ using eqn.3.4.9.

3.5 MODEL OF THE COMBINED SYSTEM USED IN ALL TESTS

Fig.3.5.1 shows the model of combined system used in the tests of estimators, detectors and adaptive filter. This model is consistent with Fig.3.2.1 and Fig.3.2.5, but it shows in greater detail the receiver configuration. The signal fed to the HF channel model is a four-level QAM (QPSK) signal with carrier frequency $f_c = 1800Hz$ and a signal element rate of *2400 bauds*. Each signal element itself comprises the sum of two binary double sideband suppressed carrier amplitude modulated elements, with their carriers in phase quadrature. The binary values of the in-phase and quadrature elements are determined, respectively by the real and imaginary parts ($s_{1,i}$ & $s_{2,i}$) of the corresponding data-symbol s_i. The QAM signal is fed to the HF channel, and is transmitted via two or three different Rayleigh fading paths (sky waves) to the receiver, with transmission delays τ_1, τ_2 taken to be constant and equal to 1.1 & 3 *msec*. relative to the first path. Each path introduces the average attenuation and the same frequency spread into the data signal. The important properties of the different channels used in the tests are given in Chapter 2.

Stationary white Gaussian noise with zero mean and a two-sided power spectral density of $N_0/2$, is added to the data signal at the output of HF transmission path. The wave forms $q_i(t)$ involved in the channel model have the same variance and the same root mean-square bandwidth in each channel. The model of HF channel used in this thesis is based on the CCIR recommended model and the full detail of this model is given in Chapter 2. The system is assumed to be synchronised. The received samples $\{r_i\}$ at the time instant $t=iT$, which are complex-valued is given by eqn.3.5.2 as :

$$r_i = \sum_{h=0}^{L} s_{i-h} \cdot p_{i,h} + v_i = P_i \cdot S_i^T + V_i \qquad 3.5.1$$

where the vector P_i is the sampled impulse response at time $t=iT$ of the linear baseband channel and is given by eqn.3.4.4. The received samples $\{r_i\}$ are also fed to the channel estimator, after being suitably delayed. The channel estimator uses the received samples $r_{i-L}, r_{i-L+1}, ..., ...r_i$ together with the early detected data-symbols $s''_{i-L}, s''_{i-L+1}, ., ., ...s''_i$ to give the estimate of P_i, where :

$$P'_i = [\, p'_{i,0}, p'_{i,1}, ..., ... p'_{i,L} \,] \qquad 3.5.2$$

where P'_i is the estimated sampled impulse response (SIR) of the channel.
The estimated SIR (P'_i) is fed to the minimum phase algorithm, which tries to determine the roots of the SIR which lie outside the unit circle in the Z-plane and then replaces these roots by another set of roots which now lies inside the unit circle. This algorithm attempts to make the sequence $\{p'_{i,h}\}$, (for $h = 0,1,.,.,...., L$), a minimum phase sequence $\{y'_{i,h}\}$, for $h = 0,1,.,.,...., L$. A minimum phase channel would have most of its energy concentrated in the first few components, which is needed by the near-maximum likelihood (NML) detector. The minimum phase algorithm operates on the roots of the Z-transform of the sequence $\{p'_{i,h}\}$, for $h = 0,1,.,.,...., L$ that have absolute values greater than 1.05, by replacing these roots by the reciprocals of their complex conjugates [23], using an iterative process. The full detail of this algorithm will be explained later in (chapter 4).

The output of the minimum phase algorithm $\{y'_{i,h}\}$, for $h = 0,1,.,.,...., L$, is then fed to the NML detector with the received samples $\{r_i\}$ for detection of the transmitted data symbols $\{s'_{i-n}\}$. The output of minimum phase algorithm can be used to adjust an adaptive linear filter just ahead of the NML detector. The output of the minimum phase algorithm is now given by :

$$\begin{aligned} Y'_i &= \text{Min.phased of } [P'_i] \\ \{y'_{i,h}\} &\approx \text{Min.phased of } \{p'_{i,h}\} \quad for \quad h = 0,1,.,.,.,.. L \end{aligned} \qquad 3.5.3$$

The minimum phase sequence $\{y'_{i,h}\}$, is now scaled such that its first component is unity.

This scaling is carried out for convenience in testing the detector. The scaled minimum phased SIR of the channel is given by $y_{i,h}$ as :

$$\{y_{i,h}\} \approx \text{scaled } \{y'_{i,h}\} = \{\frac{y'_{i,h}}{y'_{i,0}}\} \quad for \quad h = 0,1,.,.,.,.. L \qquad 3.5.4$$

The resultant SIR vector of the time-varying linear base-band channel is :

$$Y_i = [y_{i,0}, y_{i,1},.,.,.,.,.,., y_{i,L}\ ] \text{ whose } \ y_{i,0} = 1 \qquad 3.5.5$$

A similar scaling is performed on the additive noise sequence $\{v_i\}$ as well, in order to avoid any undue change in the signal-to-noise ratio of the system, such that each component of $\{v_i\}$ is divided by $y_{i,0}$:

$$\{v_i\} = \{\frac{v'_i}{y'_{i,0}}\} \qquad 3.5.6$$

Bear in mind that the sequence $\{v_i\}$ of the Gaussian random variables is passed via a scaled version of the same receiver filter that the signal passes through (Filter B).
The $\{v'_i\}$ are generated as follows :

A sequence of statistically independent, complex-valued random variables with zero mean and a variance of σ^2 in each of the real and imaginary parts, is generated using a software NAG library (routine *G05DDF*). This sequence is next convolved with $\{b'_k\}$, where $\{b'_k\}$ is the sequence obtained by scaling the sequence of receiver filter B $\{b_k\}$ such that $\sum|b'_k|^2 = 1$, and the sequence $\{b'_k\}$ is given in Table 3.3.1 and derived from the sequence $\{b_k\}$ given in Table 3.3.1 using :

$$b'_k = \frac{b_k}{M}, \quad for \quad k = 1,2,3,.,.,.,.,...,16$$

where

$$M = \sqrt{\sum_{i=1}^{16}|b_i|^2} \quad ; \quad |b_i| = \sqrt{\Re\overline{b_i}^2 + \Im\overline{b_i}^2}$$

It should be noted however that the noise sequence obtained at the output of the scaled version of the receiver filter B, would correspond to a sampling rate of *4800 samples/sec*, due to the $\{b'_k\}$, being at this rate. Thus, it is necessary to choose every other sample at the output of the scaled receiver filter, thereby ensuring that the noise samples correspond to a sampling rate of *2400 sampling /sec*, which is the baud rate of the system. Therefore the noise samples $\{v'_i\}$, are no longer uncorrelated, the correlation having been introduced by the scaled version of the receiver filter.

The detected data-symbols $\{s'_i\}$ at the output of the NML detector in Fig.3.5.1, have a delay in detection of (μ=*32*) sampling intervals introduced by the NML detector. No significant reduction in error rate is achieved by any further increase in the delay in detection [23,35]. In the case of a non-linear equaliser with threshold detector no delay is introduced in the detection of data-symbols $\{s'_i\}$. The important advantage gained by the minimum phase algorithm and adaptive filter in Fig.3.5.1 is that it avoids the need for prediction used by an estimation process over many sampling intervals, such as must be used in the absence of the filter [23,30,31,32], and the number of vectors used by the NML detector can be reduced. The performance of NML detector, channel estimator and minimum phase algorithm will be given in the following chapters. The differential coding-decoding in Fig.3.5.1 is used to convert the sequence $\{\alpha_i\}$ into the $\{s_i\}$, which prevents a serious error-extension effect (that is, a prolonged burst of errors in the $\{\alpha_i\}$) resulting from a large and rapid carrier phase change during a deep fade [17,23,26]. The full details of differential coding-decoding is given in Appendix B2.

3.6 COMPUTER SIMULATION RESULTS

Computer simulation tests have been carried out to generate the SIR of the four channel models named channel 1, 2, 3 and 4. The SIR are generated for each channel using eqn.3.4.9 and the parameters for each channel and the wave form $q_i(t)$ are given in chapter 2. The tests of the detector, minimum phase algorithm, estimator and the combined detection estimation are carried out over these channel models.

The results of the tests carried out on a simulated model are summarised in table 3.6.1 and Figs.3.6.1-3.6.12. Table 3.6.1 shows the measured, mean and variance value of the sequence q_i after the convolution process for the four channels model considered in this tests. Comparing table 3.6.1 with table 2.9.1 (chapter 2), shows that the convolution, i.e. linear interpolation process, does not unduly change the characteristic of the sequence q_i. In other words there is no change in the mean and variance value of the q_i .

Figs.3.6.1-3.6.4 show the characteristics of four channels over a duration of *24 seconds* of transmission, assuming the transmitted signal is a stream of pulses. Each figure represents the magnitude variation of the SIR of the channel, which is given by :

$$Magnitude = 10 \cdot \log_{10}[\sum_{h=0}^{L} (\overline{\Re p_{i,h}}^{2} + \overline{\Im p_{i,h}}^{2})]$$

where $\Re p_{i,h}$ & $\Im p_{i,h}$ are the real and imaginary part of $p_{i,h}$ respectively. It can be seen from Figs.3.6.1-3.6.4 that the magnitude variation is random and changes in unpredictable manner, where a typical worst fading sequence obtained from channel 2 which is 2-sky wave with large frequency spread (*2Hz*) and large time delay. Therefore the initial choice of a detector or estimator for the combined system would be based on its performance over channel 2, where channels 1 and 3 are the mildest. It is necessary to test the detector and estimator over such channels for comparison purposes. In terms of the depths of the fades, there is at least one fade in excess of *-35dB*, as can be seen from Fig.3.6.4 for channel 4. The depths of fades are expected to change, when other sky waves are added as in channel 1, 2 and 3. The phase variation of these channels is given by :

$$\theta = \tan^{-1} (\sum_{h=0}^{L} \Im p_{i,h} \Big/ \sum_{h=0}^{L} \Re p_{i,h})$$

Figs.3.6.5-3.6.6 show the phase variation of channel 1 and 2. It can be seen from these figures that the phase varies between $(-\pi \,\&\, \pi)$ randomly and the distribution of the phase is uniform. Figs.3.6.7-3.6.8 show the magnitude variation of channel 1 and 2 before and after convolution process. It is clear from these figures that the convolution process

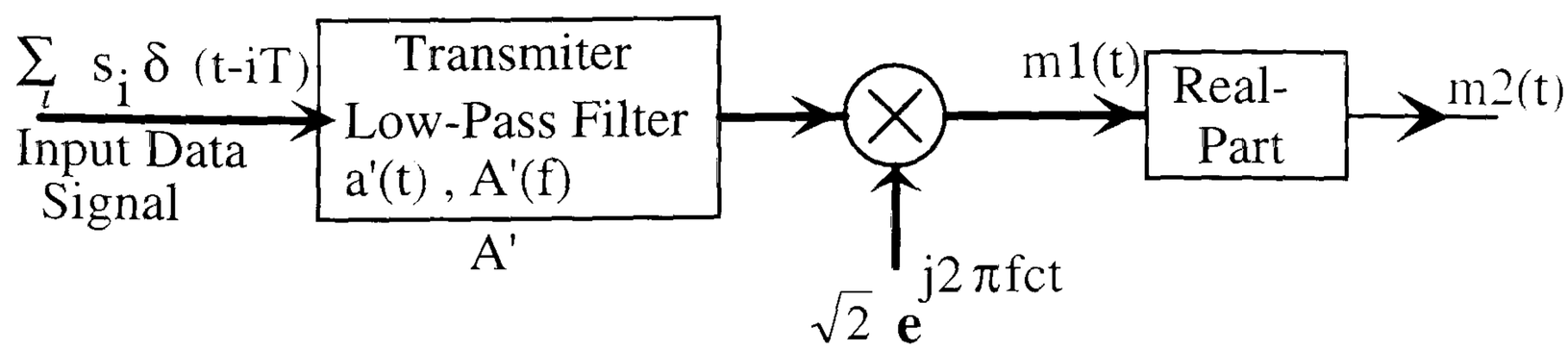
(including transmitter filter A and receiver filter B in the channel model) does not produce any distortion to the signal. Therefore the model, that is considered previously is a fair, if not exact representation of HF channel.

The results of the convolution process of the transmitter filter A, receiver filter B and HF channel are also represented in three-dimension graphs. Figs. 3.6.9-3.6.12 show a segment of the amplitude variation of the SIR of four channels in three-dimensions, where :

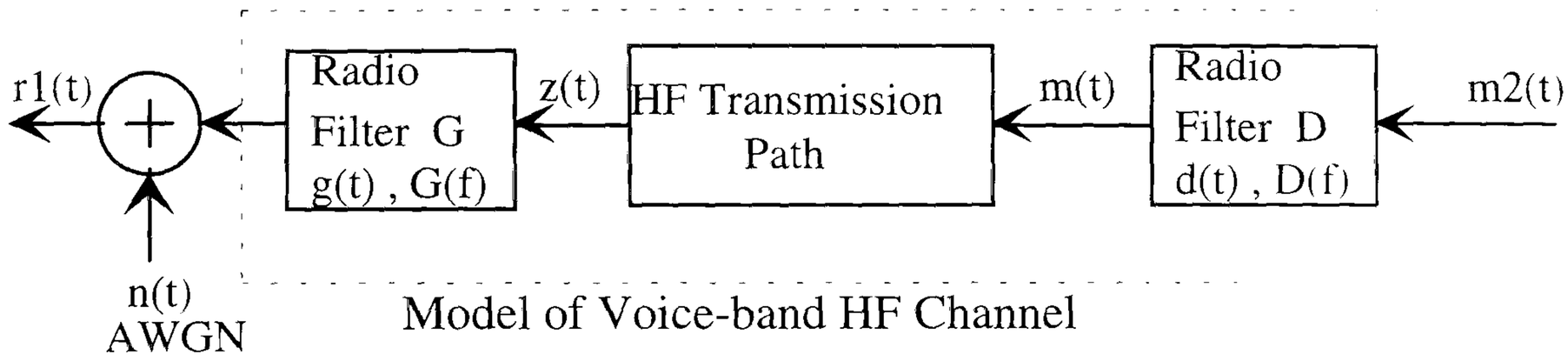
$$Amplitude_{\,i,h} = AM_{\,i,h} = [\overline{\Re p_{i,h}}^{\,2} + \overline{\Im p_{i,h}}^{\,2}] \quad for \quad h = 0,1,.,.,.....,L$$

These figures show different segments of the amplitude variation of the sampled impulse response (SIR) of the channel, which indicate that the variation is random. For every instant of time (i) there are ($L+1$)-components in the SIR (P_i), and it is not necessary for the largest component to be the first one, which is due to the phase distortion introduced by the HF channel. This is a clear reason for using some technique to remove this phase distortion by changing the channel from non-minimum phase to near-minimum phase. Such a technique is called a minimum phase algorithm. It can also be seen from the graphs that these channels introduce amplitude distortion and inter-symbol interference.

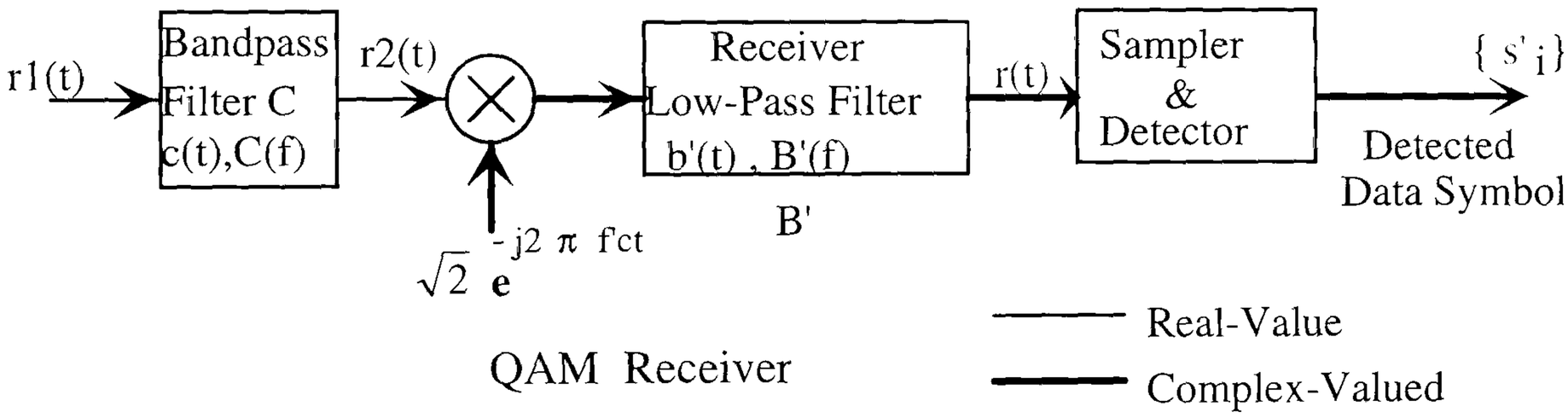
In each graph in Figs.3.6.9-3.6.12, the X-axis represents the number of components in the SIR of the channel ($L+1$), the Y-axis represents the time instant (i), where (i) is taken to be ($1,2,.,.,.....,50$) sample interval, and the Z-axis represents the amplitude variation of the SIR of the channel. The duration of each segment is written beside each graph in Figs.3.6.9-3.6.12. For example ($i=9600$-9650) means 50 samples intervals are involved in drawing the graph, where ($i=(9600$-$10776)mod24$) means that 50 samples intervals are selected from 1200 samples every 24 samples. It can be seen from the figures with ($mod24$), that channel 1 and 2 exhibit 3-sky waves and 2-sky waves respectively and they are the most severe channels, since they have a large frequency spread and time delay.



QAM Transmiter



Model of Voice-band HF Channel



QAM Receiver

Fig.3.2.1 : Model of QAM System over HF Channel

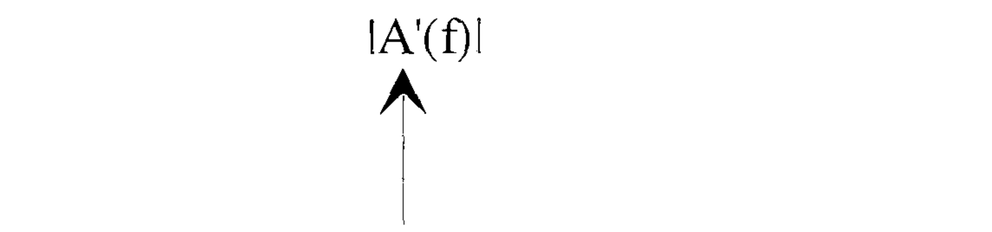


Fig.3.2.2 : Amplitude Spectrum of Filter (A') at Transmitter.

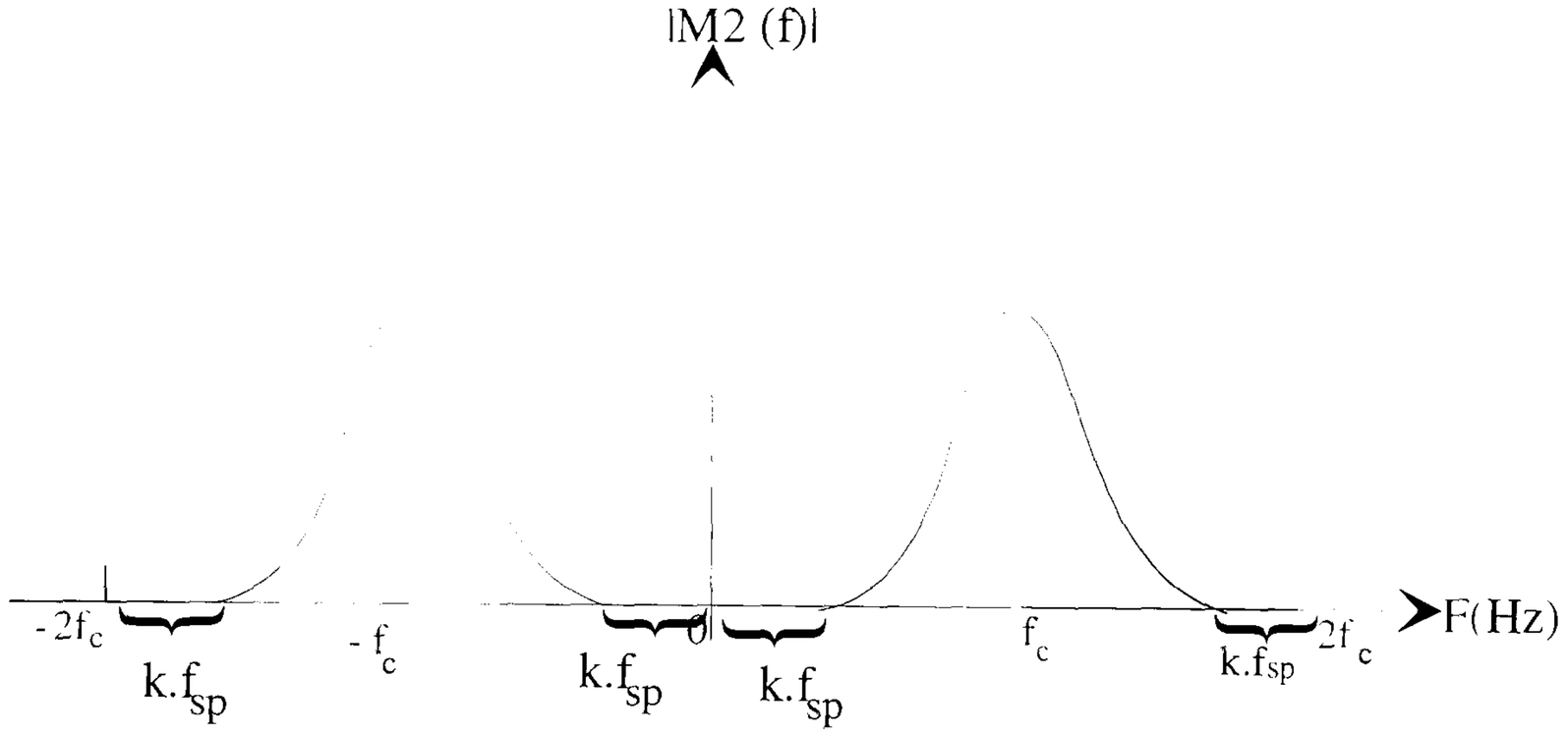
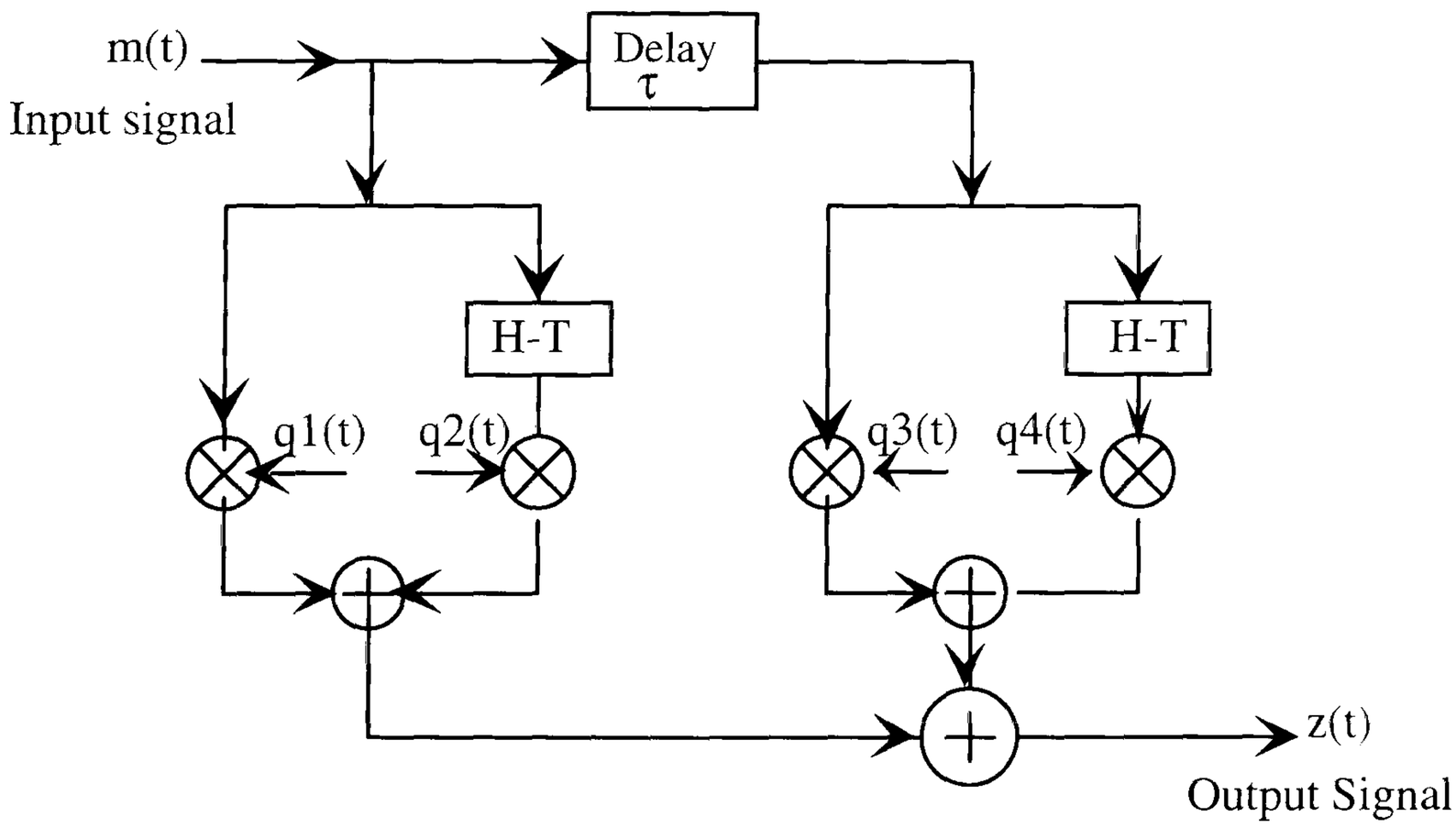


Fig.3.2.3 : Amplitude Spectrum of the QAM Signal m2(t).



where H-T is the Hilbert transform.

Fig. 3.2.4 : Two-sky wave HF Channel Model.

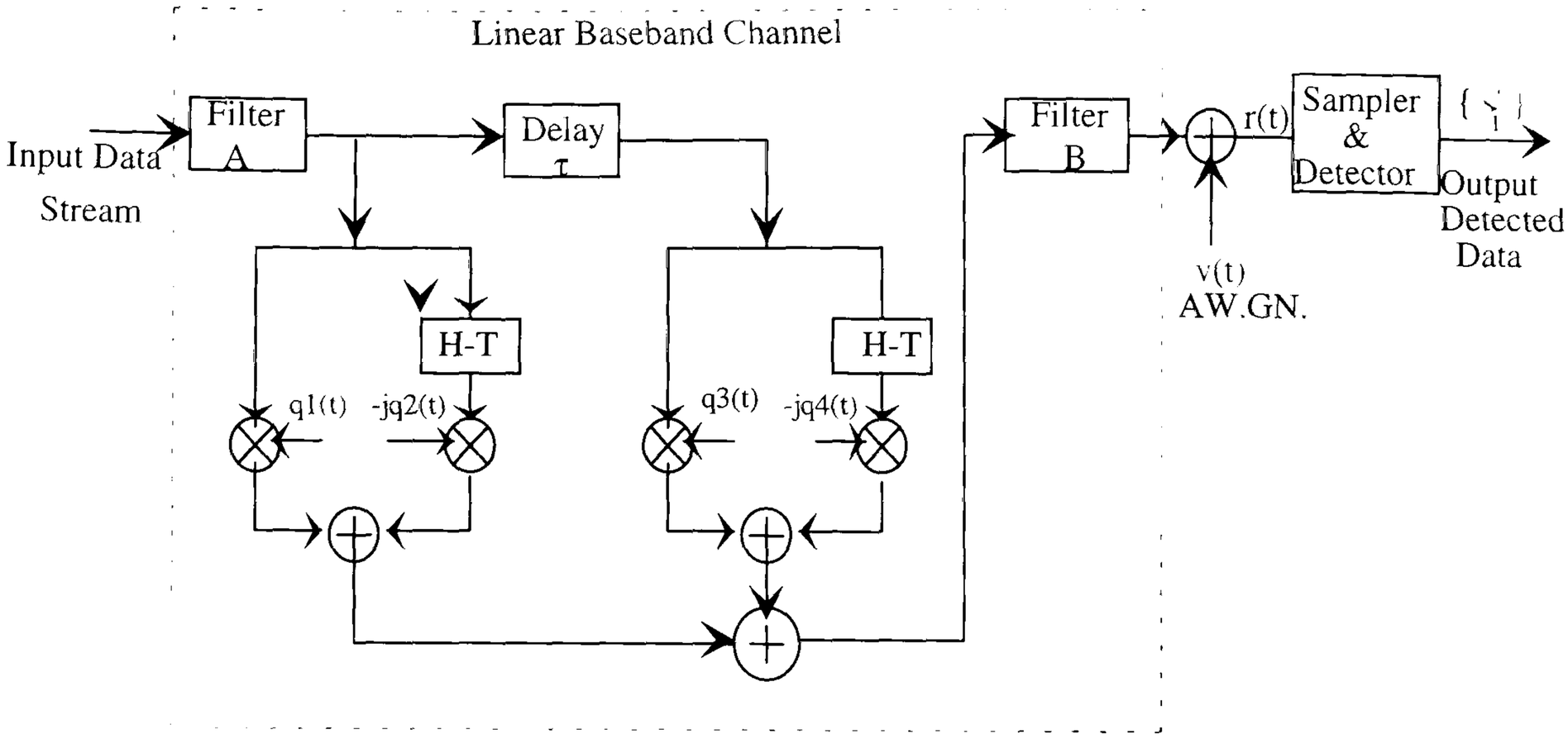


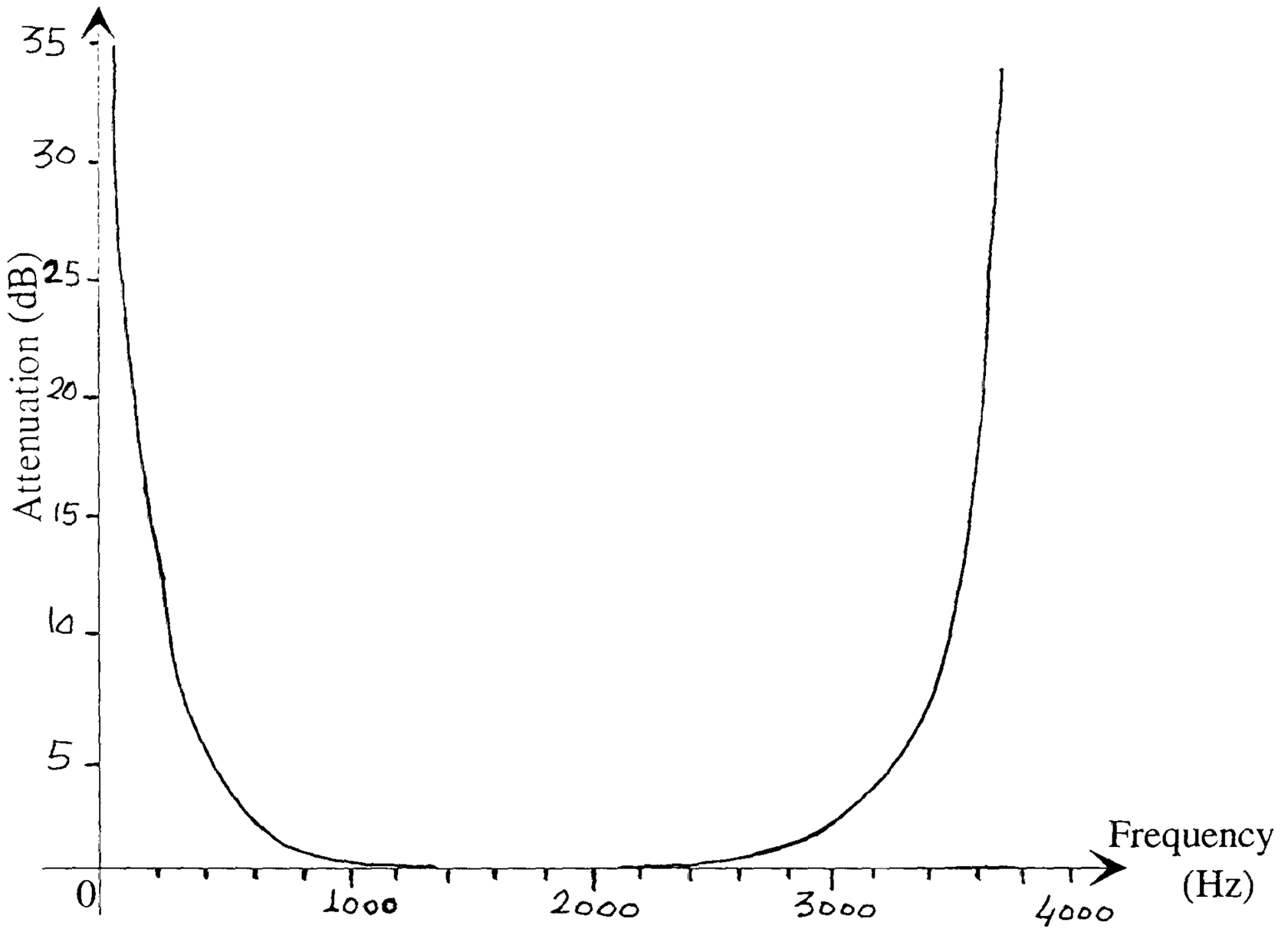
Fig.3.2.5 : An Alternative Baseband Model of QAM System over Two-sky wave HF Channel Model.

sampled response of Filter	impulse transmitter τ =0.0msec	sampled response of Filter	impulse receiver $\{b_k\}$	sampled response of receiver	impulse scaled Filter $\{b'_k\}$
Real part	Imaginary Part	Real part	Imaginary Part	Real part	Imaginary Part
-0.1796	2.3539	-1.9418	1.3626	-0.02805	0.01968
-3.0074	20.7590	-15.9798	11.5941	-0.23081	0.16746
-9.9409	45.5585	-35.1418	27.3343	-0.50758	0.39481
-11.7870	41.4910	-34.4789	28.0870	-0.49800	0.40568
-3.4618	8.7046	-11.2302	7.2715	-0.16221	0.10503
4.4438	-11.7870	7.8155	-9.2602	0.11289	-0.13375
3.0643	-5.5819	7.5124	-5.0954	0.10851	-0.0736
-1.3597	3.1582	-0.5058	3.2327	-0.00731	0.04669
-1.4974	1.7365	-3.3707	1.8975	-0.04869	0.02741
0.2926	-0.7777	-0.6760	-1.2817	-0.00976	-0.01851
0.5181	-0.1293	1.0483	-0.4830	0.01514	-0.00698
-0.1843	0.2880	0.3622	0.7615	0.00523	0.01100
-0.3168	-0.2325	-0.3106	0.1979	-0.00449	0.00286
0.0022	-0.2108	0.0438	-0.1533	0.00063	-0.00221
-0.0444	0.0392	0.0739	0.0940	0.00107	0.00136
0.0516	0.0099	-0.0647	-0.0312	-0.00093	-0.00045

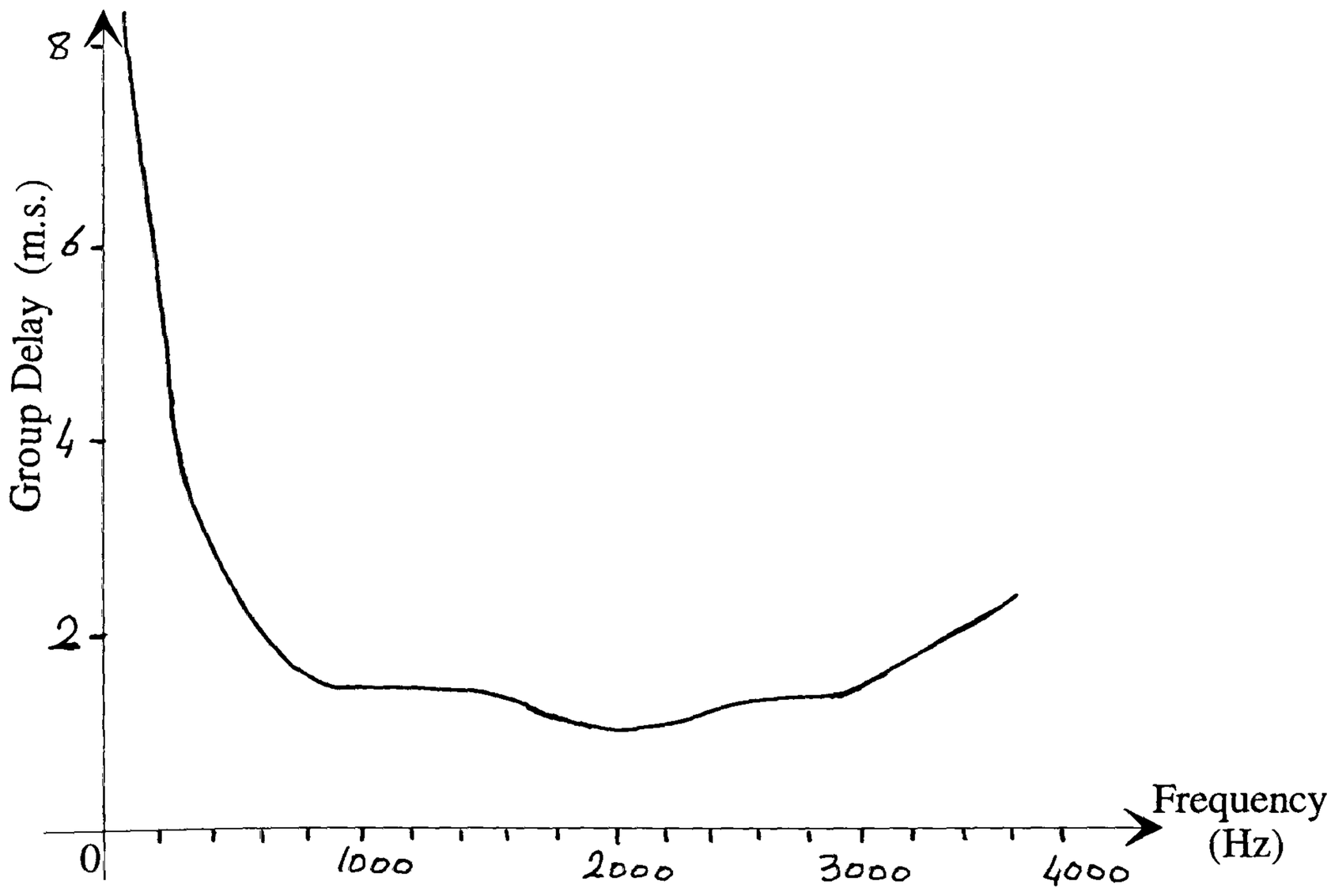
Table.3.3.1 : The Minimum Phase Sampled Impulse Response of the Transmitter , Receiver and the Scaled Version of Receiver Filter (A1, B,B').

sampled impulse response of transmitter Filter τ =1.1msec		sampled impulse response of transmitter Filter τ =2.0msec		sampled impulse response of transmitter Filter τ =3.0msec	
Real part	Imaginary Part	Real part	Imaginary Part	Real part	Imaginary Part
0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
-1.6694	13.2373	0.0000	0.0000	0.0000	0.0000
-7.8492	39.6494	0.0000	0.0000	0.0000	0.0000
-12.3887	46.9272	0.0000	0.0000	0.0000	0.0000
-6.6023	19.2347	0.0000	0.0000	0.0000	0.0000
2.9409	-8.8804	-0.7630	7.3452	0.0000	0.0000
4.3005	-9.0256	-5.6487	31.9050	0.0000	0.0000
-0.3368	1.6284	-11.9216	48.7718	0.0000	0.0000
-1.9014	2.8139	-9.3589	29.8080	0.0000	0.0000
-0.1434	-0.4311	0.5650	-3.0208	0.0000	0.0000
0.6243	-0.4537	4.9376	-11.4980	-1.3137	11.0689
0.0279	0.3082	1.0473	-0.9823	-7.1104	37.2137
-0.3820	-0.0772	-1.9766	3.5053	-12.3470	47.9575
-0.0417	-0.3043	-0.7165	0.3116	-7.5849	22.8263
-0.0440	0.0085	0.5944	-0.7219	2.2354	-7.2499
0.0749	0.0094	0.2544	0.2045	4.5939	-10.0027
-0.0594	0.0095	-0.3636	0.1085	0.0932	0.8695
		-0.1544	-0.3287	-1.9704	3.1073
		-0.0228	-0.0636	-0.3234	-0.2261
		0.0167	0.0279	0.6313	-0.5553
		-0.0610	0.0186	0.1036	0.2882
				-0.3866	-0.0157
				-0.0735	-0.3216
				-0.0387	-0.0108
				0.0608	0.0141
				-0.0710	0.0136

Table.3.3.2 : The Minimum Phase Sampled Impulse Response of the Transmitter Filters For Different Delays, and shown relative to theTransmitter Filter with (t = 0.0 *msec*), (Filter A2 & A3).

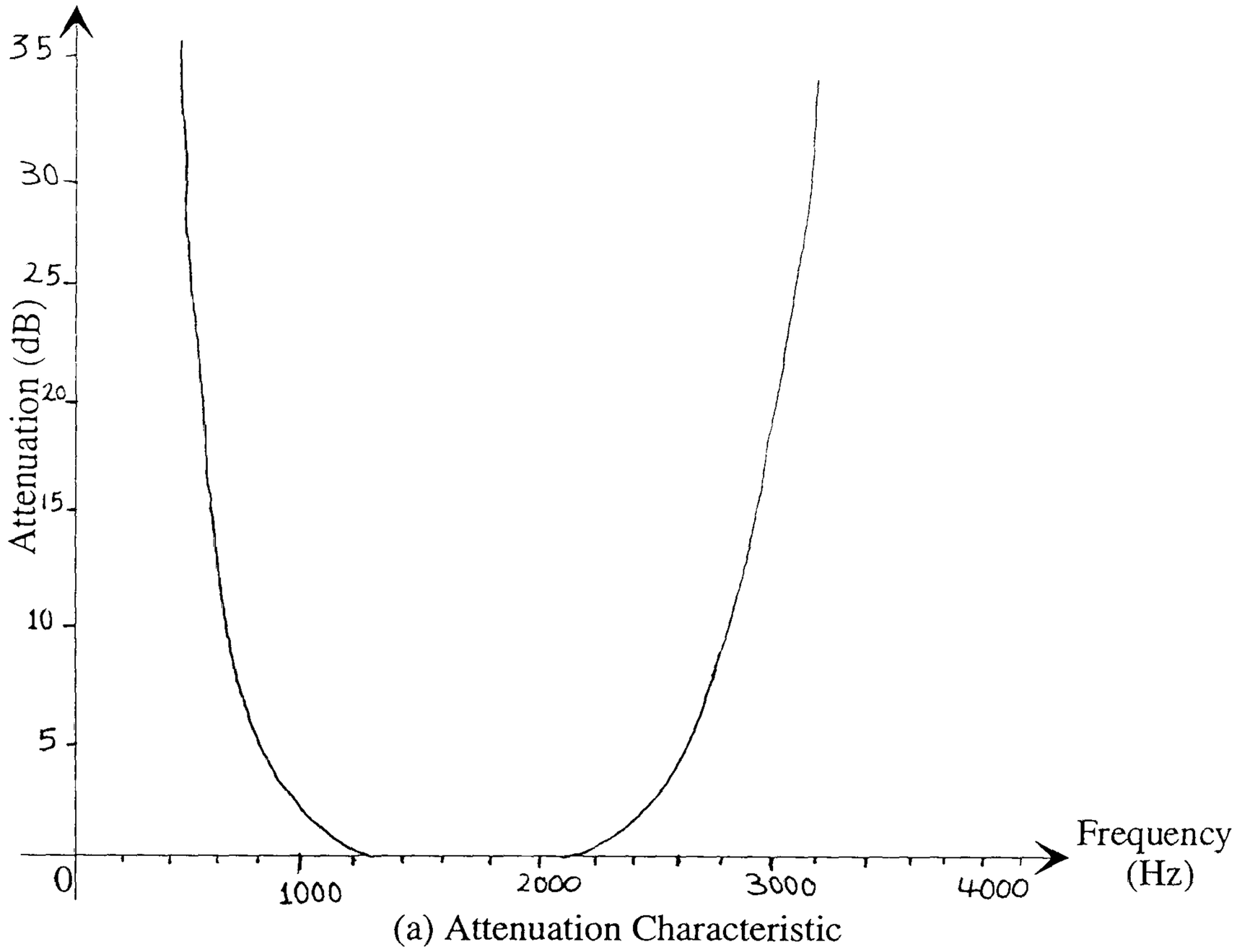


(a) Attenuation Characteristic

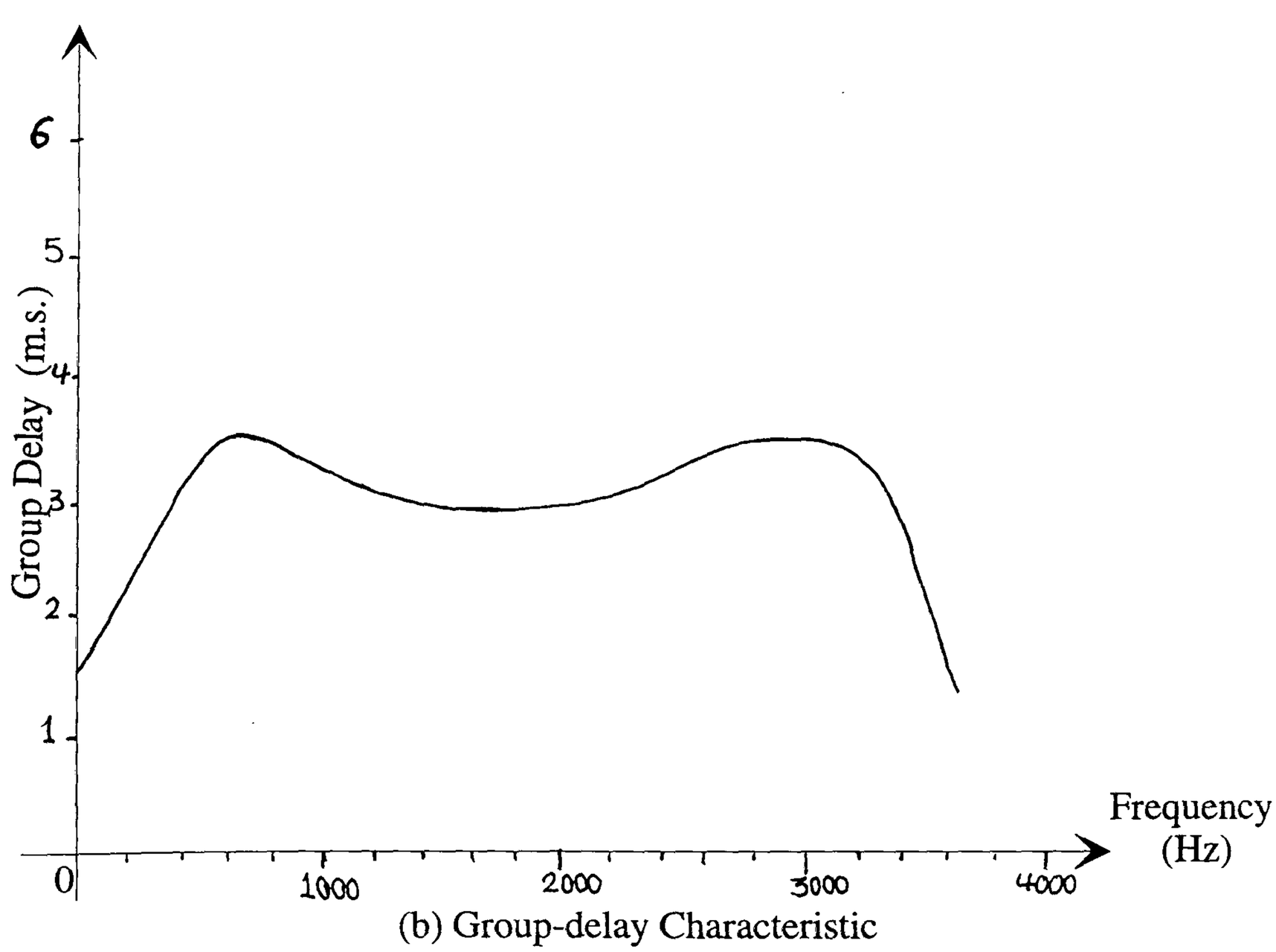


(b) Group-delay Characteristic

Fig.3.3.1 : Frequency Characteristics of the Radio Filters D & G in Cascade.

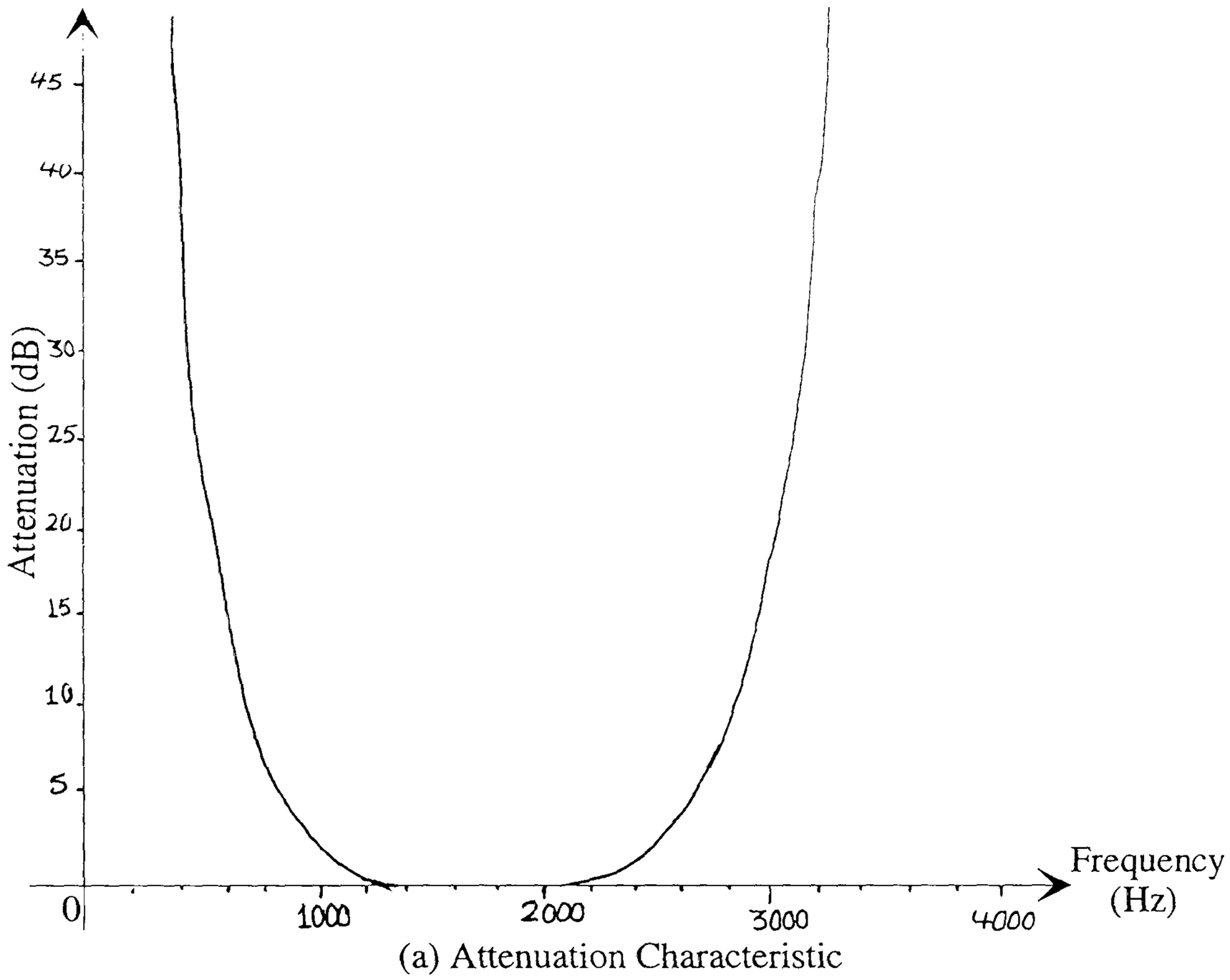


(a) Attenuation Characteristic

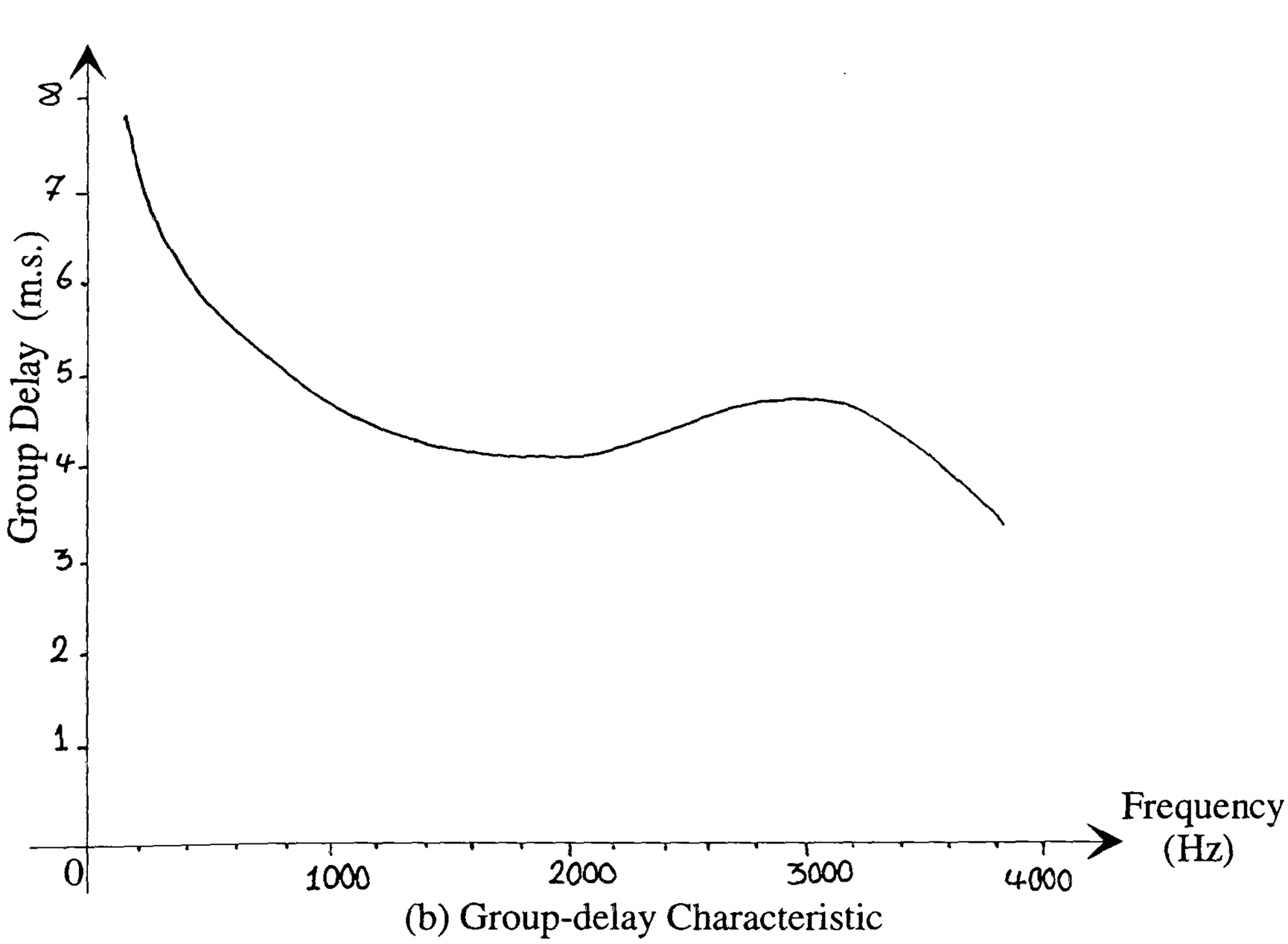


(b) Group-delay Characteristic

Fig.3.3.2 : Frequency Characteristics of the Modem Transmitter and Receiver Filter in Cascade (in the passband of the QAM signal) corresponding to: $\{a'(t)*[c(t)e^{-j2\pi f_c t}]*b'(t)e^{j2\pi f_c t}\}$.



(a) Attenuation Characteristic



(b) Group-delay Characteristic

Fig.3.3.3 : Frequency Characteristics of the Modem Transmitter and Receiver Filter in Cascade (in the passband of the QAM signal in the absence of transmission path) corresponding to: $\{[a(t)*b(t)]e^{j2\pi f_c t}\}$.

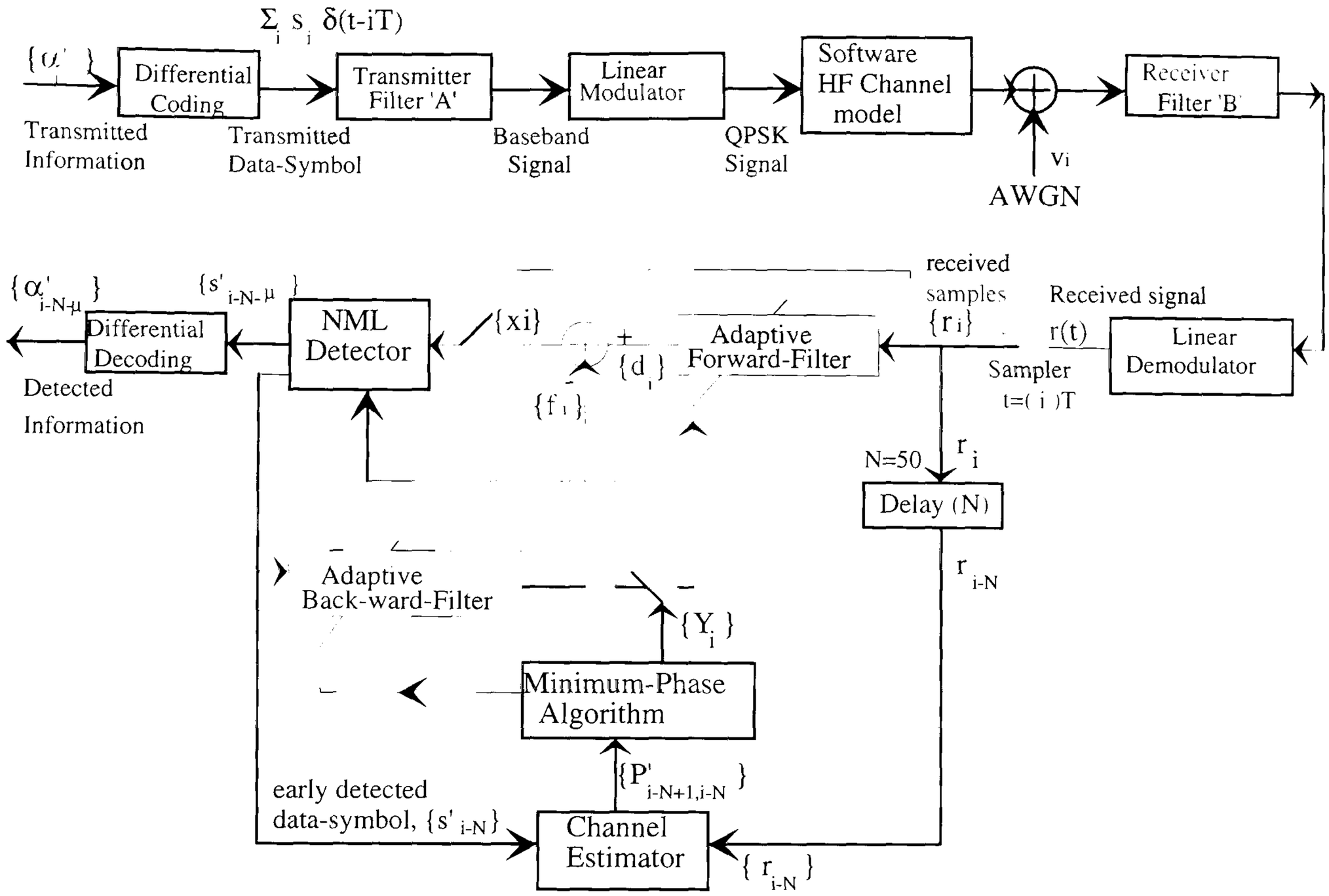


Fig.3.5.1 : The Model of Combined Data Transmission System Over HF Channel.

Channel Type	Seed Number (ID)	Mean value of q_i	Variance of q_i
Channel1 , Fsp=2Hz	75	-0.077125 -0.025706 0.06670 0.019894 0.095368 0.10940	0.15973434 0.165137 0.12990152 0.1542936 0.1891166 0.20080
Channel2 , Fsp=2Hz	57	0.07247 0.019904 0.071298 0.017518	0.24103821 0.234402 0.312247 0.2169373
Channel3 , Fsp=1Hz	57	0.109847 0.0406619 0.0989 0.0192763	0.2199507 0.2310261 0.3298936 0.2232396
Channel4 , Fsp=1Hz	57	0.16356 0.060544	0.487631 0.512185

Table 3.6.1 : Measured Characteristics of the Fading Sequences used to Model the HF Channel after convolution.

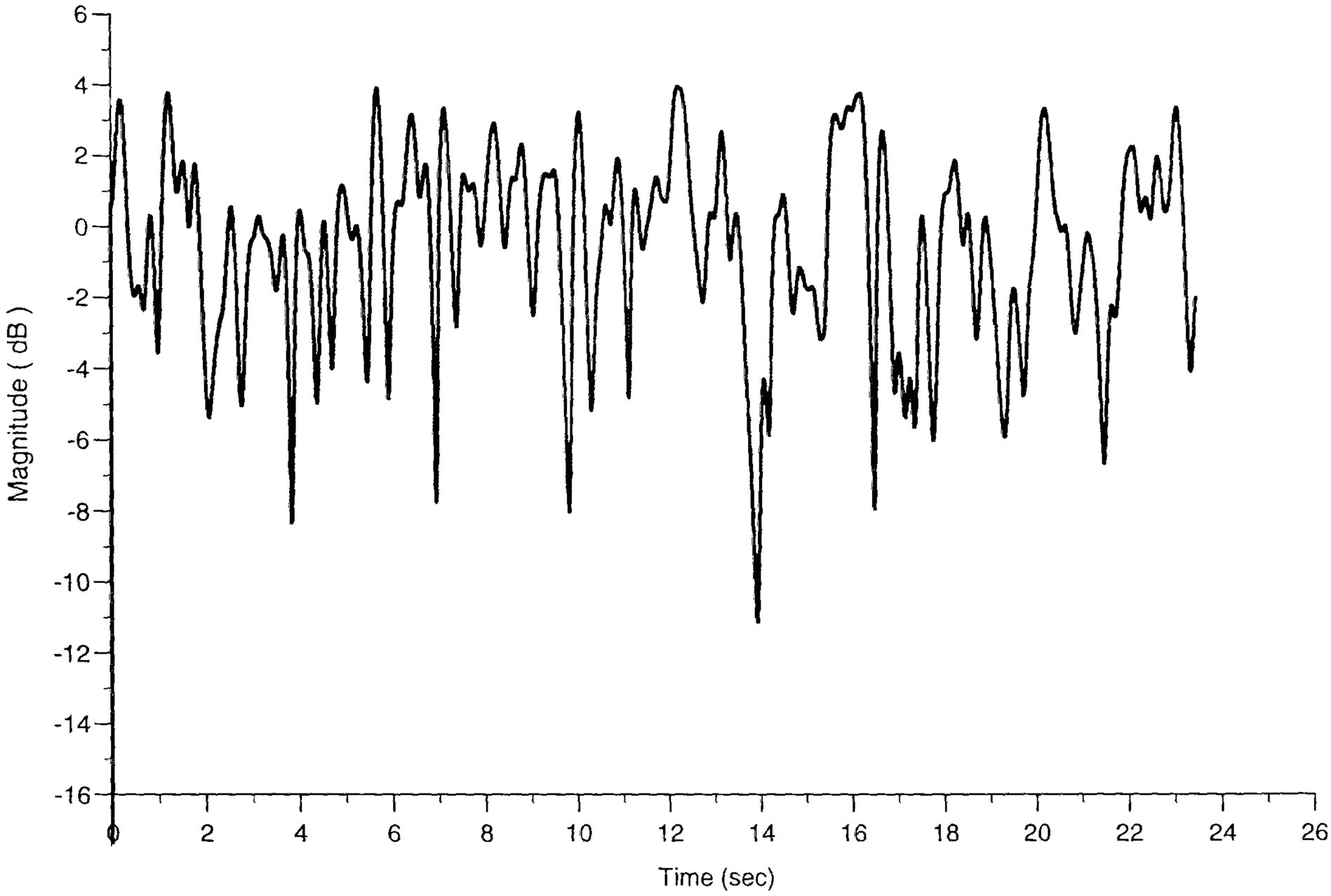


Fig. 3.6.1 The Magnitude Variation of Sampled Impulse Response of Channel 1.

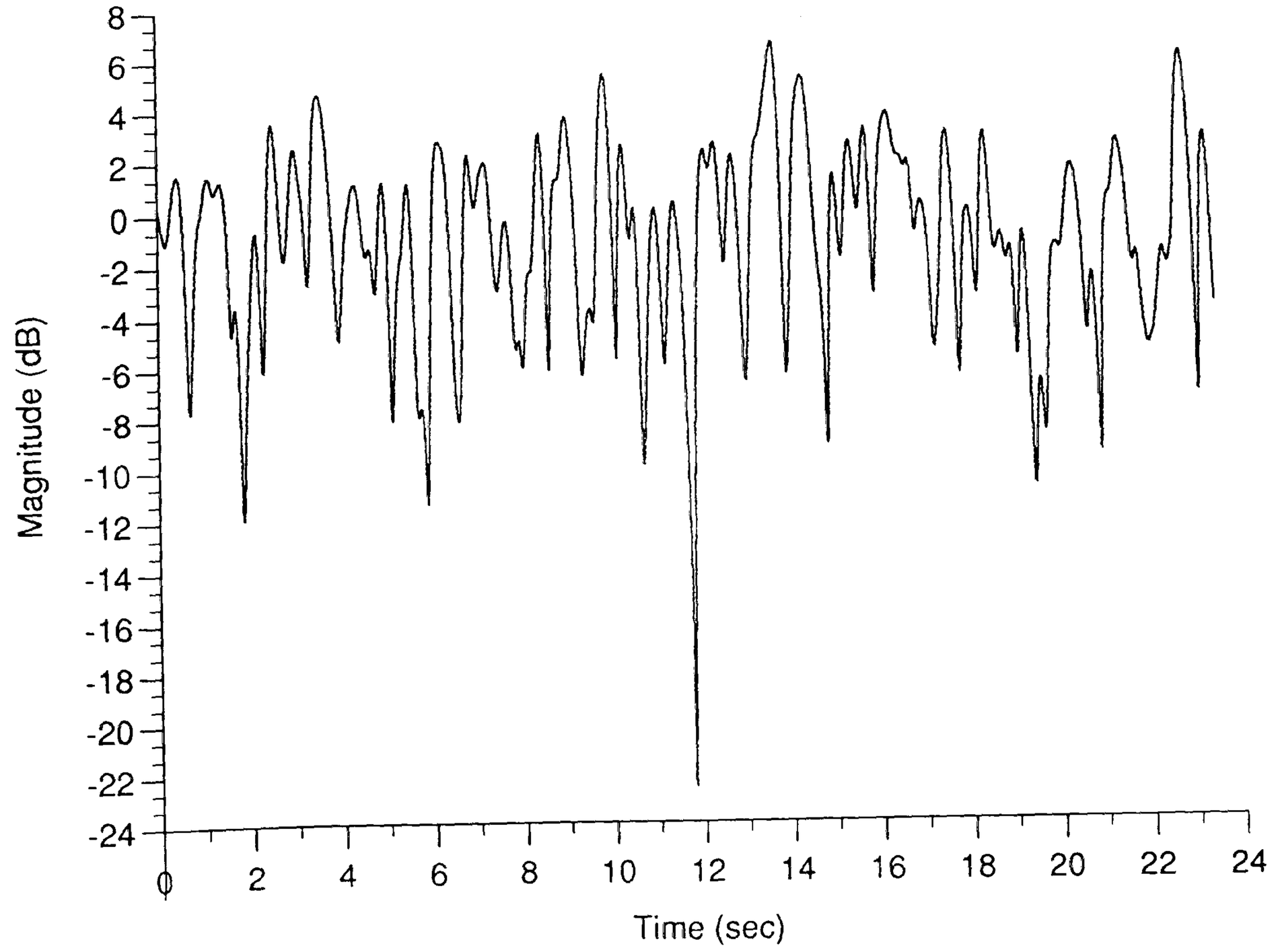


Fig. 3.6.2 The Magnitude Variation of Sampled Impulse Response of Channel 2.

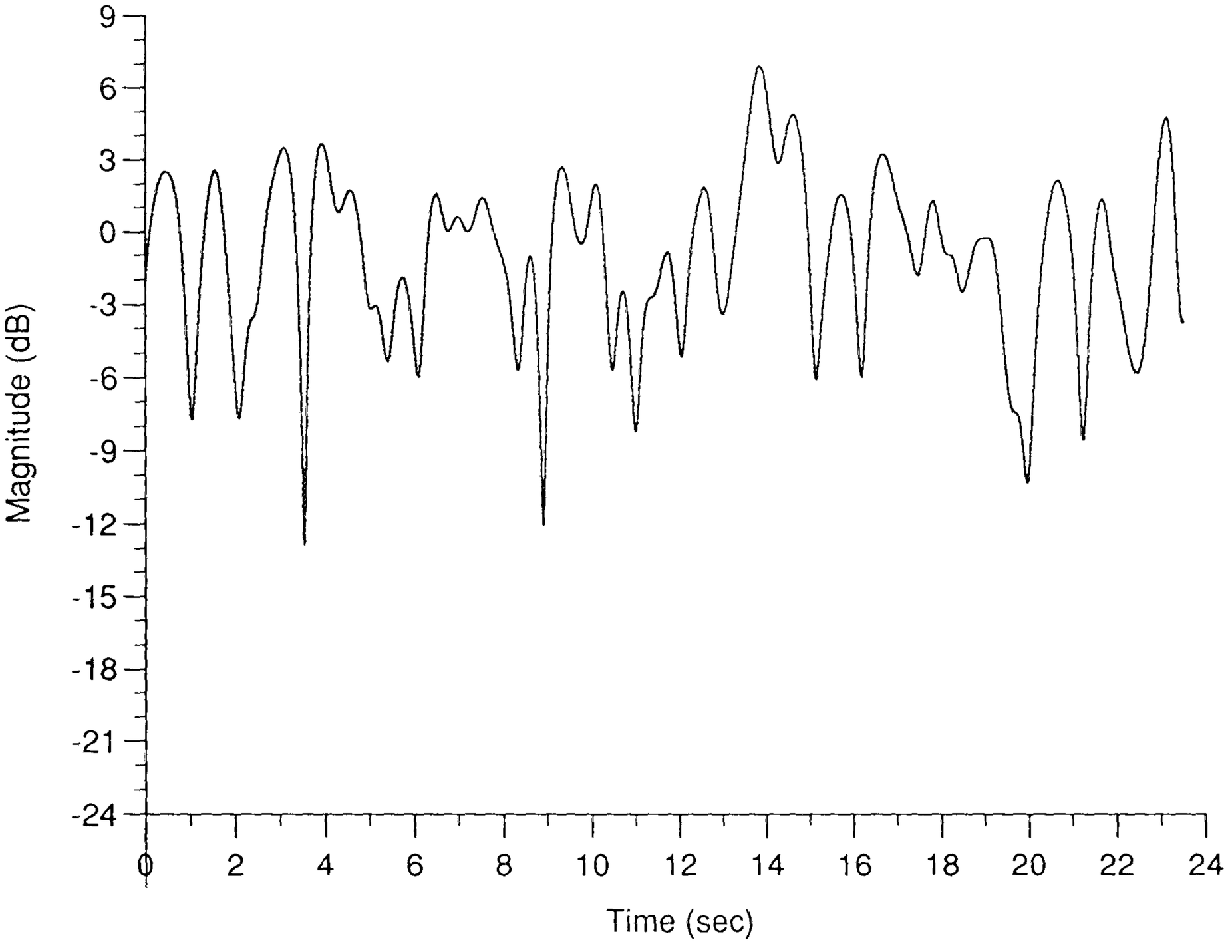


Fig. 3.6.3 The Magnitude Variation of Sampled Impulse Response of Channel 3.

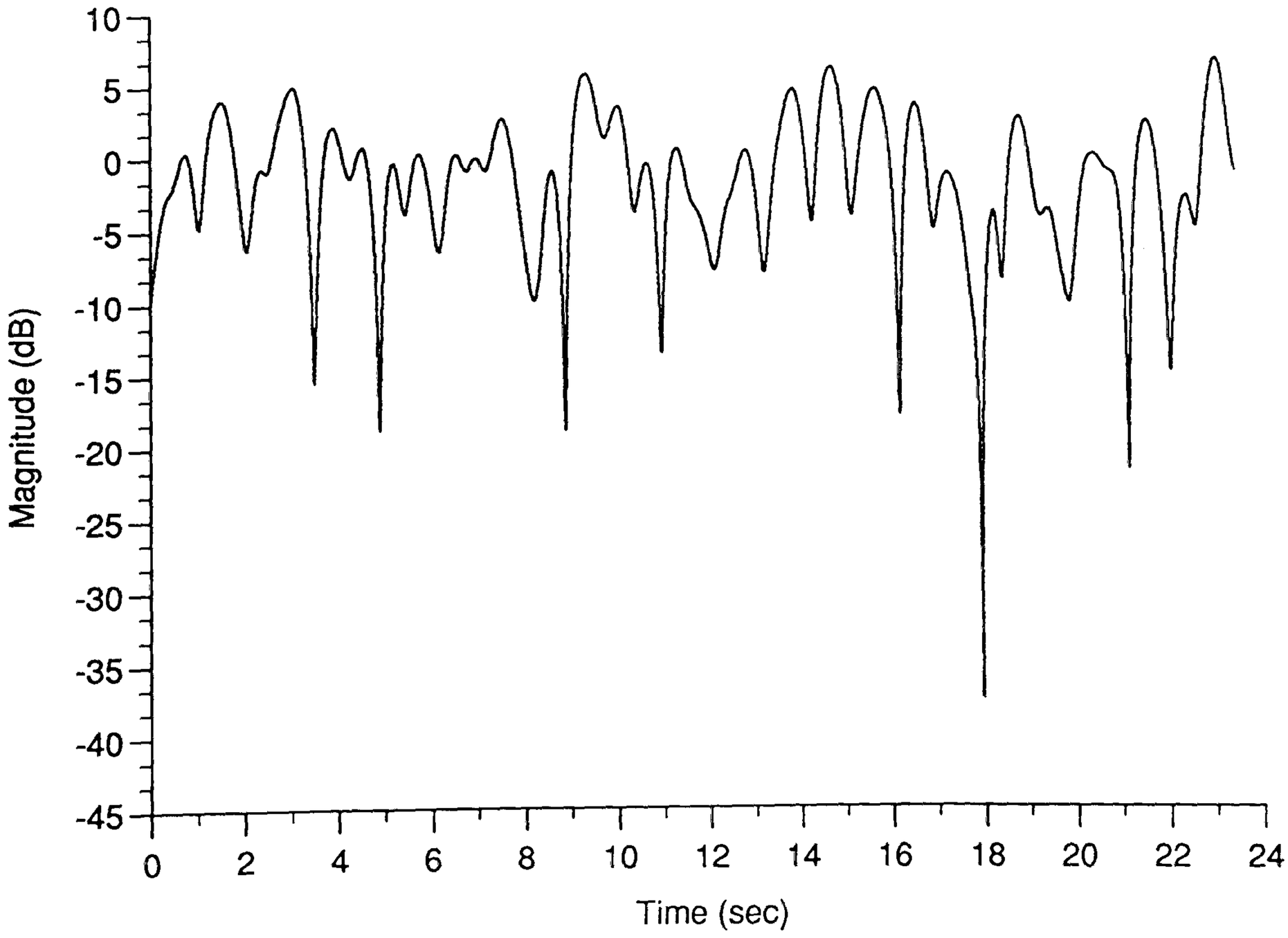


Fig. 3.6.4 The Magnitude Variation of Sampled Impulse Response of Channel 4.

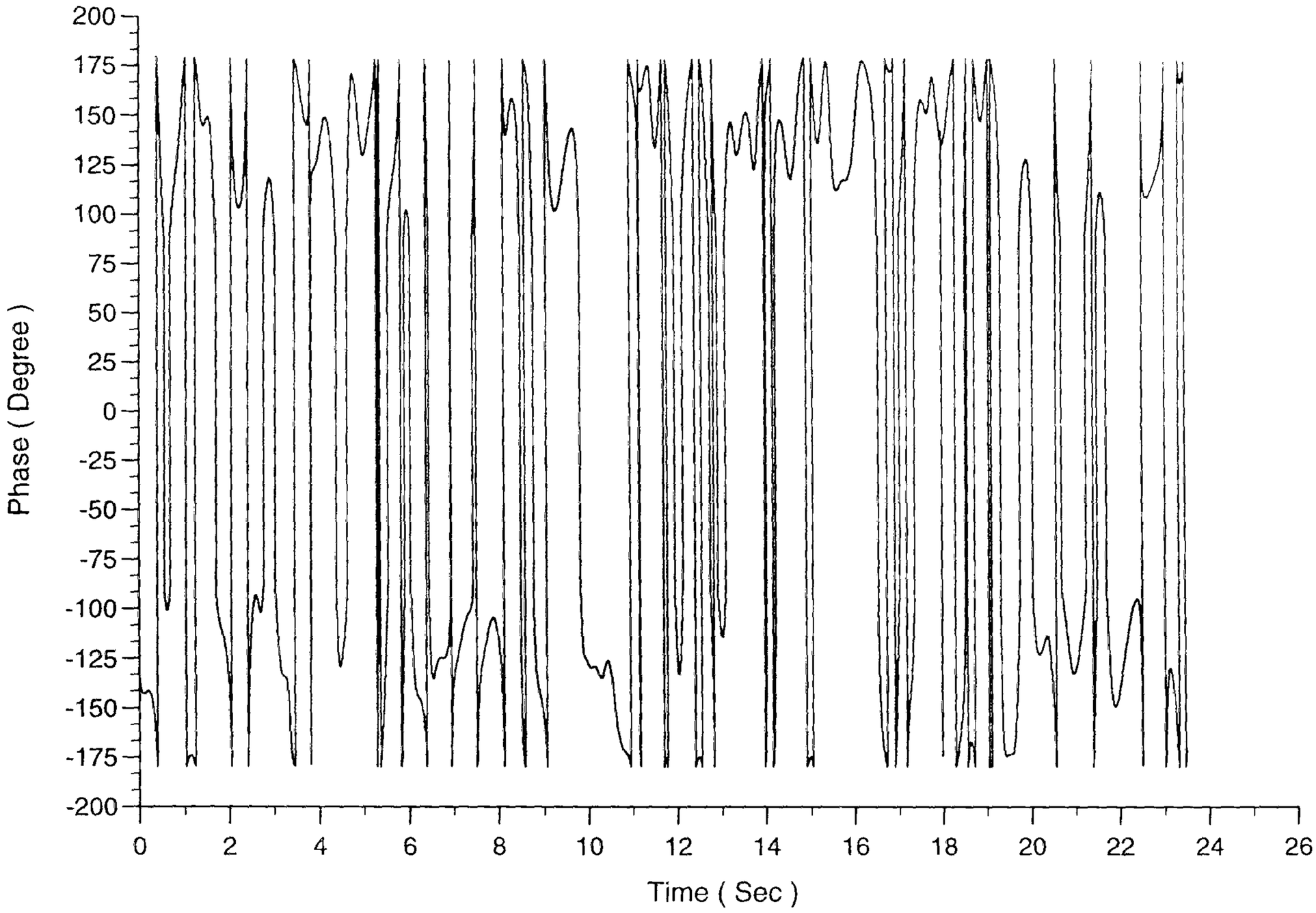


Fig. 3.6.5 Phase Variation of Sampled Impulse Response of Channel 1.

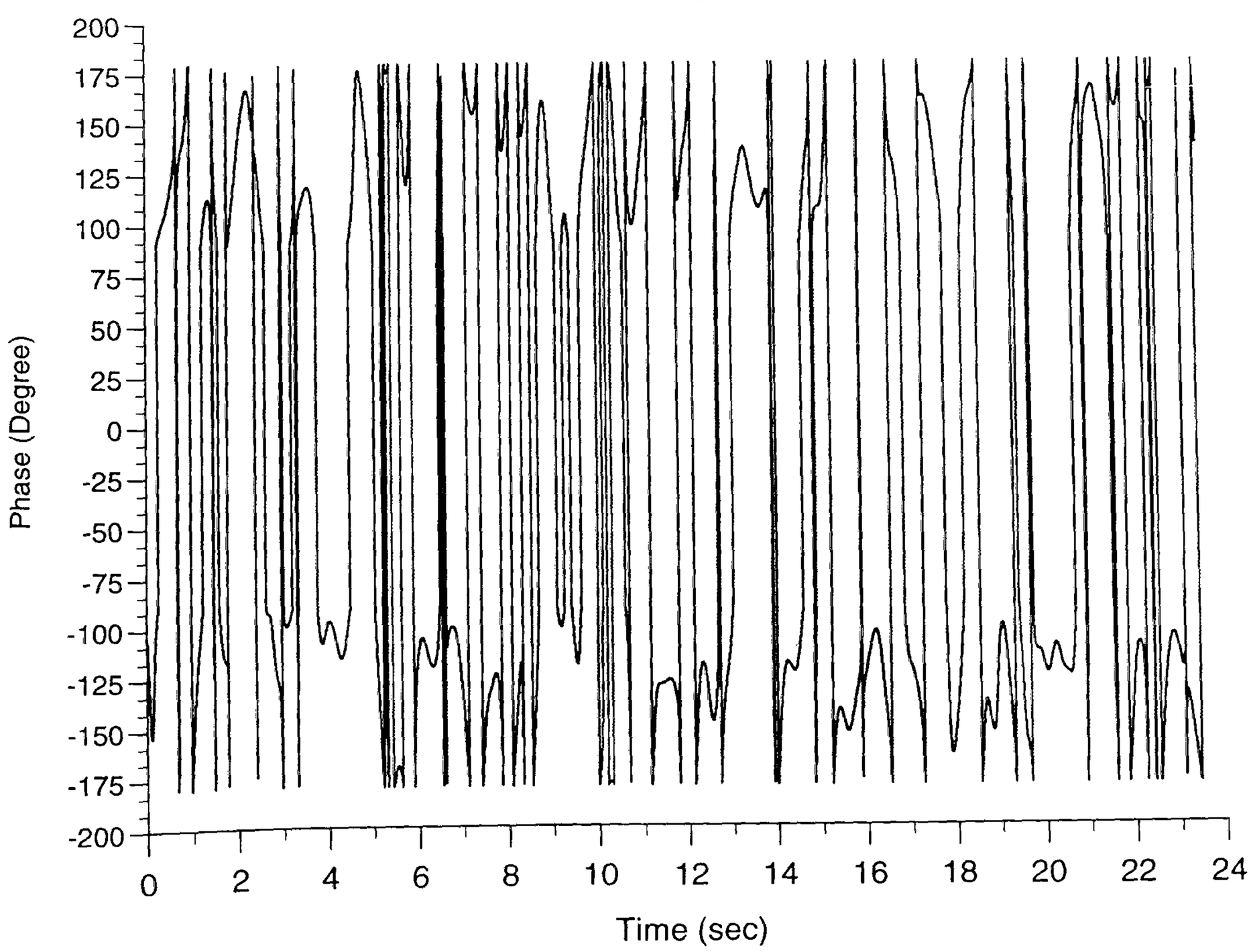


Fig. 3.6.6 Phase Variation of Sampled Impulse Response of Channel 2.

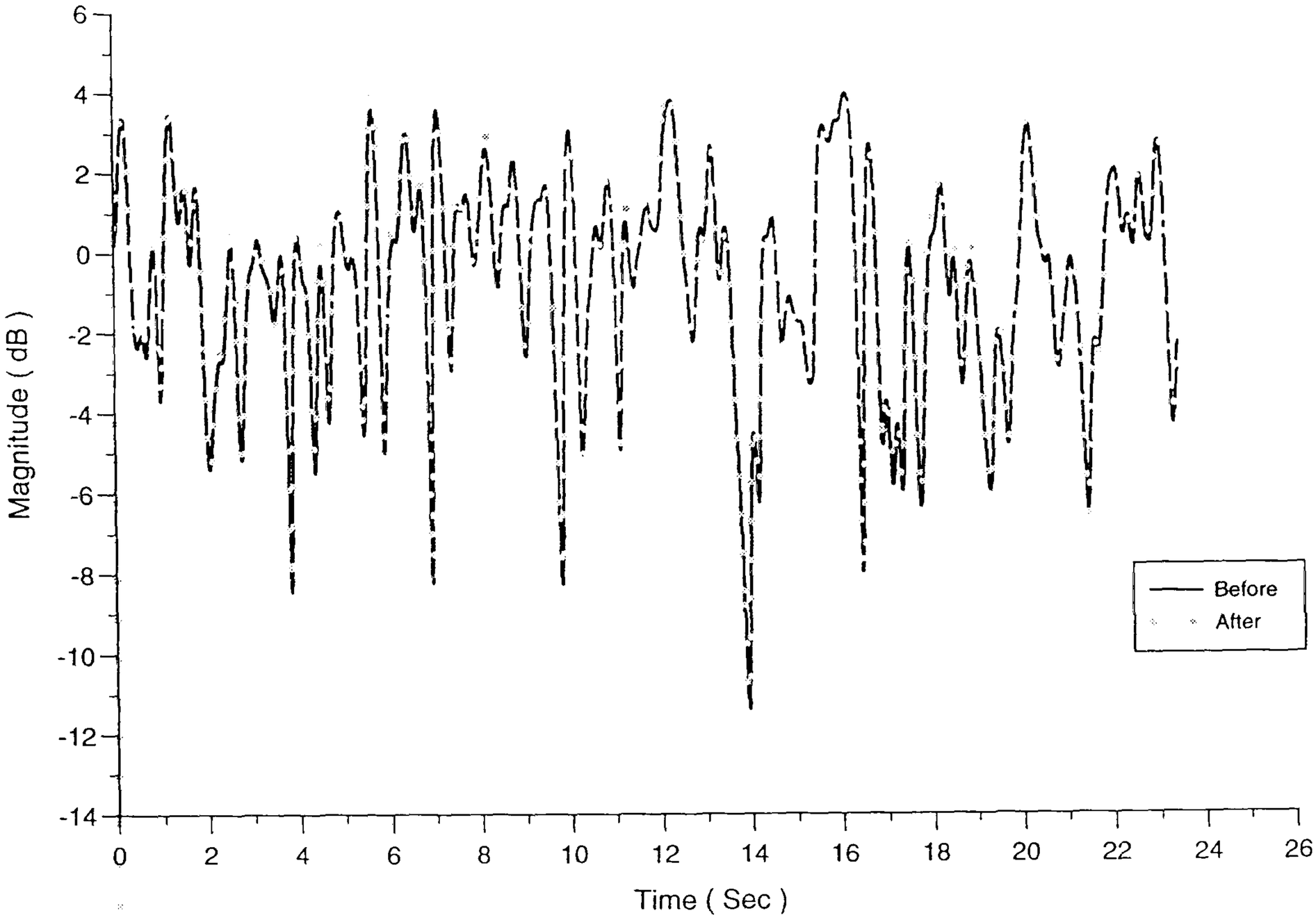


Fig. 3.6.7 Magnitude Variation of the SIR of Channel 1 before and after Convolution.

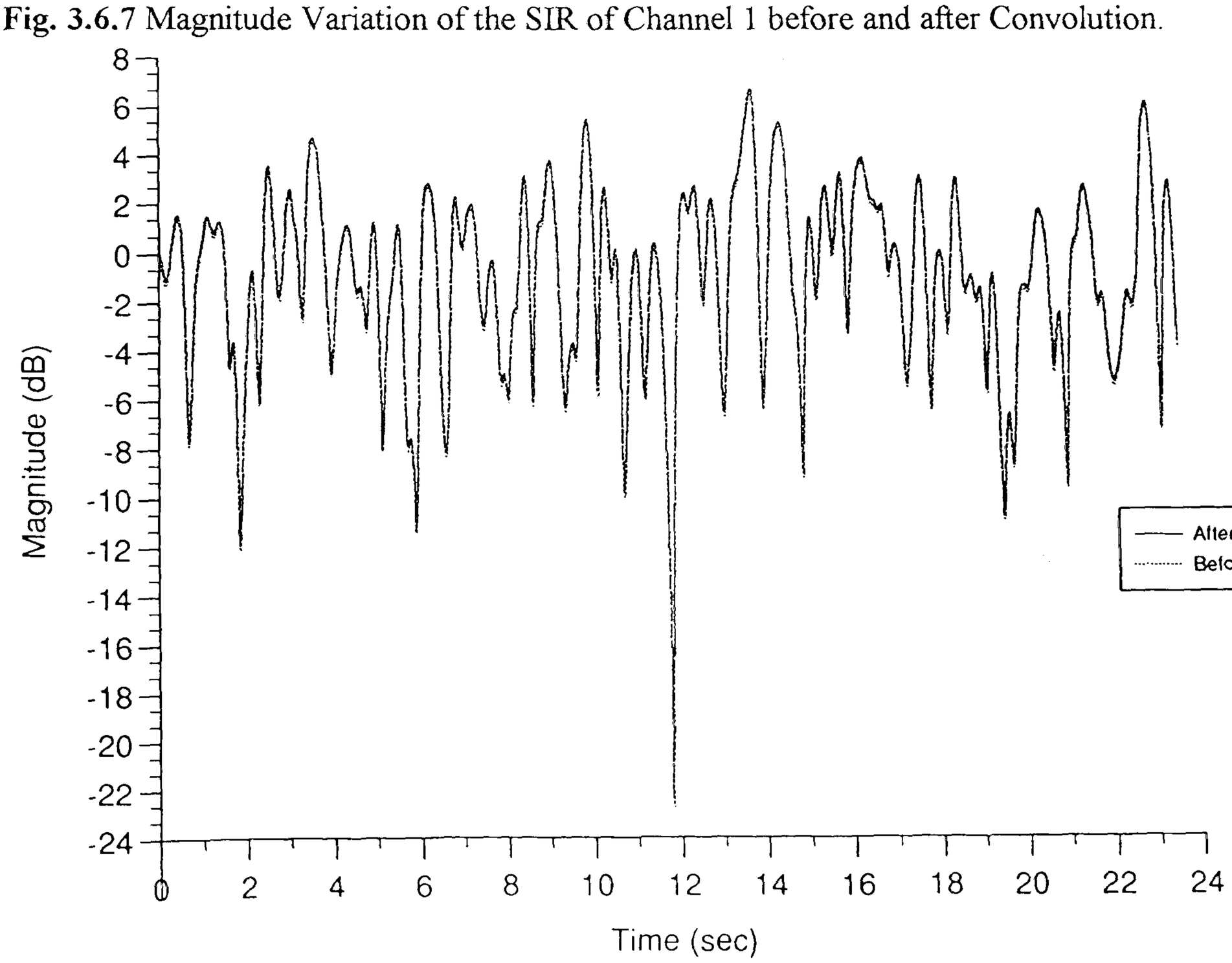
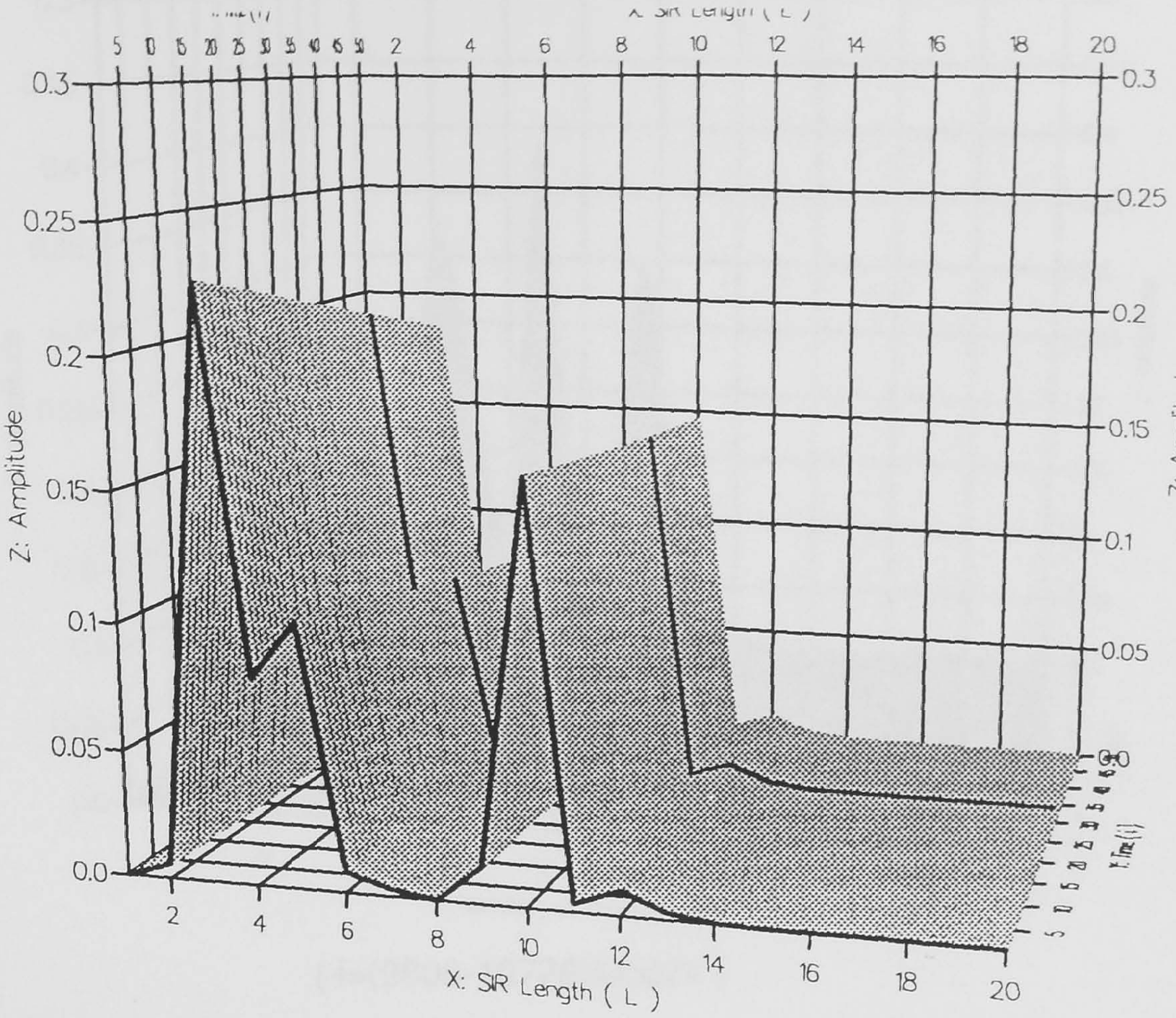
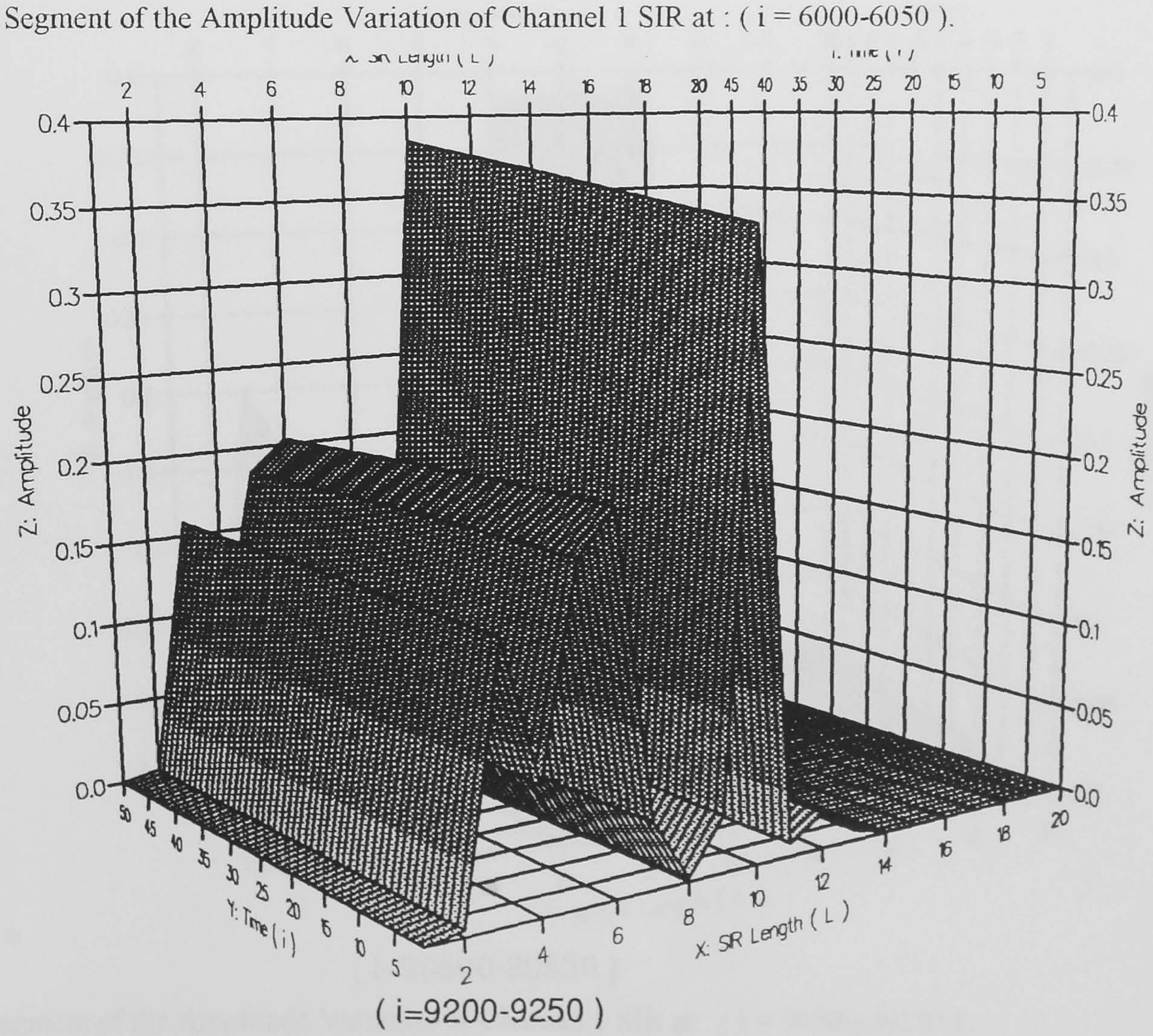


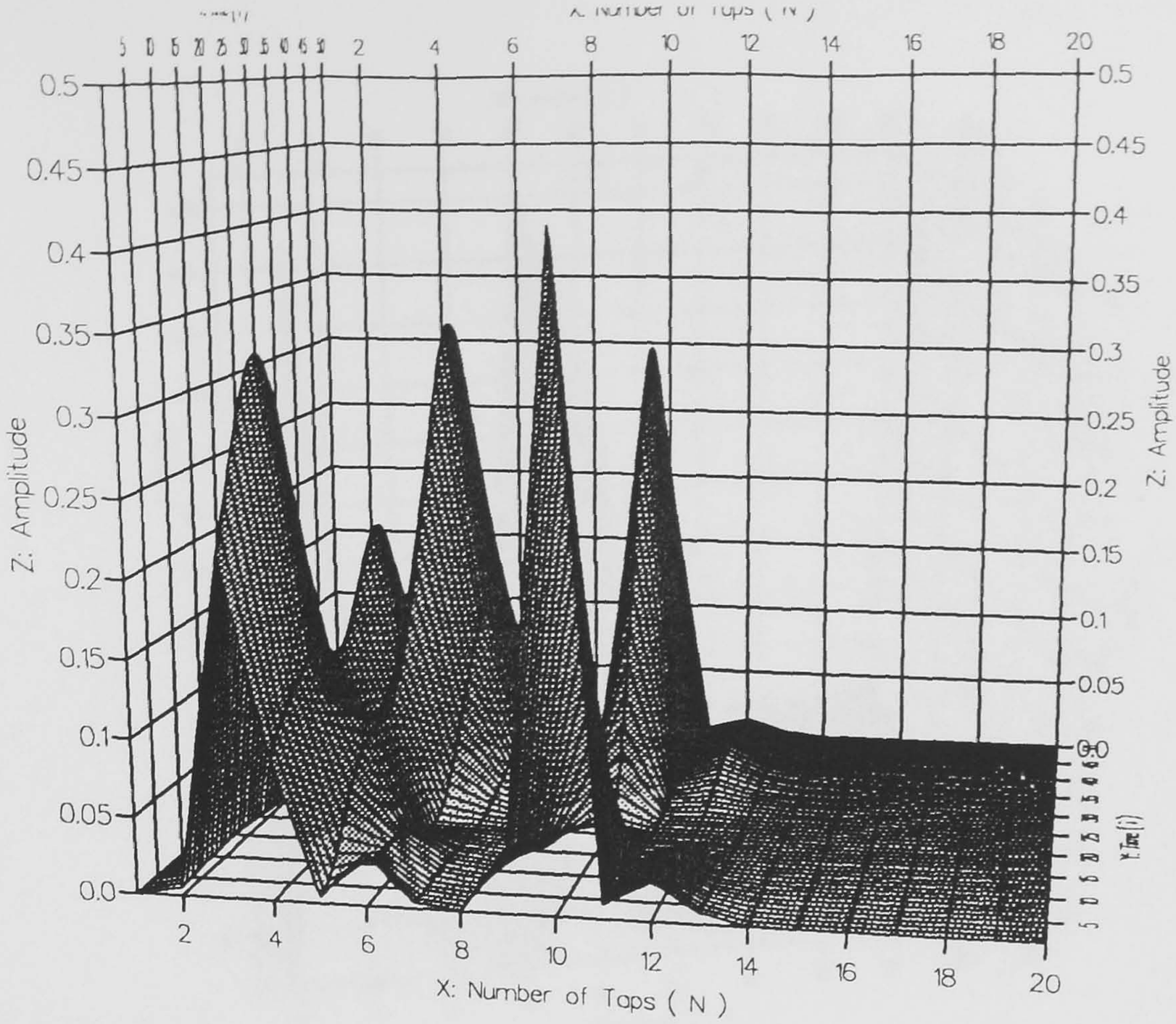
Fig. 3.6.8 Magnitude Variation of the SIR of Channel 2 before and after Convolution.



a : A Segment of the Amplitude Variation of Channel 1 SIR at : (i = 6000-6050).

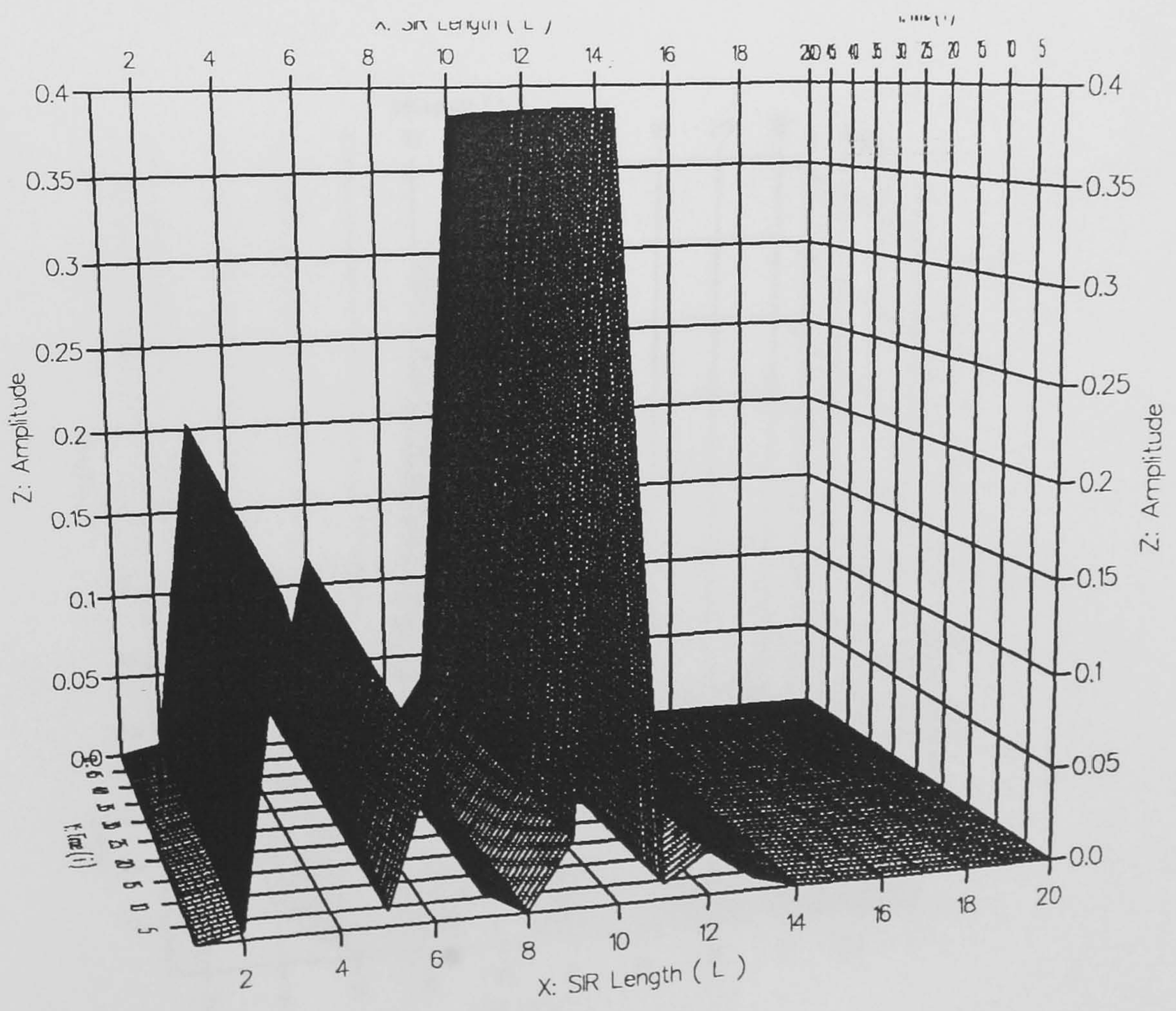


b : A Segment of the Amplitude Variation of Channel 1 SIR at : (i = 9200-9250).



(i=(9600-10776)Mod24)

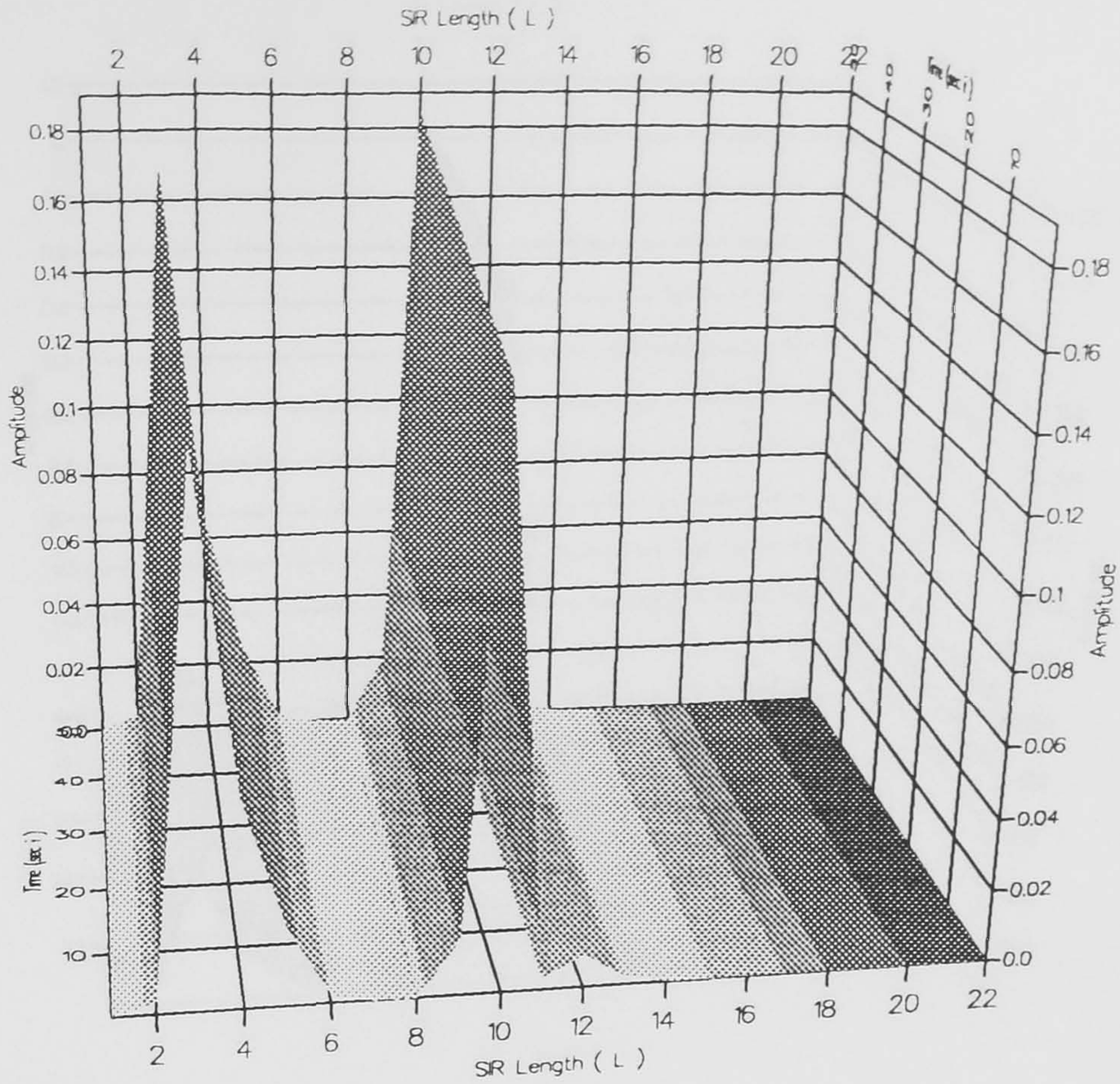
c: A Segment of the Amplitude Variation of Channel 1 SIR at : (i = (9600-10776)mod24).



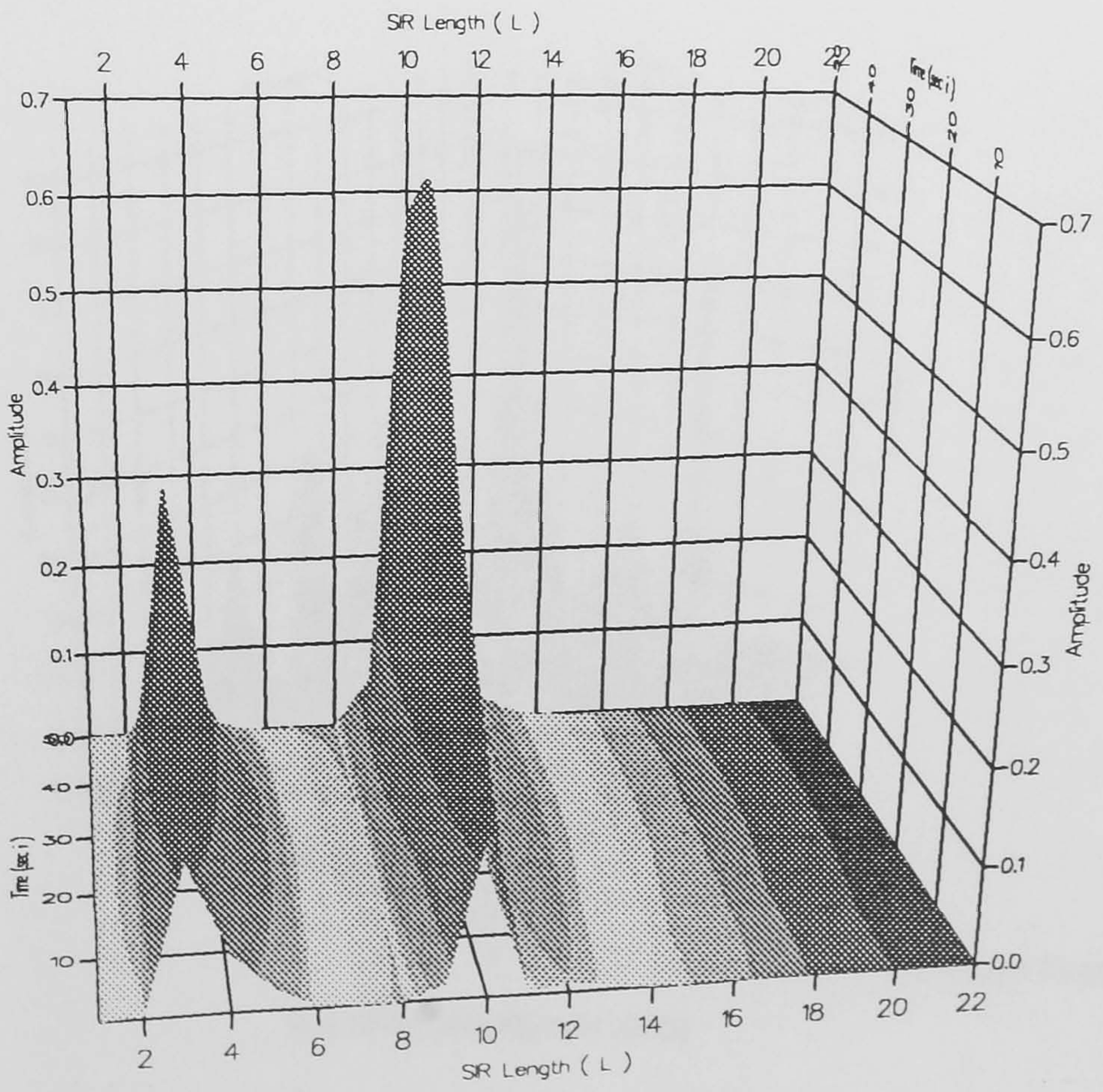
(i=30500-30550)

d : A Segment of the Amplitude Variation of Channel 1 SIR at : (i = 30500-30550).

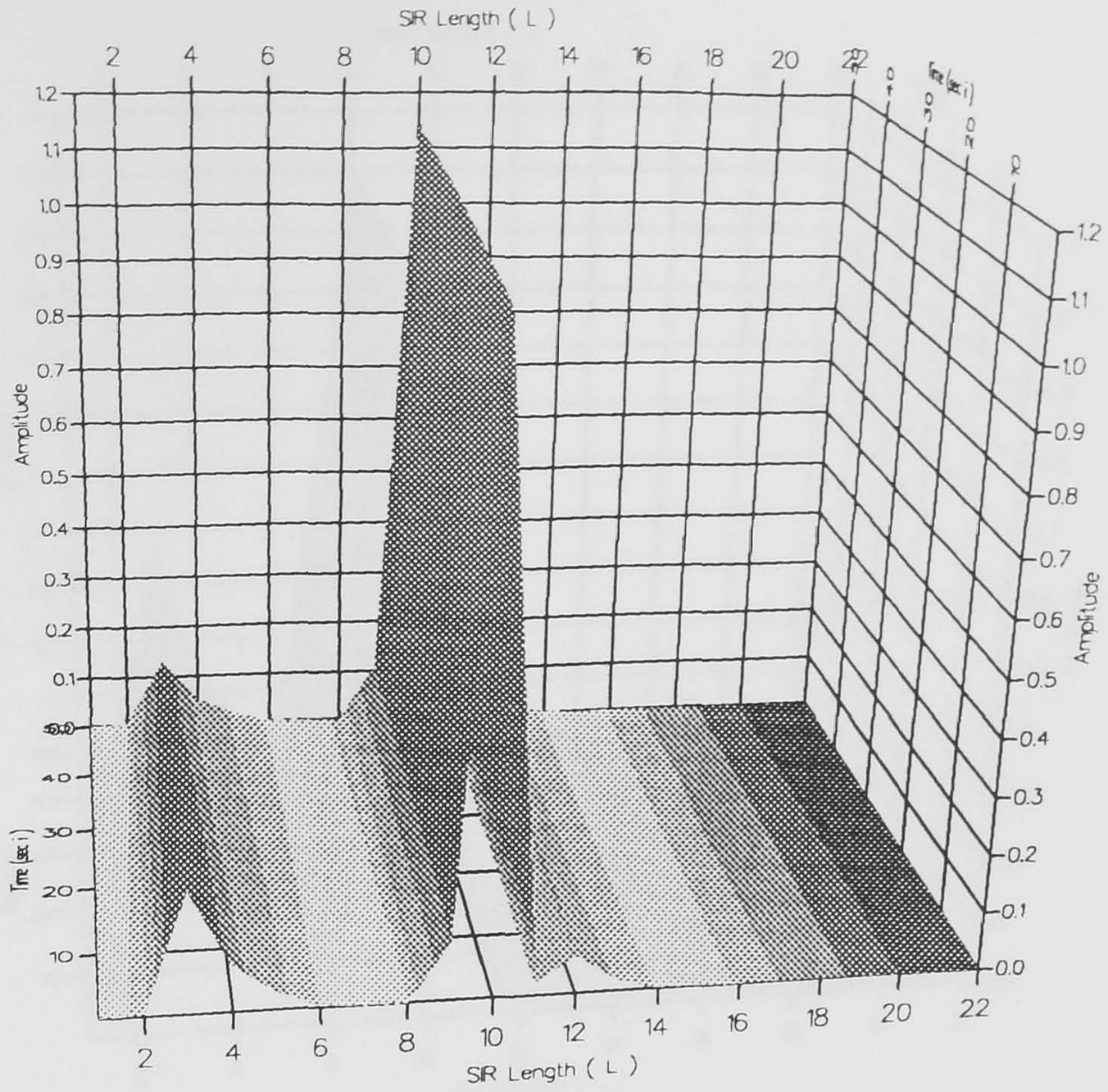
Fig. 3.6.9: Segments of Amplitude Variation of Channel 1 SIR.



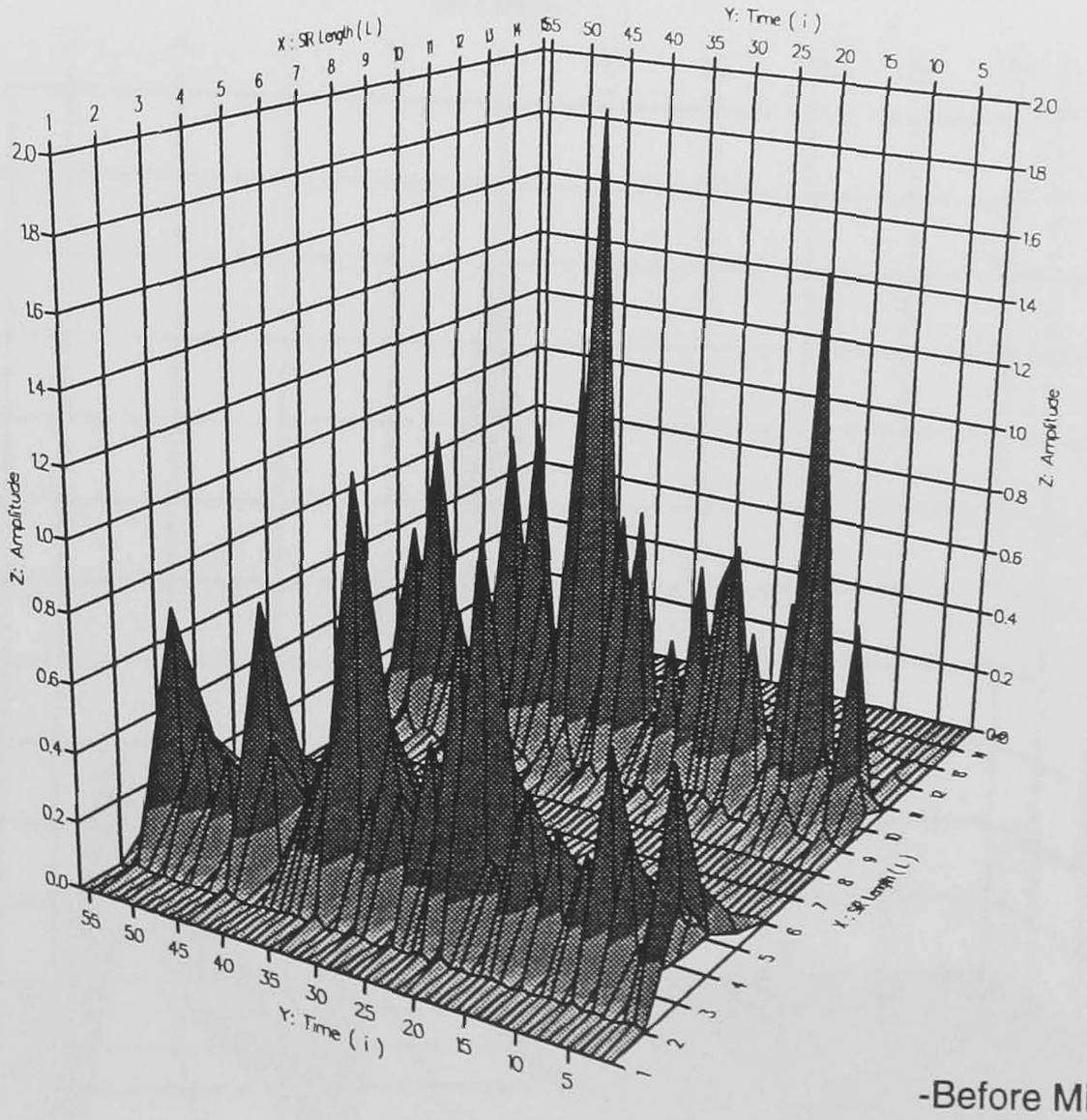
a : A Segment of the Amplitude Variation of Channel 2 SIR at : (i = 9600-9650).



b : A Segment of the Amplitude Variation of Channel 2 SIR at : (i = (9600-10776)mod 24).



c : A Segment of the Amplitude Variation of Channel 2 SIR at : (i = 30600-30650).

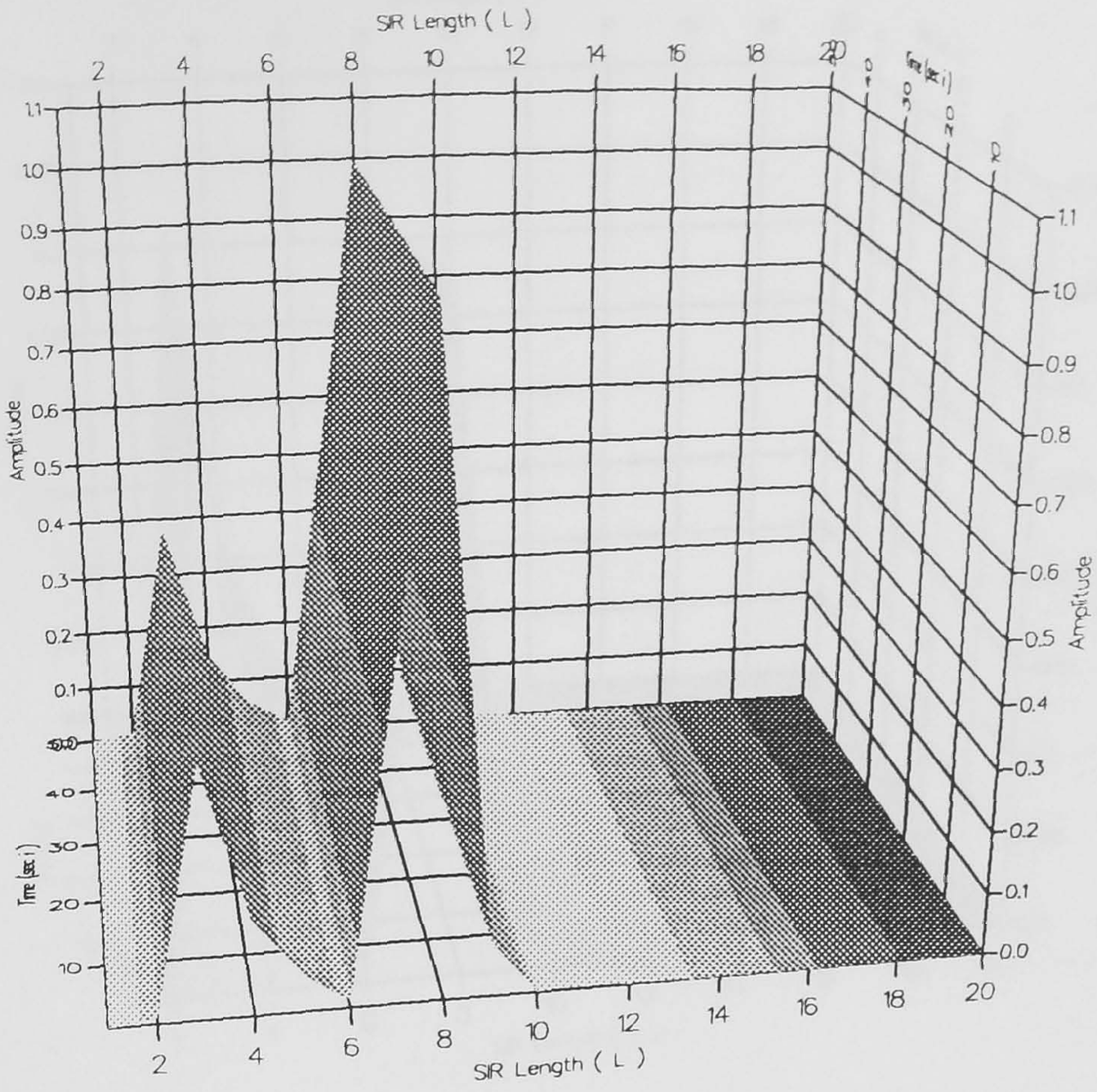


-Before Minimum Phase.

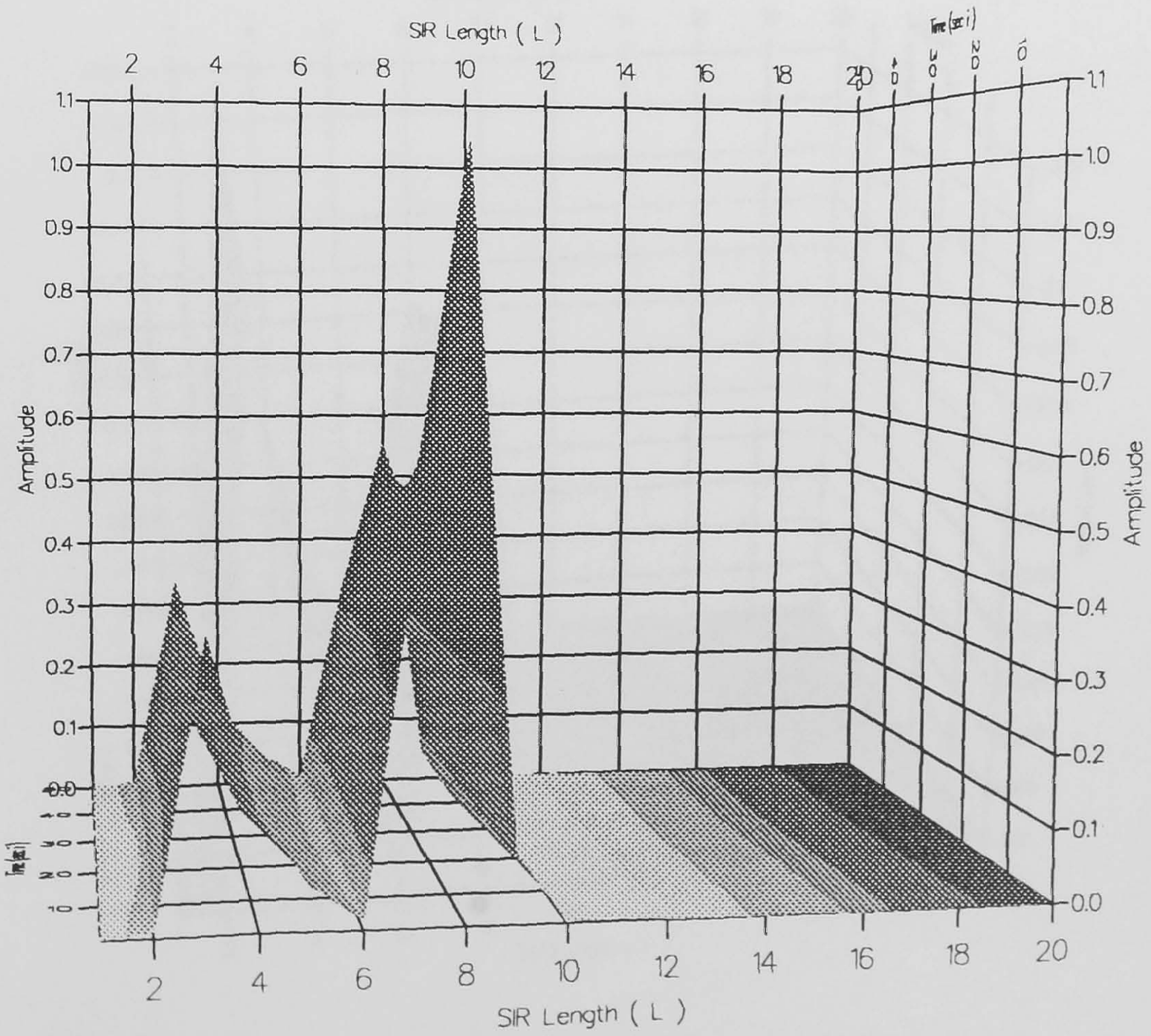
(i=(1000-56000)mod1000)

d : A Segment of the Amplitude Variation of Channel 2 SIR at : (i = (1000-56000)mod 1000).

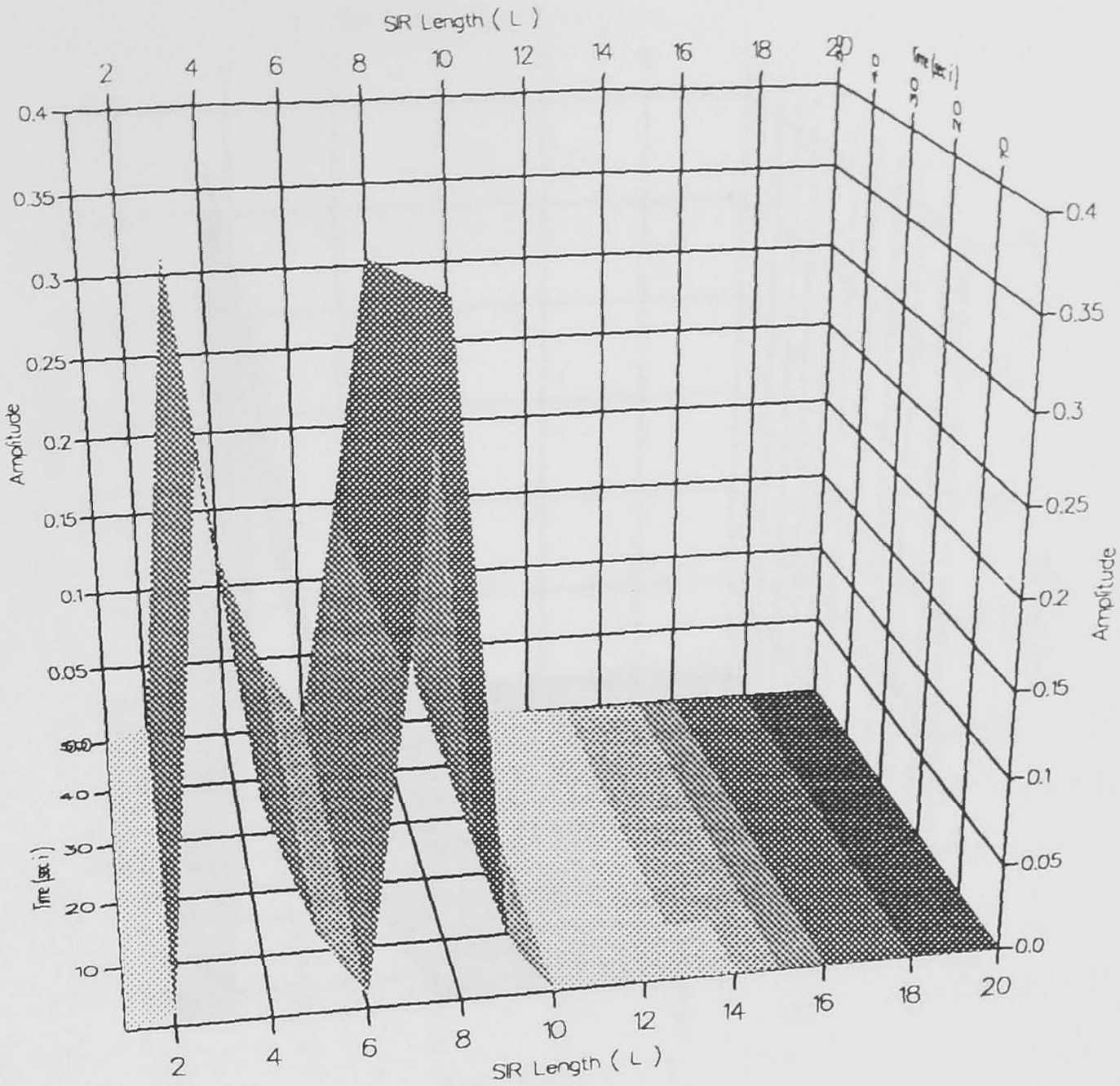
Fig. 3.6.10: Segments of Amplitude Variation of Channel 2 SIR.



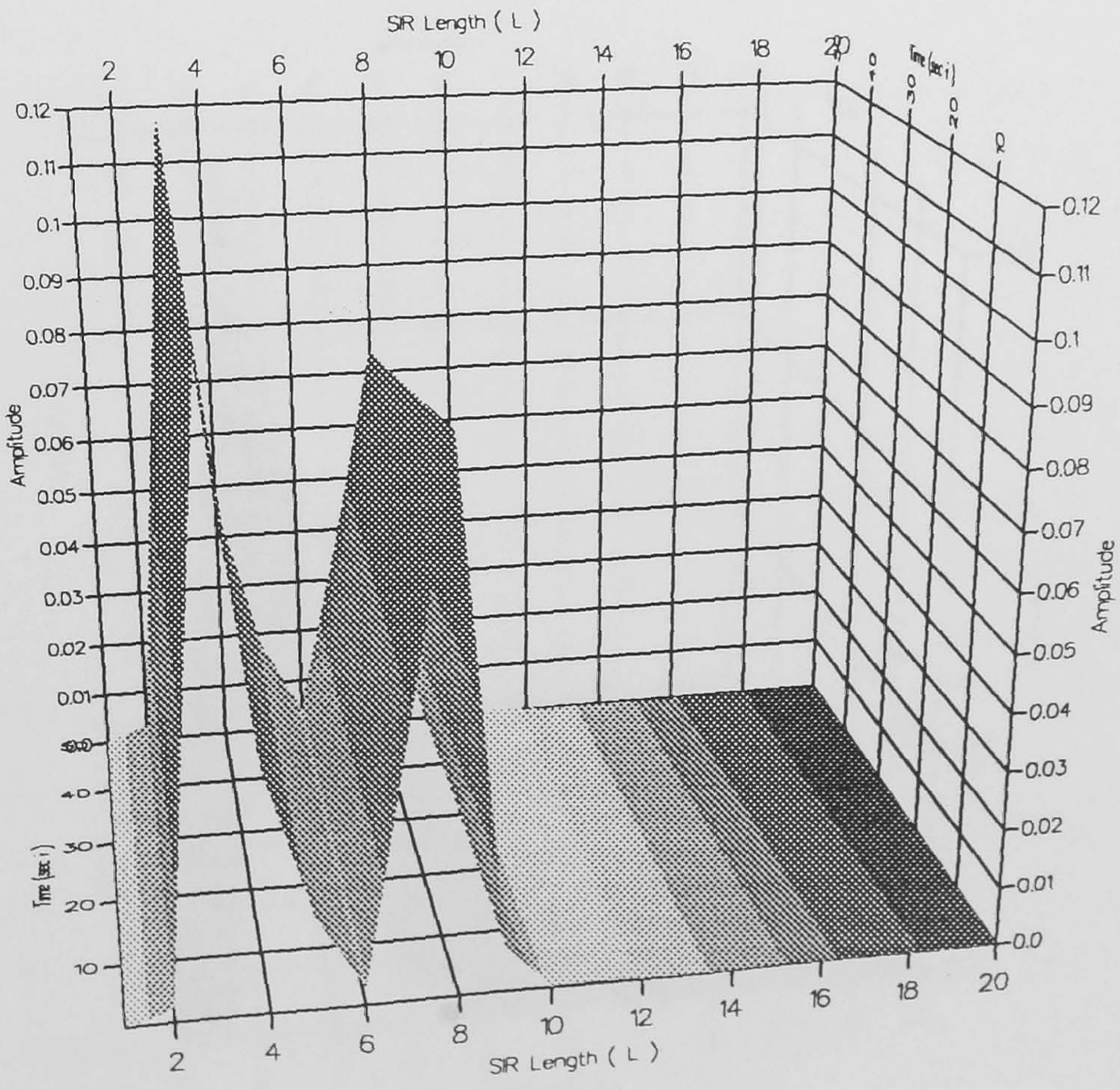
a : A Segment of the Amplitude Variation of Channel 3 SIR at : (i = 9600-9650).



b : A Segment of the Amplitude Variation of Channel 3 SIR at : (i = (9600-10776)mod 24).

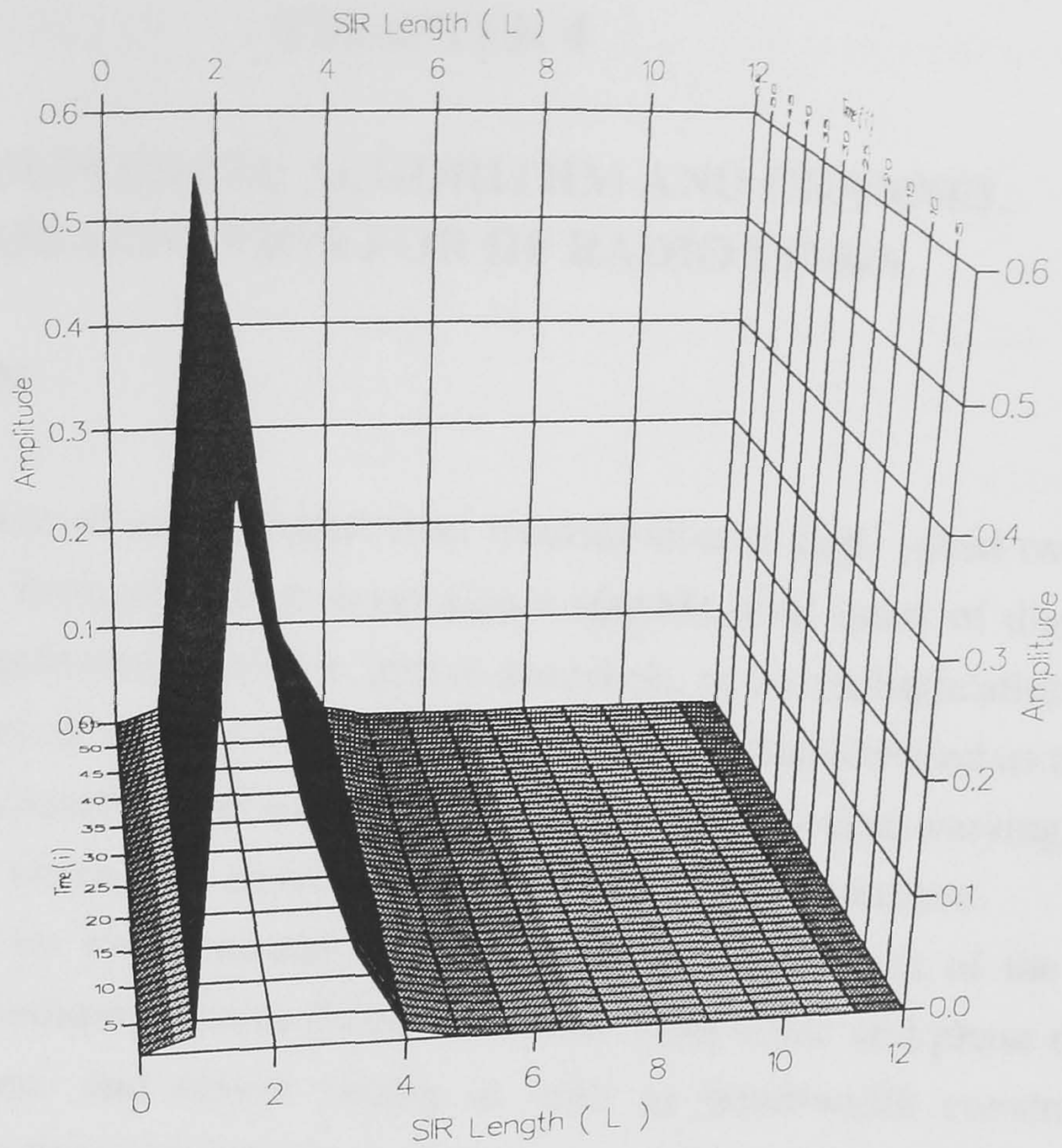


c : A Segment of the Amplitude Variation of Channel 3 SIR at : (i = 30600-30650).

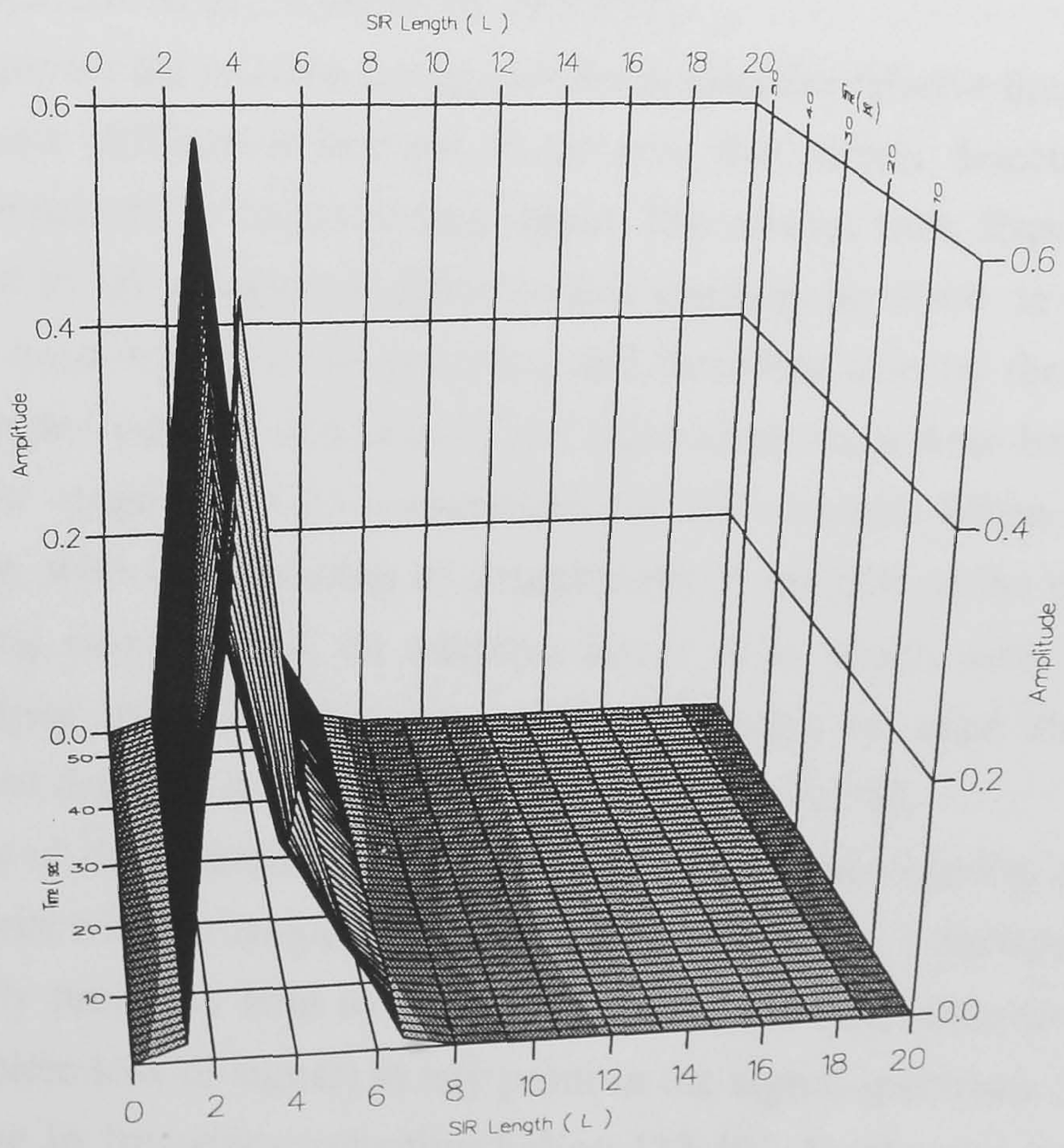


d : A Segment of the Amplitude Variation of Channel 3 SIR at : (i=53500-53550).

Fig. 3.6.11: Segments of Amplitude Variation of Channel 3 SIR.



a : A Segment of the Amplitude Variation of Channel 4 SIR at : (i = 6000-6500).



b : A Segment of the Amplitude Variation of Channel 4 SIR at : (i = (24600-25776)mod 24).

Fig. 3.6.12: Segments of Amplitude Variation of Channel 4 SIR.

CHAPTER 4

MINIMUM PHASE ALGORITHM AND CHANNEL EQUALISATION FOR HF RADIO LINKS

4.1 INTRODUCTION

In the application of serial digital data transmission at high speed over a band-limited channel in the form of a multi-level signal (QAM), four types of distortion can occur : attenuation (amplitude) distortion, phase distortion, noise multiplication and noise addition. In the application of interest, the HF channel can be characterised as a multipath time-varying channel causing time and frequency spreads. This time varying multipath characteristic produces alternately destructive and constructive interference.

The major limitations on high-data-rate HF transmission are a result of the non-ideal characteristics of the medium, such as linear distortion (amplitude and phase distortion), rapid channel variations, and severe fading as well as band-width constraints. The resulting fading can produce complete loss of transmitted signal. The detection of such a signal is thus, adversely affected and as a result, the HF channel, usually requires sophisticated techniques like adaptive linear or decision feed-back equalisation at the receiver, to correct for the inter-symbol interference (ISI) (to remove the amplitude and phase distortion) [1,23,25,26,30,31,39,40,41,48,49,53,57].

Moreover the greater the relative fading rate the greater the relative time dispersion of data signal, the more difficult it become to achieve the correct detection (correct equalisation of the channel) of the received data signal. The relative time dispersion of the data signal is measured by the number of different data symbols involved in a sample of the demodulated base-band signal at the receiver, and therefore also by the number of components in the sampled impulse response of the equivalent linear base-band channel. The latter can be in the range of *15-25* components for HF channel. When the relative time-dispersion is large, with large number of components in the SIR of the channel, but when the relative fading rate is small, an adaptive linear filter which uses a minimum phase algorithm to adjust its coefficient, can, with advantage, be used ahead of the detector, to simplify both detector and channel estimator [23,39,40,58].

The more recent of the conventional approaches to such time-varying distortion is to employ a serial system with an adaptive equaliser at the receiver, a decision feedback equaliser being normally preferred here to a linear equaliser [40,48], since the latter can not handle a null (complete loss of signal) at any point in the signal spectrum, such as can occur from time to time in frequency selective fading [23,40]. Further more, the linear equaliser enhances the noise at frequencies where relatively high attenuation is introduced

by the data link [40]. Therefore, when the channel varies with time, there may now be difficulties in holding the linear equaliser correctly adjusted [23,39,40].

Two types of equaliser-detector combinations are commonly used in a conventional serial transmission system over a fading channel. Firstly a non-linear (decision feedback) equaliser with threshold detector, which employs a feed-forward transversal filter and a feedback transversal filter, and secondly a linear feed-forward transversal filter followed by a near maximum likelihood (NML) detector. In both systems the linear feed-forward section ideally attempts to correct only the phase distortion introduced by the transmission path. With severe amplitude distortion, a decision feedback equaliser is a substantially sub-optimum detector, whereas the NML detector operates to provide near optimum detection [23,29,43]. An important property of HF channel is that its impulse response varies smoothly with time, such that there are no discontinuities in the rate of change or acceleration with time of a sample of the impulse response. Also the first component of the SIR need not be the largest component, which is necessary when using a NML detector [25,43,58]. Therefore the HF channel is a non-minimum phased channel.

From the above discussion, for most reliable operation of the receiver (NML detector or else non-linear equaliser), the feed-forward transversal filter must be adjusted to act as an all-pass network [23,39,58], which gives a resultant minimum phase response. The feed-forward section is normally adjusted adaptively, by means of the gradient algorithm [1,23,50,63,96], which is simple and quite adequate for applications where the distortion (ISI) is not severe and does not vary much with time, which is not the case considered in this study. Therefore satisfactory operation is no longer achieved with the gradient method over an HF channel. The gradient algorithm is best replaced by the Kalman algorithm [48,63,69,71], which is however, considerably more complex even in its simplified forms. The latter also suffers from the accumulation of round-off errors [27,48,78].

An alternative approach for adjusting the coefficient of the feed-forward filter is to use a minimum phase algorithm (root finding algorithm). First, replace all zeroes (roots) of the Z-transform of the sampled impulse response of the channel that have absolute values greater than unity (outside the unit circle in Z-plane), by their complex conjugate reciprocals, the remaining roots being left unchanged. Secondly, adjust the tap-gains of the filter. This algorithm was developed [29,40,39] for the telephone channel. A simple iterative process determines both the required tap-gains of the adaptive linear filter that precedes the NML detector and also the resultant SIR of the channel and filter which is necessary for NML detection.

This chapter describes first the principles of linear and non-linear equalisers. It describes next the algorithm of minimum phase and the adjustment of the linear feed-forward filter when operating over an HF channel. Finally computer simulation results over four HF channel models are presented.

4.2 SYSTEM MODEL

Fig.4.2.1 shows the model of a data transmission system, using synchronous serial digital data, four-level QAM modulation (QPSK), a carrier frequency of *1800Hz* and signal-element rate of *2400* baud. This system is represented as a linear base-band channel having a transmitter filter and linear QPSK modulator, a transmission path and a receiver filter with linear QPSK demodulator. The transmission path here, is the HF radio link. Stationary white Gaussian noise with zero mean and two-sided power spectral density of $(N_0/2)$, is added to the data signal at the output of transmission path. The full details with the basic assumptions of this system are described in Chapter 3 and only the main aspects of the system are given here.

The output of the receiver filter is sampled at time instant $\{iT\}$ to give the received sampled $\{r'_i\}$, which are fed to the adaptive linear transversal filter. The tap-gains of the latter are adjusted using a minimum phase algorithm whose input can be the estimated SIR of the channel which can be determined by the channel estimator. The complex-valued sample of demodulated signal $r'(t)$ at $t=iT$ is :

$$r'_i = \sum_{h=0}^{L} s_{i-h} \cdot p'_{i,h} + v_i \qquad 4.2.1$$

where : $P'_i = [p'_{i,0}, p'_{i,1}, \ldots\ldots\ldots, p'_{i,L}]$ 4.2.2

is taken to be the estimated sampled impulse response at $t=iT$ of the linear base-band channel $(L+1)$-components row vector. The $\{s_i\}$ are the data symbols which carry the information to be transmitted and are assumed to be statistically independent and equally likely to have any of their four possible value, and are given by :

$$S_i = [s_{i-1}, s_{i-2}, \ldots\ldots\ldots, s_{i-L}] \qquad 4.2.3$$

The $\{v_i\}$ are noise components, complex-valued Gaussian random variables with zero mean and fixed variance σ_v that is dependent on N_0 and the receiver filter.

The basic assumptions of this test are given as follows :

1. The minimum phase algorithm operates on the actual SIR of the channel P_i instead of the estimated one, or in other words perfect estimation is assumed, such that

$$P_i = P'_{i,i-n-1} \qquad 4.2.4$$

"i.e., minimum phase algorithm has exact knowledge of P_i for all $\{i\}$".

This assumption is necessary here, allowing us to examine the minimum phase algorithm on the actual SIR of the channel, preventing any errors from the estimator from affecting

the performance of the system. In the final chapter, the operation of this algorithm is examined using the estimated SIR of the channel.

2. For practical purposes, it is assumed that ($p_{i,h} = 0$, for $0 > h > L$) [29].

3. The system is assumed to be synchronised.

4. The parameters of each channel under test are given in Chapter 2.

Bearing in mind that for testing the performance of the minimum phase algorithm, its input is the sequence $\{P_{i,h}\}$, for $h=0,1,\ldots\ldots,L$ which is generated from the channel model described in Chapter 2. The output of the algorithm is given by the $(L+1)$-component row vector Y'_i:

$$Y'_i = [y'_{i,0}, y'_{i,1}, \ldots\ldots\ldots, y'_{i,L}] \qquad 4.2.5$$

And after an appropriate scaling (see Fig.4.2.2), then :

$$Y_i = [y_{i,0}, y_{i,1}, \ldots\ldots\ldots, y_{i,L}] \qquad 4.2.6$$

where Y_i is the scaled minimum phase version of the sampled impulse response, which needed by the near maximum likelihood detector.

In this chapter we are not concerned with the techniques for estimating the SIR of the channel, and such techniques will be studied in the following chapter. Therefore the performance of minimum phase algorithm is examined by using the actual SIR of time-varying HF channel instead of the estimated one. Eqn.4.2.1 can be written as :

$$r'_i = s_i \cdot p_{i,0} + \sum_{h=1}^{L} s_{i-h} \cdot p_{i,h} + v_i \qquad 4.2.7$$

The term $(\sum_{h=1}^{L} s_{i-h} \cdot p_{i,h})$ represents inter-symbol interference (ISI), which arises whenever one transmitted signal-element, at the receiver input, does not die away completely before the arrival of the next one. Several techniques are developed to handle the detection of the digital data signals in the presence of ISI. An equaliser [40,49,87,88,96] is employed to remove the ISI from the received signal eqn.4.2.7, such that the signal at the detector input becomes :

$$e_i = s_i + u_i \qquad 4.2.8$$

u_i relates to v_i according to some formula depending on the equaliser.

Therefore a quick look here, will be made at the basic equalisation techniques followed by a detailed description of the method to remove the phase distortion caused by the HF channel by means of the minimum phase algorithm. The received signal after adaptation process at the input to detector can be written now as :

$$r_i = s_i \cdot y_{i,0} + \sum_{h=1}^{L} s_{i-h} \cdot y_{i,h} + u_i \qquad 4.2.9$$

4.3 LINEAR AND NON-LINEAR EQUALISER

The inter-symbol interference can be removed by a linear or non-linear (decision feed-back) equaliser, which is inserted between the sampler and detector [29,40,49,54,55,77]. In the presence of pure phase distortion and additive white Gaussian noise, a linear equaliser followed by a detector is the optimum detection process [29,40,49,77]. In the presence of any amplitude distortion an improved tolerance to noise is normally achieved by using a non-linear equaliser in place of a linear equaliser [29,57]. Hence the basic principle of both techniques will be explained starting first with the linear equaliser.

4.3.1 LINEAR EQUALISER

Fig.4.3.1 shows the linear feed-forward transversal equaliser. The equaliser is a taped delay-line (non-recursive). In such an equaliser, the current and past values r'_i of the received signal are linearly weighted by equaliser coefficients (tap-gains) and assumed to produce the output. From Fig.4.3.1 the received samples at the symbol rate is stored in a digital shift register (or memory), and the equaliser output samples (sums of products r_i) are computed digitally, once per symbol according to :

$$r_i = \sum_{h=0}^{k} r'_{i-h} \cdot c_h \qquad 4.3.1$$

where : $C = [c_0, c_1, \ldots\ldots\ldots, c_k]$ 4.3.2

C is the equaliser coefficients $(k+1)$-components row vector, or the sampled impulse response (SIR) of the equaliser, whose Z-transform is given by :

$$C(z) = c_0 + c_1 z^{-1} + \ldots\ldots\ldots + c_k \cdot z^{-k} \qquad 4.3.3$$

Consider the Z-transform of the received sequence in the absence of noise, [40] as :

$$R'(z) = S(z) \cdot P(z) \qquad 4.3.4$$

Where : $S(z) = \sum_{i=1}^{m1} s_i \cdot z^{-i}$ 4.3.5a

$$R'(z) = \sum_{i=1}^{m2} r'_i \cdot z^{-i} \qquad 4.3.5b$$

and $P(z) = \sum_{h=0}^{L} p_{i,h} \cdot z^{-i}$ 4.3.5c

S_{m1} & R'_{m2} are row vectors such that :

$$S_{m1} = [s1, s2, \ldots\ldots\ldots, s_{m1}] \qquad 4.3.6$$

$$R'_{m2} = [r'1, r'2, \ldots\ldots\ldots, r'_{m2}] \qquad 4.3.7$$

S_{m1} represents the transmitted data sequence, whereas the vector R'_{m2} represents the received sequence. $P(z)$ is the Z-transform of the sampled impulse response of the linear base-band channel. If the number of the tap-gains of a linear equaliser increase without bound, we would obtain an infinite length equaliser with zero ISI at its output [40]. For the accurate equalisation of the channel, the Z-transform of the channel and equaliser can be written as :

$$C(z) \cdot P(z) = z^{-h} \qquad 4.3.8$$

$$\Rightarrow C(z) = P^{-1}(z) \cdot z^{-h} \qquad 4.3.9$$

Thus the ISI can be removed by passing the received sequence through a linear equaliser with Z-transform $P^{-1}(Z)$. This gives :

$$R'(z) \cdot P^{-1}(z) \cdot z^{-h} = S(z) \cdot z^{-h} \qquad 4.3.10$$

Where h is a non-negative integer in the range 0 to $(k+L)$. The factor z^{-h} allows for the delay likely to be introduced by the linear equaliser, and has no effect, apart from the delay, on the signal detection. Approximate equalisation of the channel with Z-transform $P(z)$ can be achieved, whether the roots of $P(z)$ lie inside or outside, but not on, the unit circle in Z-plane [40]. The Z-transform of the signal at the output of the equaliser is given by :

$$R(z) = R'(z) \cdot C(z) \qquad 4.3.11$$

Substitute eqn.4.3.4 into eqn.4.3.11, then :

$$R(z) = S(z) \cdot P(z) \cdot C(z) = S(z) \cdot Y'(z) \qquad 4.3.12$$

Where : $Y' = [y'_0, y'_1, \ldots\ldots\ldots, y'_{k+L}]$ 4.3.13

is the sampled impulse response of the channel and linear equaliser, whose Z-transform $Y'(z)$ is given by :

$$Y'(z) = P(z) \cdot C(z) \qquad 4.3.14$$

Y' must have only one non-zero component with value 1.0, when exact equalisation is achieved. Let that component be y'_h. This means that eqn.4.3.9 is held. The number of taps in the equaliser $(k+1)$, is in general infinite [40], but in practice must be restricted to a

finite value, and consequently $C(z)$ can not satisfy eqn.4.3.9, but it can be chosen to do so approximately. Thus, eqn.4.3.9 becomes :

$$C(z) \approx P^{-1}(z) \cdot z^{-h} \qquad 4.3.15$$

As a result Y' in eqn.4.3.13 may have components with non-zero amplitude beside the main component y'_h, which represents the residual distortion in the equalised signal. Various techniques are known to determine $C(z)$, i.e. the tap-gains of the linear equaliser, and different criteria are used to minimise the residual distortion in the equalised signal. The details of such studies are beyond the scope of this chapter and are given in [1,29,40,63,52]. From eqn.4.3.12, 4.3.14 and 4.3.15, then :

$$R(z) \approx S(z) \cdot z^{-h} \qquad 4.3.16$$

which gives at time $t=(i+h)T$, at the input of the detector (in the absence of noise) :

$$R_{i+h} \approx S_i \qquad 4.3.17$$

In the presence of noise, from eqn.4.2.1(or 4.2.7) and 4.3.1, the noise components in r_i is v_i, where :

$$u_i = \sum_{j=0}^{k} v_{i-j} c_j \qquad 4.3.18$$

Therefore, in the presence of noise eqn.4.3.17 becomes :

$$R_{i+h} \approx S_i + U_{i+h} \qquad 4.3.19$$

Since $\{v_i\}$ in eqn.4.3.18 are statistically independent Gaussian random variables with zero mean and variance σ_v^2, the $\{u_i\}$ are Gaussian random variables with zero mean and variance η^2 [40] given by :

$$\eta^2 = \sigma_v^2 \cdot \sum_{j=0}^{k} |c_j|^2 = \sigma_v^2 \cdot |C|^2 \qquad 4.3.20$$

Where $|c_j|$ is the modulus of the complex-valued quantity c_j. From eqn.4.3.20 can be seen that the linear equaliser enhances the noise if $|C|^2 > 1$, the latter being the case when the channel introduces amplitude distortion into the transmitted signal [1,40]. This is undesirable, particularly for radio channels, which may be subject to frequency selective fades. Since $P(z)$ may contain one or more zeros on (very close to) the unit circle, the channel introduces infinite attenuation at certain discrete frequencies, which means that the corresponding frequency components of the transmitted signal are lost completely. Therefore the linear equalisation of the channel in fact requires a linear filter that

introduces infinite gain at these frequencies, which is not physically realisable. In this case of a linear feed-forward transversal equaliser, the infinite gain needed must be achieved through an infinite number of taps [40]. The channel can not therefore be equalised linearly, no matter what type of linear filter is used, if exact equalisation is required.

The detector in Fig.4.3.1 operates on the sample value given by eqn.4.3.19 to detect the value of the data symbol s_i. An error occurs in the detection of s_i whenever the noise component u_{i+h} is such that R_{i+h} is closer to a possible value of s_i different from its correct value, that is, whenever the noise component u_{i+h} carries R_{i+h} onto the opposite side of a decision threshold with respect to the transmitted data element s_i [15].

4.3.2 NON-LINEAR EQUALISER

A simple non-linear equaliser [1,29,40,49,52,54,56,77], which is particularly useful for channels with severe amplitude distortion, uses decision feed-back to remove the interference from symbols which have already been detected. Fig.4.3.2 shows the non-linear (decision feed-back) equaliser. The equalised signal ε_{i+n} is the sum of the outputs of the forward and feedback parts of the equaliser. The forward part is like the linear transversal equaliser discussed earlier. Decisions made on the equalised signal are fed-back through a second transversal filter. The aim of the latter is to remove the ISI. The basic idea is that if the values of the symbols already detected are known (past decision are assumed to be correct), then the ISI contributed by these symbols can be cancelled exactly by subtracting past symbol values with suitable weighting from the equaliser output. The weights can be the samples of the tail of the linear base-band channel impulse response. The linear feed forward transversal filter in Fig.4.3.2 has $(n+1)$-taps and a sampled impulse response :

$$D = [d_0, d_1, \ldots\ldots\ldots, d_n] \qquad 4.3.21$$

with Z-transform :

$$D(z) = \sum_{j=0}^{n} d_j \cdot z^{-j} \qquad 4.3.22$$

This filter is an all pass network, which therefore does not change any amplitude distortion introduced by the channel, but affects only the phase distortion [29,39], i.e. its objective is to remove any phase distortion introduced by the channel. The latter can be translated using Z-transform analysis by having some roots of the SIR of the channel outside the unit circle. The filter is also such that the SIR of the channel and filter in cascade is minimum phase and has a first component of value close to unity [40,39].
The filter introduces a delay of n-sampling intervals. Also let :

$$D(z) = C(z) \cdot Y'(z) \qquad 4.3.23$$

where $Y'(z) = \sum_{j=0}^{L} y'_j \cdot z^{-j}$ 4.3.24

and $C(z)$ is the Z-transform of the k-taps linear equaliser for the base-band channel with Z-transform $P(z)$, such that :

$$C(z) = P^{-1}(z) \cdot z^{-n} \qquad 4.3.25$$

The Z-transform of the channel and linear filter D is :

$$D(z) \cdot P(z) = C(z) \cdot Y'(z) \cdot P(z) \approx Y'(z) \cdot z^{-n} \qquad 4.3.26$$

So that, the sampled impulse response of the channel and filter D is approximately :

$$Y' \approx [1, y'_1, y'_2, \ldots\ldots\ldots, y'_L] \qquad 4.3.27$$

assuming correct equalisation, and neglecting the delay of n sampling intervals. As n increases the approximation improves quite rapidly [29]. The vector Y' has the same number of components as P but it is approximately minimum phase, which means that the energy of the sequence is concentrated into its earliest component [39,40]. The signal sample at the output of the filter D is given by :

$$r_{i+n} = \sum_{j=0}^{n} r'_{i+n-j} \cdot d_j \qquad 4.3.28$$

From eqns.4.2.7, 4.3.23, 4.3.25 and 4.3.26, assuming correct equalisation ($y'_0 \approx 1$), eqn.4.3.28 can be written as :

$$r_{i+n} = s_i + \sum_{j=1}^{L} s_{i-j} \cdot y'_j + u_{i+n} = s_i + s_{i-1} \cdot y'_1 + \ldots\ldots + s_{i-L} \cdot y'_L + u_{i+n} \qquad 4.3.29$$

where u_{i+n} is the noise component at the output of linear filter, which is given by :

$$u_{i+n} = \sum_{j=0}^{n} v_{i+n-j} \cdot d_j = v_{i+n} \cdot d_0 + v_{i+n-1} \cdot d_1 + \ldots\ldots + v_i \cdot d_n \qquad 4.3.30$$

All terms here are complex valued, and since $\{v_i\}$ are statistically independent Gaussian random variables with zero mean and variance σ_v^2, the $\{u_i\}$ are Gaussian random variables with zero mean and variance η^2 [40], such that :

$$\eta^2 = \sigma_v^2 \cdot \sum_{j=0}^{n} |d_j|^2 = \sigma_v^2 \cdot |D|^2 \qquad 4.3.31$$

where $|D|$ is the length of the vector D. The feed-back transversal filter Fig.4.3.2 (Filter E) has L-taps, which are given by L-components row vector, such that :

$$E = [e_1, e_2, \ldots\ldots\ldots, e_L] \;\&\; e_h \approx y'_h \;;\; \text{for } h = 1,2,\ldots\ldots,L \qquad 4.3.32$$

The output signal at time $t=iT$ of this filter can be written as :

$$g_{i+n} = \sum_{j=1}^{L} s'_{i-j} \cdot e_j = s'_{i-1} \cdot e_1 + s'_{i-2} \cdot e_2 + \ldots\ldots + s'_{i-L} \cdot e_L \qquad 4.3.33$$

Assuming correct detection , which means :

$$s'_{i,h} = s_{i,h} \;;\; \text{for } h = 1,2,\ldots\ldots, \qquad 4.3.34$$

Then :
$$g_{i+n} = \sum_{j=1}^{L} s_{i-j} \cdot e_j = s_{i-1} \cdot e_1 + s_{i-2} \cdot e_2 + \ldots\ldots + s_{i-L} \cdot e_L \qquad 4.3.35$$

The equalised signal at the input of the detector (ε_{i+n}) at time $t=(i+n)T$ is now given by:

$$\varepsilon_{i+n} = r_{i+n} - g_{i+n} = s_i + u_{i+n} \qquad 4.3.36$$

Now, the detector takes as the detected data-symbol s'_i the possible value of s_i nearest to (ε_{i+n}). Let now, the Z-transform of the SIR of the linear base-band channel be factorised into two factors $P_1(z)$ & $P_2(z)$ such that :

$$P(z) = P_1(z) \cdot P_2(z) = p_0 + p_1 \cdot z^{-1} + p_2 \cdot z^{-2} + \ldots\ldots + p_L \cdot z^{-L} \qquad 4.3.37$$

where : $P_1(z) = 1 + a_1 \cdot z^{-1} + a_2 \cdot z^{-2} + \ldots\ldots + a_{L-m} \cdot z^{-L+m}$ 4.3.38

All roots of $P_1(z)$ lie inside the unit circle in the Z-plane (of absolute value less than unity), and :

$$P_2(z) = b_0 + b_1 \cdot z^{-1} + b_2 \cdot z^{-2} + \ldots\ldots + b_m \cdot z^{-m} \qquad 4.3.39$$

with all its roots (zeros) outside the unit circle in Z-plane (of absolute value greater than unity). Let also $P_3(z)$ be, such that its roots are the complex conjugates of the reciprocals of those of $P_2(z)$, such that :

$$P_3(z) = b_m^* + b_{m-1}^* \cdot z^{-1} + b_{m-2}^* \cdot z^{-2} + \ldots\ldots + b_0^* \cdot z^{-m} \qquad 4.3.40$$

Where b^*_{m-j} is the complex conjugate of b_{m-j}. Therefore the roots (zeros) of $P_3(z)$ lie inside the unit circle in Z-plane. With an infinite number of taps $(n \to \infty)$ of the filter D, it can be shown that the Z-transform of the filter D [40] is :

$$D(z) = (b^*_m)^{-1} \cdot z^{-n} \cdot P_2^{-1}(z) \cdot P_3(z) \qquad 4.3.41$$

Therefore the Z-transform of the channel and filter D is :

$$P(z) \cdot D(z) = (b^*_m)^{-1} \cdot z^{-n} \cdot P_1(z) \cdot P_3(z) = z^{-n} \cdot Y'(z) \qquad 4.3.42$$

where :
$$Y'(z) = (b^*_m)^{-1} \cdot P_1(z) \cdot P_3(z) \qquad 4.3.43$$

Whose Z-transform is given by :

$$Y'(z) = 1 + y'_1 \cdot z^{-1} + y'_2 \cdot z^{-2} + \ldots\ldots + y'_L \cdot z^{-L} \qquad 4.3.44$$

Eqn.4.3.43 simply says that the optimum choice of the linear filter is that which replaces all the roots of the Z-transform of the channel, which lie outside the unit circle by their reciprocals complex-conjugate. The resultant SIR of the channel and filter is now a minimum phase sequence. The change in level introduced by the filter D is ignored by omitting the component $(b^*_m)^{-1}$ [40] in eqns.4.3.41, 4.3.42 and 4.3.43, to give the Z-transform $(z^{-n} \cdot P_2^{-1}(z) \cdot P_3(z))$, the filter introduces an orthogonal transformation into the received signal [40].

Another feature of filter D is that, since it performs a pure phase equalisation, i.e. 'orthogonal transformation' into the received signal, it does not change the signal-to-noise ratio [39,40]. Using eqns.4.3.31, and 4.3.36, the signal-to-noise ratio in the equalised signal (ε_{i+n}), given the correct detection of $s_{i-1}, s_{i-2}, \ldots\ldots, s_{i-l}$, is given by :

$$\Psi_{eq} = \frac{|\bar{s}_i|^2}{2\eta^2} \qquad 4.3.45$$

Where $|\bar{s}_i|^2$ is the mean-square absolute value of s_i and η^2 is the variance of each of the real and imaginary parts of the noise component u_{i+n}. The important property of filter D is that it maximises the signal-to-noise ratio at the detector input, subject to the accurate equalisation of the channel, so that it minimises η^2 [29], and therefore also the sum of the sequares of the absolute values of the components of D, subject to the first component of Y' being held at unity.

Alternatively, it can be seen that the all pass filter D maximises the ratio $(|y'_0|^2 / 2\eta^2)$, where y'_0 is the first component of Y'. This means that the first non-zero component of Y', is maximised relative to the other components in Y', which implies that

the energy of the transmitted signal-element, which is spread over the components of P, is now shifted towards the first component of Y'. The result of this, is that Y' now contains less ISI than P. Bear in mind that the remaining ISI in Y' is removed by the feed-back filter E. Because of cancellation of the ISI, the non-linear equaliser suffers from error extension effects [40]. The tap-gains of both filter D and E can be adjusted adaptively using an iterative solution to update their coefficients [1,39,40,52]. In the next section we will study one way to adjust the tap-gains of the equaliser which is referred here as a minimum phase algorithm.

4.4 MINIMUM PHASE ALGORITHM (Adjustment Method)

The algorithm is an iterative process and can be used to make the channel at least approximately minimum phased by replacing the roots of the sampled impulse response (SIR) of the channel, that lie outside the unit circle in the Z-plane, by their reciprocal complex conjugate. Therefore the resultant SIR now has no roots outside the unit circle. It then uses a knowledge of these roots to determine the tap-gains of the linear feed forward filter, and tap-gains of the feedback filter for the decision feedback equaliser, or to form an estimate of the SIR of the channel and filter, which is needed by the near-maximum likelihood detector. The latter can be preceded by an adaptive linear filter which minimises the number of pre-cursor components (that is, those components appearing before the main components) in the SIR. This reduces the amount of operation that is required by the detector [58,79]. In other words, a simple near-maximum likelihood detector can be developed for detecting the transmitted data-symbols [36,39,58,79].

This algorithm has been developed by Clark and Hau [39], for operating over a telephone channel. However, it is necessary to modify this algorithm to operate over an HF channel. The algorithm requires a knowledge of the roots of the Z-transform of the SIR of the linear base-band channel $P(z)$, that lie outside the unit circle. This is achieved using a suitable root finding algorithm to locate the roots of $P(z)$. These roots are then used to determine the tap-gains of the filter D. In practice the algorithm operates on the estimated SIR of the channel as given by a channel estimator, where the latter is assumed to work perfectly. Therefore the algorithm will operate on the actual SIR instead of the estimated one for these tests as shown in Fig.4.2.2. The input of the algorithm is formed by the software equivalent model of the linear base-band channel, which generate the SIR (P_i) of the four channels considered in these tests, as shown in Fig.4.2.2. The SIR of the invariant channel (P) is given by an ($L+1$)-component row vector :

$$P = [p_0, p_1, \ldots\ldots\ldots, p_L] \qquad 4.4.1$$

whose Z-transform is :

$$P(z) = p_0 + p_1 \cdot z^{-1} + p_2 \cdot z^{-2} + \ldots\ldots + p_L \cdot z^{-L} \qquad 4.4.2$$

Let now : $P(z) = P_1(z) \cdot P_2(z)$ 4.4.3

where: $P_1(z) = \mu \cdot (1 + \theta_1 \cdot z^{-1})(1+\theta_2 \cdot z^{-1}) \cdot \ldots\ldots \cdot (1 + \theta_{L-m} \cdot z^{-1})$ 4.4.4

$$P_2(z) = z^{-m} \cdot (1 + \gamma_1 \cdot z\ )(1+\gamma_2 \cdot z\ ) \cdot \ldots\ldots \cdot (1 + \gamma_m \cdot z\ ) \qquad 4.4.5$$

The $\{\theta_i\}$ from $i=1,2,\ldots..(L-m)$, are the roots (zeros) of $P(z)$. For convenience, it is assumed that no roots of $P(z)$ lie exactly on the unit circle in the Z-plane, and :

$$|\theta_i| < 1 \quad \text{and} \quad |\gamma_i| < 1 \qquad 4.4.6$$

where θ_i is the negative of a root of $P(z)$, and γ_i is the negative of the reciprocal of a root of $P(z)$. It is assumed that all the roots of $P_1(z)$ lie inside the unit circle, where all the roots of $P_2(z)$ lie outside the unit circle. In the case where $P(z)$ has one or more roots on the unit circle, these are taken to be roots of $P_1(z)$, such that : $|\theta_i| = 1$.
The symbol μ in eqn.4.4.4 is an appropriate complex value needed to satisfy eqns.4.4.3-4.4.5. $|\theta_i|$ and $|\gamma_i|$ are the moduli of θ_i and γ_i respectively. The adaptive linear filter (Fig.4.2.1) has $(n+1)$-taps where n is typically between *20* and *50* [23,39]. When the filter is adjusted correctly, the Z-transform of the SIR of it is approximately [40] :

$$D(z) \approx z^{-n} \cdot P_2^{-1}(z) \cdot P_3(z) \qquad 4.4.7$$

where: $P_3(z) = (1 + \gamma_1^* \cdot z^{-1})(1 + \gamma_2^* \cdot z^{-1}) \cdot \ldots\ldots \cdot (1 + \gamma_m^* \cdot z^{-1})$ 4.4.8

and γ_i^* is the complex conjugate of γ_i, such that : $(\gamma_i^* \cdot \gamma_i = 1 \Rightarrow \gamma_i^* = 1/\gamma_i)$.
From eqns.4.4.3 and 4.4.7, the Z-transform of SIR of the channel and filter is given by :

$$Y'(z) = P(z) \cdot D(z) = z^{-n} \cdot P_2^{-1}(z) \cdot P_3(z) \cdot P_1(z) \cdot P_2(z) = \\ = z^{-n} \cdot P_1(z) \cdot P_3(z) \qquad 4.4.9$$

Eqns.4.4.7 and 4.4.9 only apply accurately to the given system when $(n \to \infty)$, but a very good approximation is usually achieved here without using an unduly large value of n. The resultant sampled impulse response of the channel and filter is now minimum phased, i.e. all the roots of $Y'(z)$ lie inside the unit circle in Z-plane. The roots of $P_3(z)$ are the complex conjugate of the reciprocal of the roots of $P_2(z)$, so that the algorithm replaces all roots of $P(z)$ that lie outside the unit circle by the complex conjugate of their

reciprocals, leaving the remaining roots (those of $P_1(z)$) unchanged. The full details of this algorithm are given in (Appendix C), which is based on ref. [39]. Finally the description of operation of the latter algorithm over the HF channel is explained next.

4.4.1 MINIMUM PHASE ALGORITHM FOR HF CHANNEL

The algorithm is based on the one developed by [39] for time invariant channels (telephone channel), which is explained in Appendix C.

For operating over time-varying channels (HF channel), the algorithm has to be modified. The reason for this modification is that :

The sampled impulse response (SIR) of the linear base-band channel in eqn.4.4.1, which does not vary with time, is here given by the $(L+1)$-component vector P_i such as :

$$P_i = [p_{i,0}, p_{i,1}, \ldots\ldots\ldots, p_{i,L}] \qquad 4.4.10$$

where now P_i varies with time. For the test considered here, the algorithm is assumed to operate with the actual SIR of the linear base-band channel rather than the estimated one, i.e. the algorithm has an exact knowledge of the SIR, or in other words we assume perfect estimation. This assumption is necessary here, for testing the algorithm over an HF channel without introducing any errors caused by the estimator. In practice, the algorithm will operate on the estimated SIR. Therefore the iteration process described in Appendix C must operate on the sequence $\{p_{i,h}\}$, for $h=0,1,\ldots\ldots,L$ every instant of time and adjusts the tap-gains of the filter continuously, i.e. both Y'_i and the tap-gains are calculated every $t=iT$ *for* $i=1,2,3,\ldots\ldots\ldots\ldots\ldots$ The algorithm now operates on the sequence $\{p_{i,h}\}$, for $h=0,1,\ldots\ldots,L$ in the following manner :

The algorithm always starts with β_1 set to the first starting point, where nine starting points are used here instead of five used by [39], for the telephone channel. The algorithm uses all the nine possible starting points (see section 4.5, p.107) for determining any new root of the HF channel. This is done in order to facilitate the tracking of as many as possible of the roots which lie outside the unit circle, at the beginning of the process. The values of the nine starting points are given later. To do this, the algorithm appropriately adjusts the one-tap feedback filter shown in Fig.C-2 whose input is now the SIR of the HF channel eqn.4.4.10. The Z-transform of the filter is given by eqn.C-17 as:

$$(1+\beta_j \cdot z^{-1})^{-1} = 1-\beta_j \cdot z^{-1}+\beta_j^2 \cdot z^{-2}-\beta_j^3 \cdot z^{-3}+\cdots\cdots\cdots \qquad 4.4.11$$

The process works on the estimate of the root γ_1 eqn.4.4.5 which is given by eqn.C-16 as

$$\beta_{j+1} = \beta_j + d \cdot \frac{e_{j,0}}{\delta_j} \qquad 4.4.12$$

where β_{j+1} is a better estimate of γ_1 than is β_j, d is a positive real constant in the range *(0-1.1)*, δ_j is given by eqn.C-14, and $(e_{j,0})$ is the output element in the sequence of the filter output in Fig.C-2. Bear in mind that the first estimate of γ_1, at the start of the process is β_0 which is set to the first starting point, as explained in Appendix C. The delay of one sampling interval T in the feed-back filter (Fig.C-2) becomes an advance of T, with Z-transform z. Thus the effective Z-transform of the previous filter becomes $A_j(z)$ in eqn.C-1, as :

$$A_j(z) = (1 + \beta_j \cdot z)^{-1} \qquad 4.4.13$$

using eqn.4.4.12 a new one-tap feed-back filter is formed replacing β_j by β_{j+1} such as :

$$A_{j+1}(z) = (1 + \beta_{j+1} \cdot z)^{-1} \qquad 4.4.14$$

The process continue in this way, until terminated when one of the following condition have been reached :

1. $|e_{j,0}/\delta_j| < b$ 4.4.15
2. j = the chosen number of iterations, which is in the range *(30-50)*.
3. $|\beta_j| > d = 1.05$ 4.4.16
4. $\delta_j \to 0$ 4.4.17

where b is an appropriate small positive real constant. When condition *2* or *3* is reached. the algorithm is said to have diverged and so is terminated, for each of all nine starting points. In this case it is assumed that all m roots that lie outside the unit circle have been found. In case *4*, when $\delta_j \to 0$, overflow occurs in $(e_{j,0}/\delta_j)$, so that the algorithm has to be terminated. When condition *1* is achieved the iteration process is taken to have converged. Let the value of j at convergence be k, so that :

$$\beta_k = \gamma_1 \qquad 4.4.18$$

The sequence obtained at the output of the feed-back filter is now given by eqn.C-21 as:

$$P(z) \cdot A_k(z) \approx e_{k,0} + e_{k,1} \cdot z^{-1} + e_{k,2} \cdot z^{-1} + \ldots\ldots + e_{k,L} \cdot z^{-L} \qquad 4.4.19$$

where $(e_{k,0} = 0)$ [39] (see Appendix C). For subsequent runs however, the m roots that were tracked in the previous run, are added to the nine starting points, to give a total of $(m+9)$ new starting points. This is useful for the next run to improve the performance of

the process to find most of the roots in the next sequence. These $(m+9)$ starting points are arranged as shown in Fig.4.4.1, and the iteration process starts with β_1 set to γ_1, where γ_1 now is the first root determined in the previous run.

The algorithm next appropriately adjusts the two-tap feed-forward filter that shown in Fig.C-3, whose Z-transform is given by :

$$B_k(z) = (1+\beta_k^* \cdot z^{-1}) \qquad 4.4.20$$

where now the input to this filter coming in the correct order to give the output sequences of $(L+2)$-components whose Z-transform is given by eqn.C-23 as :

$$\begin{aligned} Y'_1(z) &= y'_{1,-1} + y'_{1,0} \cdot z^{-1} + y'_{1,1} \cdot z^{-2} + \cdots\cdots + y'_{1,L} \cdot z^{-L-1} \\ &\approx P\ (z) \cdot A_k(z) \cdot B_k(z) \end{aligned} \qquad 4.4.21$$

where $(y'_{1,-1} \approx 0)$ [39] (Appendix C). Multiplying eqn.4.4.21 by z, i.e. $\{y'_{1,h}\}$ is advanced by one place (sampling interval), then eqn.4.4.21 becomes :

$$Y'_1(z) = y'_{1,0} + y'_{1,1} \cdot z^{-1} + \cdots\cdots + y'_{1,L} \cdot z^{-L} \approx C_1(z) \cdot P\ (z) \qquad 4.4.22$$

where :

$$C_1(z) = A_k(z) \cdot B_k(z) \approx (1+\gamma_1 \cdot z)^{-1} \cdot (1+\gamma_1^* \cdot z^{-1}) \qquad 4.4.23$$

The two filters in Fig.C-2 and Fig.C-3 are equivalent to one filter whose Z-transform is $C_1\ (z)$, an all-pass filter. As a result the root $(-1/\gamma_1)$ of $P(z)$ is replaced by the root $(-\gamma^*_1)$, which is the complex conjugate of its reciprocal and lies inside the unit circle in the Z-plane. The sequence Y'_1 is now reversed in order and used in place of sequence $P_{i,1}$ eqn.4.4.10, as the starting point for allocating the next root. The process is repeated in the same way until all the roots of $P_i\ (z)$ have been allocated. Whenever a root is found or a new starting point is needed due to divergence of the iteration process, the next $(m+9)$ starting points are used $(\gamma_2, \cdots\cdots)$.

The iteration process is repeated until all $(m+9)$ starting points have been used, at which stage it is assumed there are no more roots of $P_i\ (z)$ that lie outside the unit circle and the process is finished. Thus the starting points for any subsequent run are the roots determined from the previous run, plus the initial nine starting points, i.e. $(m+9)$.

The number of iterations (j) required for a normal HF channel (slowly-varying channel) can be in the range $(30\text{-}40)$, which is adequate to allocate all the roots. For the very worst HF channel it can be increased to $(50\text{-}60)$, so that greater latitude is allowed in the iteration process, before determining that the process has diverged [23,39]. The roots (zeros) of the time-varying channel $P_i\ (z)$ may from time to time cross the unit circle [23],

therefore an infinite number of taps are required for the adaptive filter to process the roots on the unit circle [23,39]. To avoid this difficulty, the algorithm is now limited, so that it only treats the roots of $P_i(z)$ with absolute values greater than d, where d is chosen equal to *1.05* eqn.4.4.16. The initial nine starting points used in the minimum phase algorithm, when operating over an HF channel are given below :

1.)	0.00+j0.00	4.)	0.0+j0.90909	7.)	0.64282+j0.64282
2.)	0.90909+j0.0	5.)	-0.90909+j0.0	8.)	-0.64282+j0.64282
3.)	0.0-j0.90909	6.)	0.64282-j0.64282	9.)	-0.64282-j0.64282

TABLE 4.1 The Initial Nine Starting Points for HF Channel.

when d is set to *1.05*, the algorithm only searches and tracks those roots of $P_i(z)$ that have absolute values greater than *1.05* which mean γ_1 eqn.4.4.5 and also (γ_2,.....) all have absolute values less than (*1/1.05=0.92381*).

The resultant sequence (Y'_i) with ($L+1$)-components is given now by the row vector :

$$Y'_i = [y'_{i,m0}, y'_{i,m1}, y'_{i,m2}, \cdots\cdots\cdots, y'_{i,mL}] \qquad 4.4.24$$

which is the minimum phased sequence. The latter sequence is needed by NML detectors after an appropriate scaling to make the first component of this sequence equal to unity (1), as shown in Fig.4.2.2. The output at the scaling terminal is given by ($L+1$)-components row vector Y_i such that :

$$Y_i = [y_{i,0}, y_{i,1}, y_{i,2}, \cdots\cdots\cdots, y_{i,L}] \qquad 4.4.25$$

$$\text{where : } \; y_{i,0} = 1.0 \quad \text{and} \quad \{y_{i,h}\} = \{\frac{y'_{i,h}}{y'_{i,0}}\} \; ; \quad \text{for} \quad h = 1,2,\ldots\ldots,L \qquad 4.4.26$$

and the vector Y_i is now the scaled minimum phased version of the linear base-band channel, needed by NML detectors for satisfactory operation. In the next chapter the performance of different NML detectors will be examined using the sequence $\{y_{i,h}\}$ generated by the previous method (algorithm). The received signal at NML detector in the absence of the linear feed forward filter is now given by :

$$r_i = \sum_{h=0}^{L} s_{i-h} \cdot y_{i,h} + v_i \qquad 4.4.27$$

where the $\{v_i\}$ are scaled components of the noise. The reason for this scaling will be explained in the next chapter.

4.4.2 UPDATING THE TAP-GAINS OF LINEAR PRE-DETECTION FILTER

The adaptive linear feed forward transversal filter inserted between the linear base-band channel and the NML detector is shown in Fig.4.4.2. The filter has the tap-gains given by the $(N+1)$-component row vector :

$$D_i = [d_{i,0}, d_{i,1}, d_{i,2}, \cdots\cdots\cdots, d_{i,N}] \qquad 4.4.28$$

These taps are adjusted adaptively, using the knowledge of the new set of roots, which have been found previously. The Z-transform of the filter is given by :

$$D_i(z) = d_{i,0} + d_{i,1} \cdot z^{-1} + d_{i,2} \cdot z^{-2} + \cdots\cdots\cdots + d_{i,N} \cdot z^{-N} \qquad 4.4.29$$

The tap-gains of this filter are updated at every sampling instant. N is the number of taps in the range *30-50*.

The adjustment procedure is given in ref.[39] and can be summarised as follows :

At the start of the adjustment procedure $\{d_{i,h}=0$; *for* $h=0,1,....N-1$ *and* $d_{i,N}=1.0\}$, thus the filter has Z-transform given by : $(D_0(z) = z^{-N})$.

The sequence $\{d_{0,i}\}$ is then fed through the two-tap filter shown in Fig.4.4.3, whose Z-transform is given by eqn.4.4.20, $B_1(z)=(1+\gamma^*_1 z^{-1})$, where γ_1 is the negative of the reciprocal of the first root of $P(z)$, which lies out side the unit circle. γ_1 has been determined by the previous algorithm. This gives an output sequence with $(N+2)$-components whose Z-transform is $\{B_1(z).D_0(z)\}$, which is reversed in order and fed via the one-tap filter as shown in Fig.4.4.3, whose Z-transform is approximately :

$$D_1(z) = D_0(z).B_1(z).A_1(z) \qquad 4.4.30$$

Now $D_1(z)$ has just $(N+1)$-components and in the correct order. In the next run of the process $D_0(z)$ is replaced by $D_1(z)$ and γ_1 by γ_2 to determine $D_2(z)$.

When the process has allocated all the m roots of $P(z)$ that lie outside the unit circle, the sequence of the feed-back filter in Fig.4.4.3, will have the Z-transform $\{D_m(z) \approx D(z)\}$ (eqn.4.4.7), so that the Z-transform of the resultant channel and adaptive filter is :

$$Q(z) \approx P(z) \cdot D_m(z) \approx Y'(z) \approx \rho_0 + \rho_1 \cdot z^{-1} + \cdots\cdots + \rho_{N+L} \cdot z^{-N-L} \qquad 4.4.31$$

where $\rho_{i,h} \approx 0$ for $h=0,1,2.....N-1$.

Therefore $\{Y'_m(z) \approx z^N \cdot Y'_i(z) = P_1(z) \cdot P_3(z)\}$, (eqn.4.4.24), is the resultant SIR, that is used by the detector. More details about this process is given elsewhere in ref.[39].

The above procedure can be used to adjust the tap-gains of the non-linear (decision feed-back) equaliser or the feed forward filter, that precedes the NML detector, and both systems will be examined in the next chapter. In this case the received samples $\{r'_i\}$, eqn.4.2.1 instead of being fed directly to the detector, are now passed first through an adaptive filter shown in Fig.4.4.2, whose output signal is now given at $t=(i+N)T$ by :

$$r_{i+N} = \sum_{h=0}^{N} r'_{i+N-h} \cdot d_{i+N,h} \qquad 4.4.32$$

using eqn.4.2.1, then :

$$r_{i+N} = \sum_{h=0}^{N} \{ \sum_{h'=0}^{L} s_{i+N-h'} \cdot p_{i+N,h'} + v_{i+N} \} \cdot d_{i+N,h} = \sum_{h=0}^{N+L} s_{i+N-h} \cdot c_{i+N,h} + u_{i+N} \qquad 4.4.33$$

where :

$$u_{i+N} = \sum_{h=0}^{N} v_{i+N} \cdot d_{i+N,h} \qquad 4.4.34$$

is the noise components and :

$$C_{i+N} = P_{i+N} \cdot D_{i+N} \ or \ c_{i+N,h} = p_{i+N,h} \cdot d_{i+N,h} \ , \text{for } h=0,1,.....N+L \qquad 4.4.35$$

is the sampled impulse response of the channel and filter, and at time $t=(i+N)T$ is given by the $(N+L+1)$-component row vector as :

$$C_{i+N} = [c_{i+N,0}, c_{i+N,1}, \cdots\cdots, c_{i+N,N-1}, c_{i+N,N}, \cdots\cdots, c_{i+N,N+L}] \qquad 4.4.36$$

The samples $\{r_{i+N}\}$ from the adaptive filter are fed to a detector. Assuming ideal operation of the filter and comparing eqn.4.4.35 with eqn.4.4.31, it can be shown that :

$$c_{i+N,h} \approx 0 \ , \text{for } h=0,1,.....N-1 \qquad 4.4.37$$

Therefore eqn.4.4.33 becomes :

$$r_{i+N} = \sum_{h=N}^{N+L} s_{i+N-h} \cdot c_{i+N,h} + u_{i+N} \Rightarrow \qquad 4.4.38$$

$$r_i = \sum_{h=0}^{L} s_{i-h} \cdot c_{i,h} + u_i$$

The filter introduces a delay of N sampling intervals, such that the earliest symbol that could be detected at $t=iT$ is s_{i-N} . The filter here gives at its output the sequence Y'_i where

$$Y'_i = [y'_{i,0}, y'_{i,1}, y'_{i,2}, \cdots\cdots, y'_{i,L}] \qquad 4.4.39$$

where ($y'_{i,h} \approx c_{i+N,N+h}$) for $h=0,1,2,.....L$. Y'_i is an estimate of the value of sampled impulse response of the linear base-band channel and the feed forward linear filter in cascade at $t=(i+N)T$. When the correct adjustment is achieved the sequence Y'_i should be at least approximately minimum phased. The detector uses (r_{i+N} and Y'_i) to detect the value $s_{i+N+\mu}$, where μ is the delay introduced by the NML detector itself, so that the total delay in detection is $(N+\mu)$ sampling intervals. In the case of the non-linear equaliser the delay is (N) sampling intervals, because the non-linear equaliser does not introduce any delay. The signal at the feed-back section of the decision feed-back equaliser Fig.4.3.2 is given by :

$$g_{i+N} = \sum_{h=1}^{L} s'_{i-h} \cdot y'_{i,h} \qquad 4.4.40$$

where s'_{i-h}, for $h=1,2,3,.....L$ are the early detected data symbols. The error signal ε_i in Fig.4.3.2 is now given by :

$$\varepsilon_{i+N} = r_{i+N} - g_{i+N} \qquad 4.4..41$$

s'_{i+N} can be determined from ε_i. With correct detection of $s'_{i-1}, s'_{i-2}, \ldots\ldots, s'_{i-L}$, then :

$$g_{i+N} = \sum_{h=1}^{L} s_{i-h} \cdot y'_{i,h} \qquad 4.4.42$$

$$\Rightarrow \varepsilon_{i+N} = s_i \cdot c_{i+N,N} + \sum_{h=1}^{L} s_{i-h} \cdot c_{i+N,h+N} - \sum_{h=1}^{L} s_{i-h} \cdot y'_{i,h} + u_{i+N} \approx s_i + u_{i+N} \qquad 4.4.43$$

Then at $t=iT$, $\varepsilon_i = s_{i-N} + u_i$, and an exact equalisation of the channel has been achieved such as :

$$c_{i+N,N} \approx 1 \quad \text{and} \quad \sum_{h=1}^{L} s_{i-h} \cdot c_{i+N,h+N} \approx \sum_{h=1}^{L} s_{i-h} \cdot y'_{i,h} \qquad 4.4.44$$

Finally, the previous algorithm should be implemented every sampling instant for the HF channel, such that both Y'_i and tap-gains of the feed forward filter are calculated every instant of time, ($t=iT$, for $i=1,2,3,.....$). The results of computer simulation tests are given at the end of this chapter to assess the performance of the minimum phase algorithm, when operating over four HF channel models.

4.5 SIMULATION RESULTS

Computer simulation tests have been carried out on the minimum phase algorithm (MPA) described in section 4.4, operating over four different HF channel models. The model of the system under test is that described in section 4.2. The algorithm is assumed to operate on the actual sampled impulse response (SIR) of the linear base band channel rather than the estimated one. The SIR's for the four channels have been generated using

an equivalent simulator software model, that described in Chapter 2. The length of the SIR ($L+1$) is taken to be *20*, *20*, *16*, and *12* for channel 1, 2, 3, and 4 respectively. The parameters of the four channels are given in Chapter 2, and the computer simulation model for testing the MPA is represented in Fig.4.2.2.

The algorithm has been tested over each channel separately and its performance is given in two and three dimensional graphs in Figs.4.5.1-4.5.8. In all tests considered here, the threshold level (b) in eqn.4.4.15 is in every case set to $b=10^{-10}$, which is used to terminate the iterative process. The roots (zeros) of the SIR of the HF channel $P_i(z)$ may from time to time cross the unit circle, therefore if the adaptive filter is now adjusted, so that it operates on all roots of $P_i(z)$ outside the unit circle with absolute values greater than unity ($d=1$), an infinite number of taps are required for the filter, since it must, from time to time process roots that effectively on the unit circle. To avoid this difficulty, the algorithm is now constrained, so that it only processes roots of $P_i(z)$ with absolute values greater than d, where d is now selected to be $d=1.05$. Therefore the algorithm only searches and tracks those roots of $P_i(z)$ that have absolute values greater than *1.05*, which means that β_1 in eqns.4.4.11-4.4.23 and also β_2, β_3, ... all have absolute values less than *0.9480*. This suggests the initial values of nine starting points with which to start the algorithm. The nine starting points are given in section 4.4.

Figs. 4.5.1-4.5.4 show some segments in three-dimensional plots of the over-all linear base band channel SIR before and after minimum phase, where the X-axis represents the number of components in the SIR of the channel ($L+1$), taken in this graph to be *20*, *20*, *16* and *12* for channel 1, 2, 3, and 4 respectively. The Y-axis represents the time instant (i), where i is taken to be *1,2,3,.....,50* sample intervals, and the Z-axis represents the amplitude variation of the channel that is given by :

$$z_{i,h} = \Re\, \bar{y}'^{2}_{i,h} + \Im \bar{y}'^{2}_{i,h} \quad ; \quad \text{for} \quad h = 0,1,2,.....L$$

The figures also show the coefficient settings of the filter D.

Each figure presents a segment of *50 intervals* from the whole sequence of each channel, which is about *56000* samples, and represents about *25 seconds*. The number of each segment is written beside each graph. Also some segments are represented by taking samples every *24* or *960 intervals* to allow us to see more samples from each channel. It can be seen from the non-minimum phase (raw SIR) graphs that as the number of sky waves is increased the length of SIR ($L+1$) of the channel increases and the peaks are not concentrated around the earliest components or any particular one. The MPA operates on the SIR of the channel trying to make the latter at least approximately minimum phase. The figures clearly indicate that the energy of the SIR is now concentrated around the earliest components, therefore the algorithm succeeds in converting the SIR from non-minimum phased to at least approximately minimum phased. As a result of this the impulse response of the channel and filter will then exhibit a more rapid rise to its peak

value, and also be of shorter duration (shorter than the non-minimum phased SIR). It can also be seen from the graphs in Fig.4.5.1-4.5.4 that, the coefficients of the filter D are now situated around the latest components, which means that, when the filter has *50* taps (*N=50*), then the largest taps will be around the component (*40-50*). This is an interesting result that indicates the first few components of each received signal at the detector input are the largest components. Therefore near maximum likelihood (NML) detection can be achieved quite simply, leading to a considerable reduction in the complexity of the receiver. The coefficient of the filter will now be $D_i=[d_0, d_1, d_2,.....,d_{N-k}, d_{N-k+1},....,d_N]$ where k is in the range *10-2*.

The filter D is simply delays the received signal by N places. The first $(N-k)$ tap is very close to zero as we can see clearly from the previous graphs, therefore the vector D_i becomes now :

$$D_i \approx [0, 0, ...0, d_{N-k}, d_{N-k+1},...., d_N]$$

Other graphs are presented here in 2 dimensions, which show the over-all linear base band channel SIR before and after MPA. They also show the taps for the filter D. They show the variation of the energy contained within each sample at the given time instant. These graphs make the idea of MP very clear, where each graph represents the SIR of each channel at one instant of time. It can be seen that the MPA shifts the energy towards the earliest component. Also the graphs show the taps of the adaptive filter setting around the last *10* components.

Figs.4.5.1-4.5.8 indicate that the MPA operates better over channel 4 and 3 than it does over channel 1 and 2. This is due to fact that the latter channels are the most severe channels, since they exhibit 3-sky and 2-sky waves with a large frequency spread and time delay, causing the algorithm some times to miss some roots to allocate (process) them. Even when the algorithm fails to process a zero that lies outside the unit circle, this does not of itself introduce any discrepancy between the actual and estimated values of the SIR of the channel and filter, so that the only consequence of the given failure is that the SIR of the channel approximates less closely to ideal form. With a good NML detector, this does not noticeably degrade the performance of the system as we will see in the next chapter. Therefore the algorithm operates over channel 1 and 2 quite well as it can be seen from the Figs.4.5.1-4.5.8.

The number of iteration processes is taken to be *40* in all tests considered here, and this is not restricted, and can be decreased to less than *40*, as we will see in the next chapter. In summary the MPA converts the linear base band channel from non-minimum phased to at least approximately minimum phased, i.e. most of the phase distortion introduced by the channel is now removed. In the presence of both amplitude and phase distortion, the algorithm first corrects the phase distortion and next it converts the resulting linear-phase amplitude distortion into the corresponding minimum phased amplitude distortion, without however changing the amplitude distortion itself.

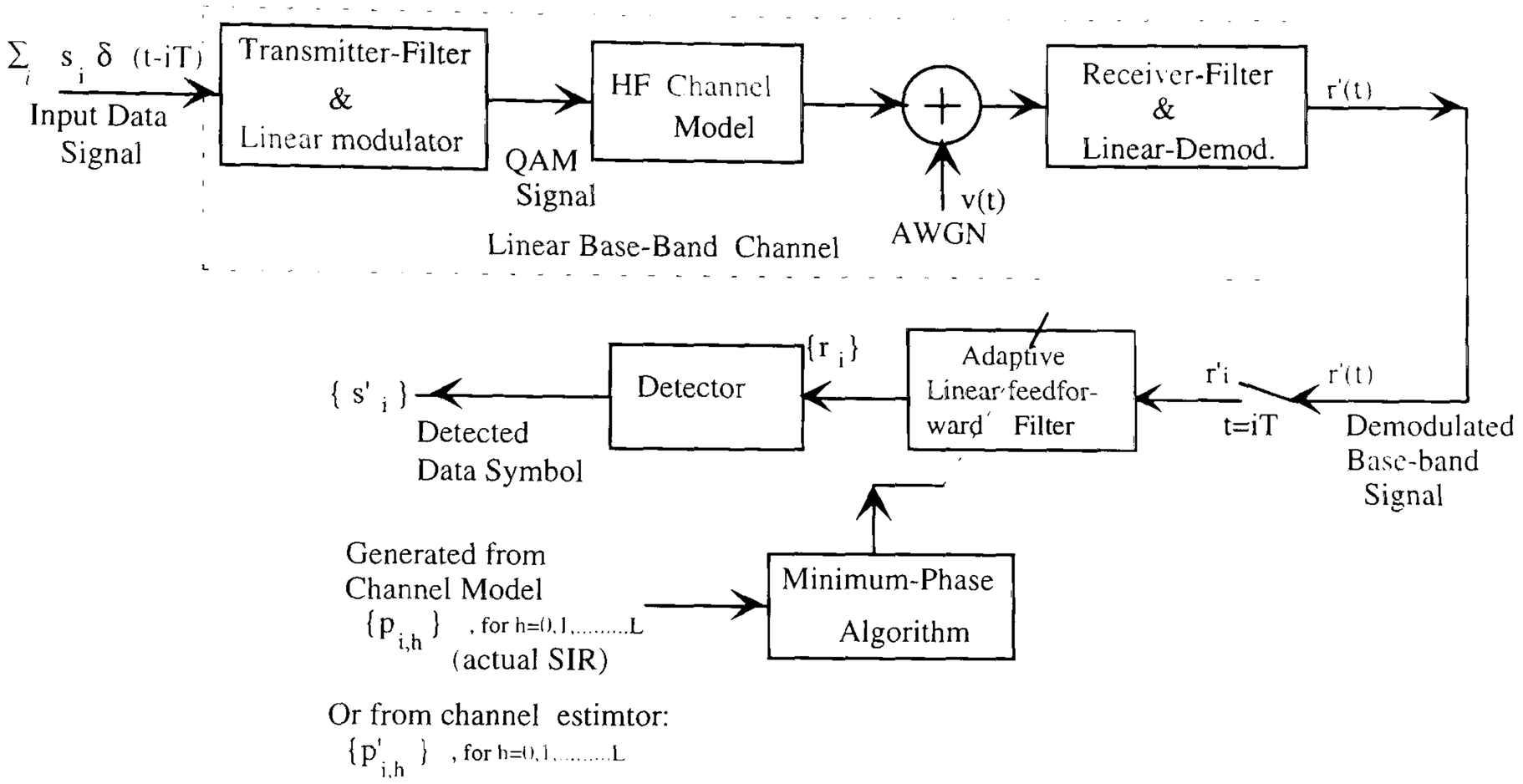


Fig.4.2.1 Model of The System .

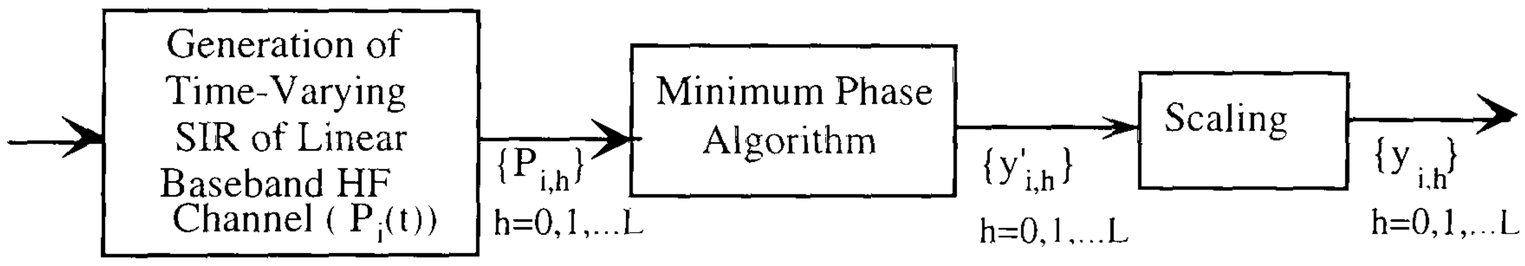


Fig.4.2.2 Generation of the Scaled Min. Phased SIR of the Linear Base-band channel.(Computer-simulation model for testing Min. Phase Algorithm).

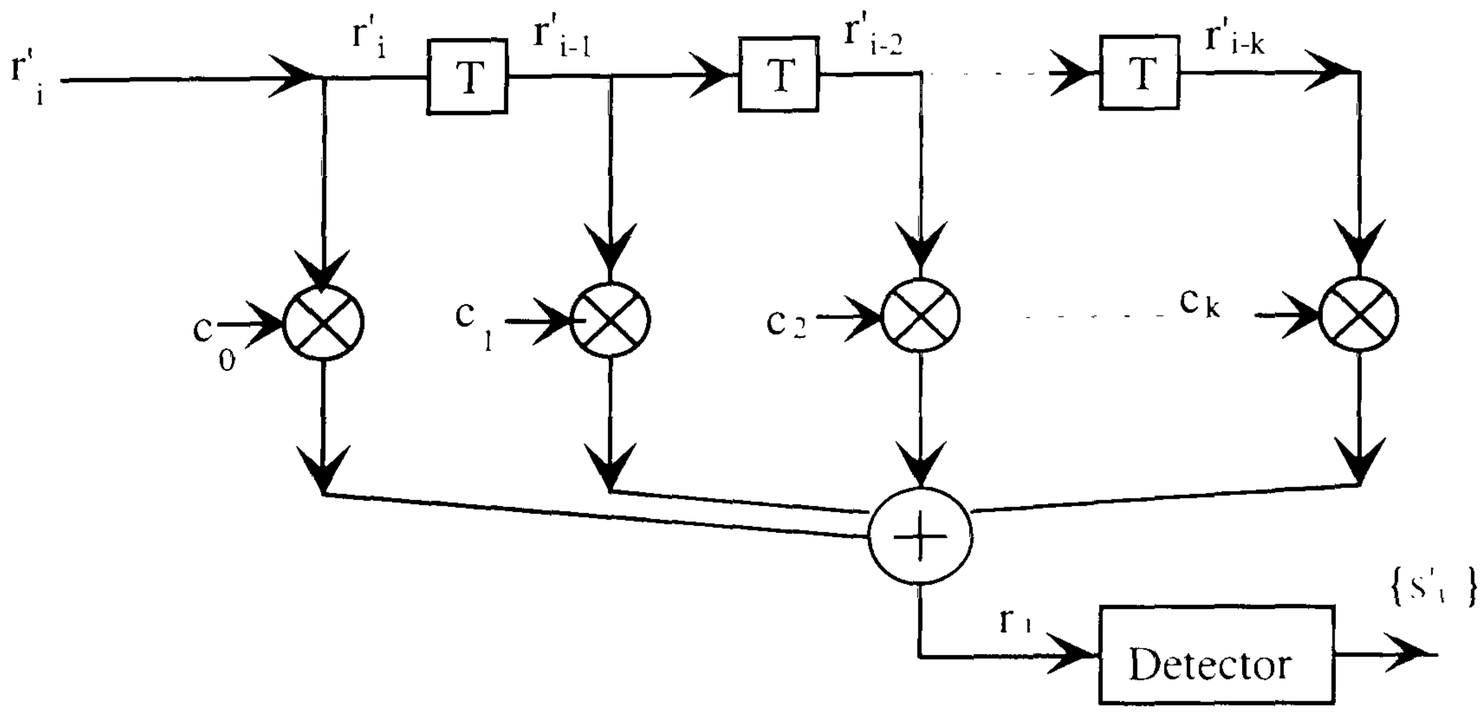


Fig.4.3.1 Linear Equaliser.

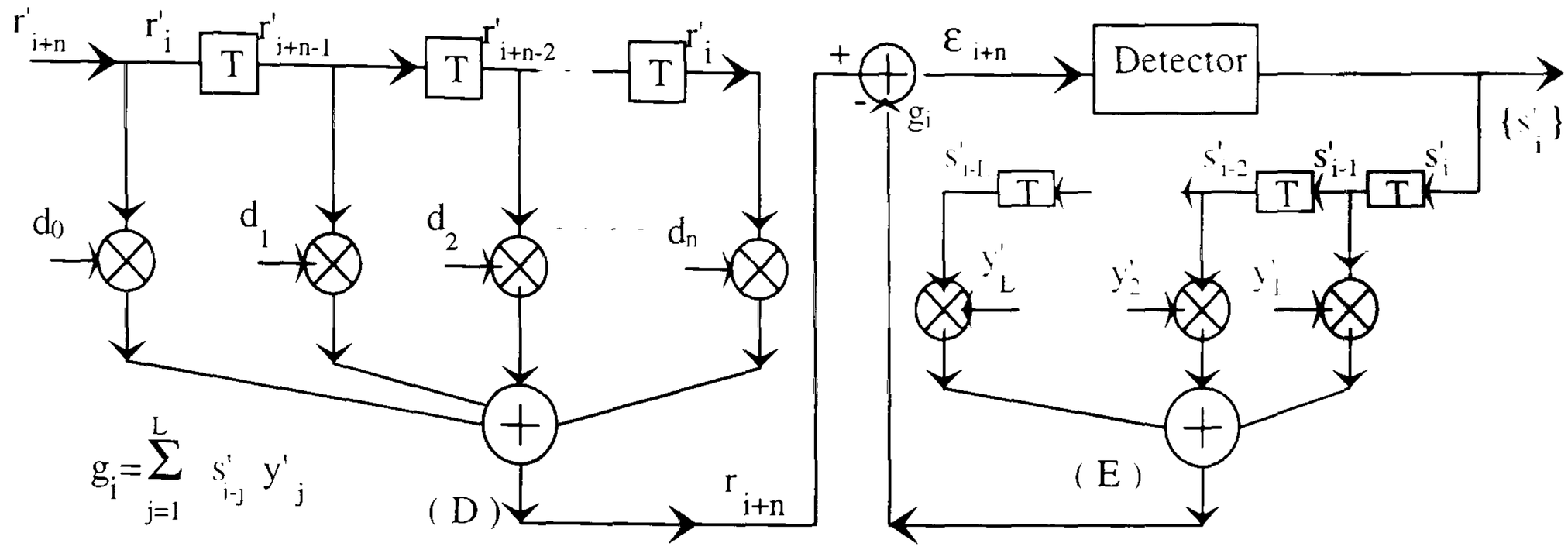


Fig.4.3.2 : Non-Linear (Decision Feedback) Equaliser.

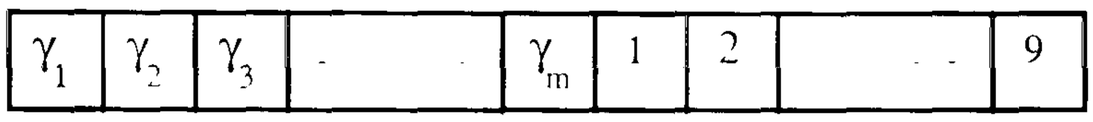


Fig.4.4.1 : Arrangement of Starting Points for the Iteration Process.

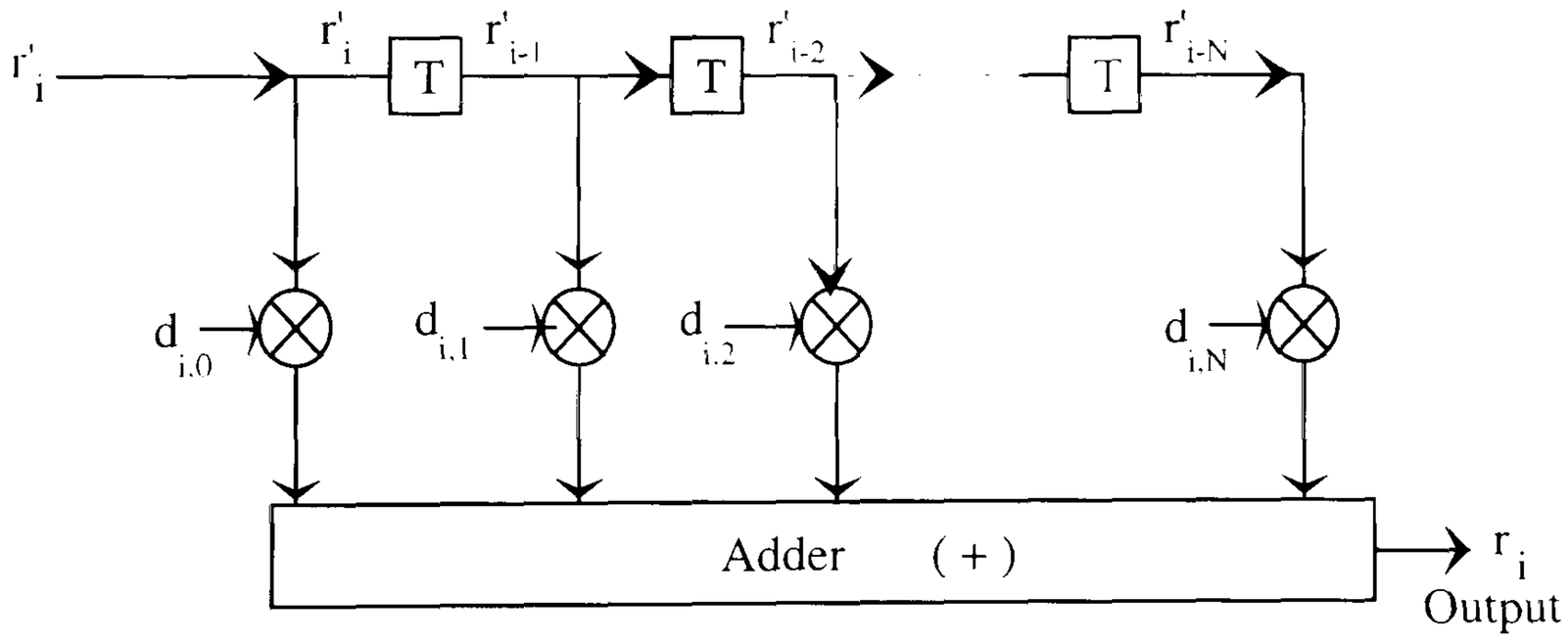


Fig.4.4.2 : Pre-Detection Transversal Filter.

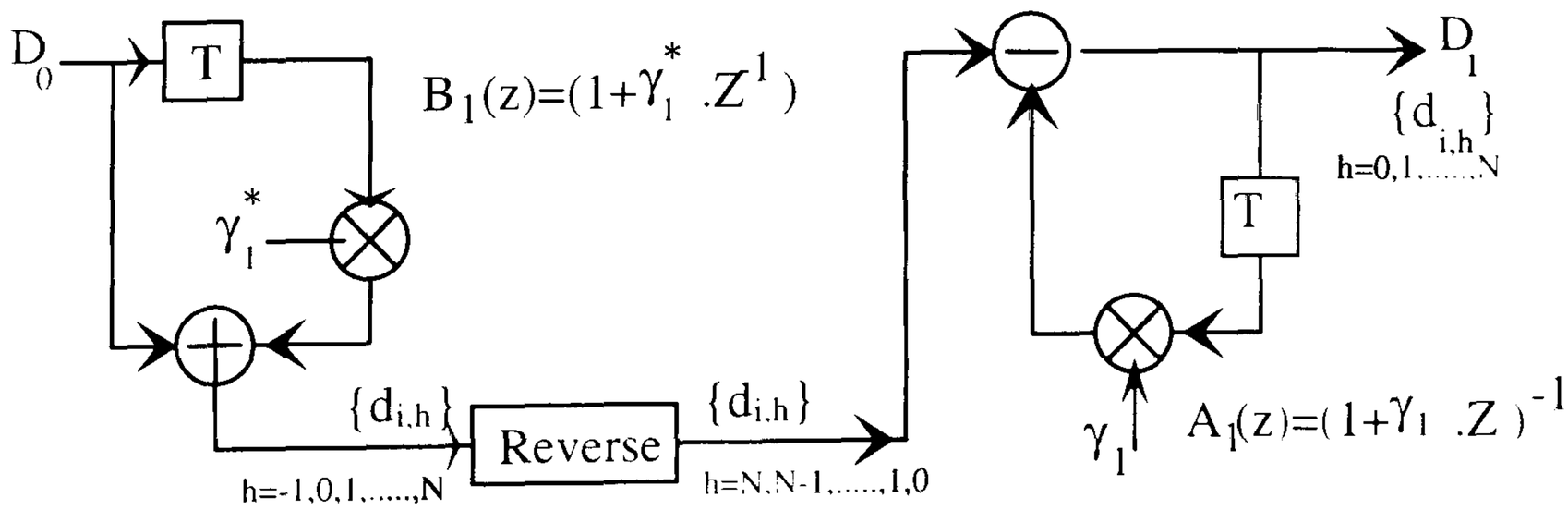
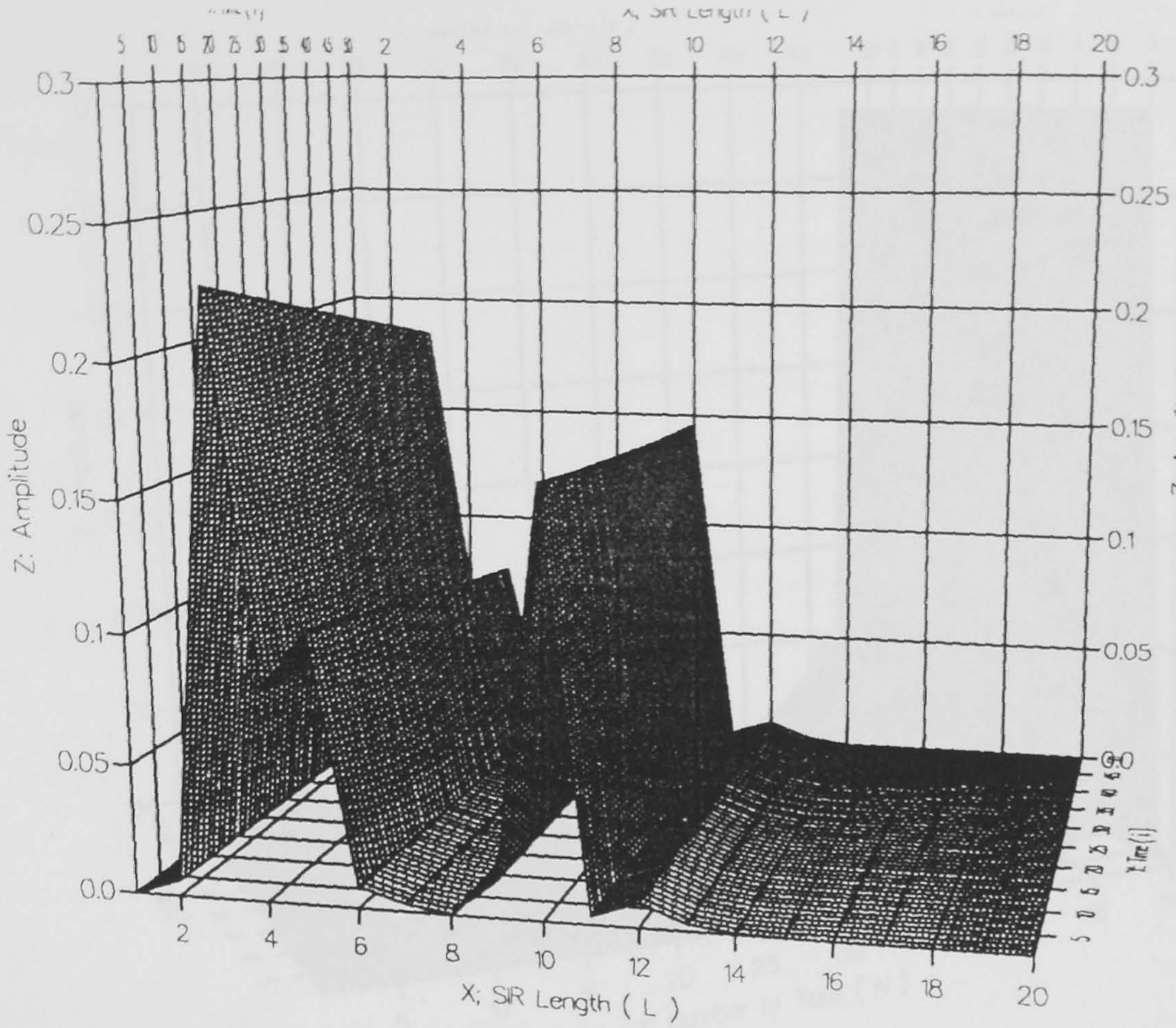


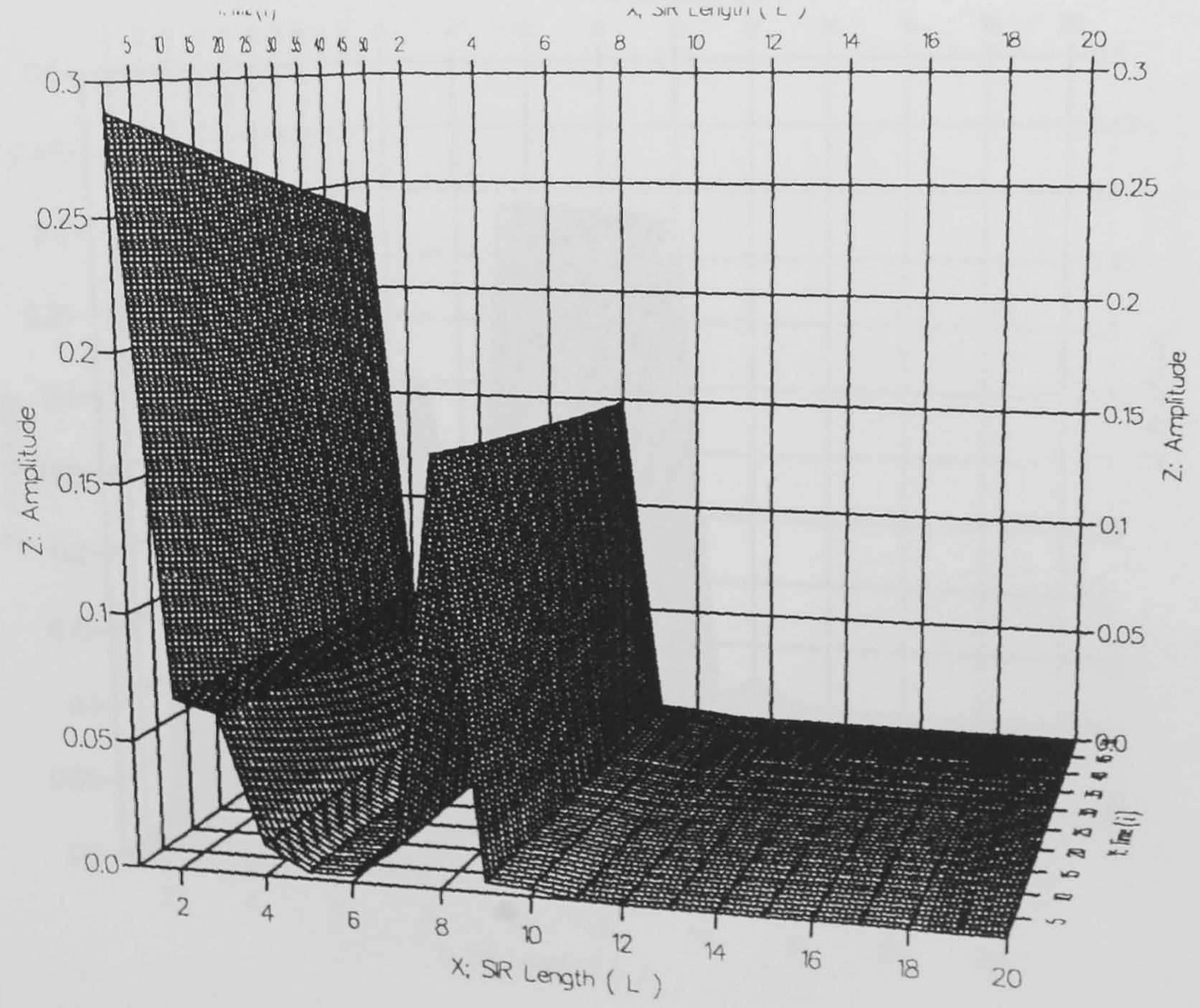
Fig.4.4.3 : Mechanism of Adjustment Procedure .



(i=6000-6050)

-Before Minimum Phase

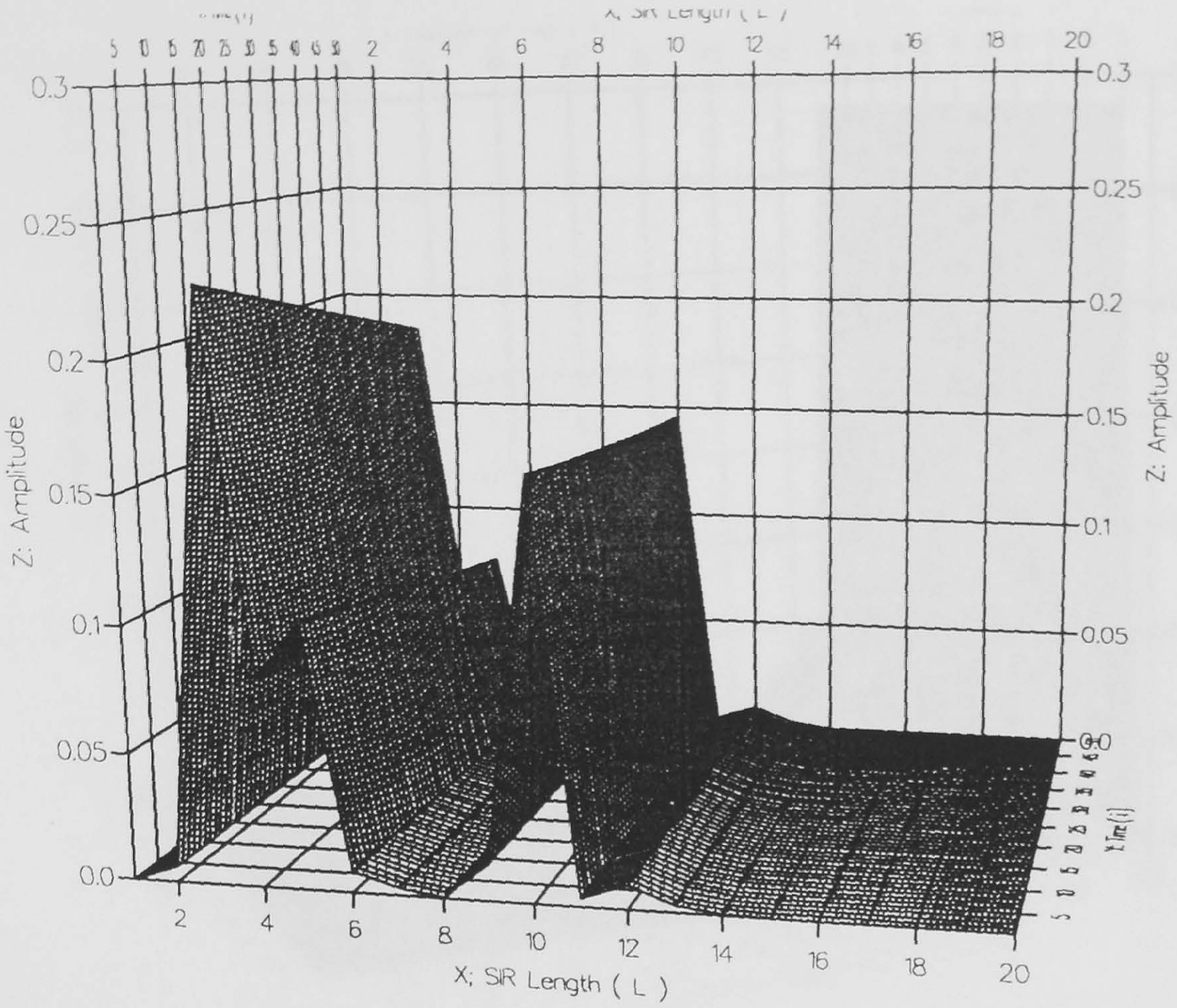
a : A Segment of the Amplitude Variation of Channel 1 SIR before MP at : (i = 6000-6050).



(i=6000-6050)

-After Minimum Phase.

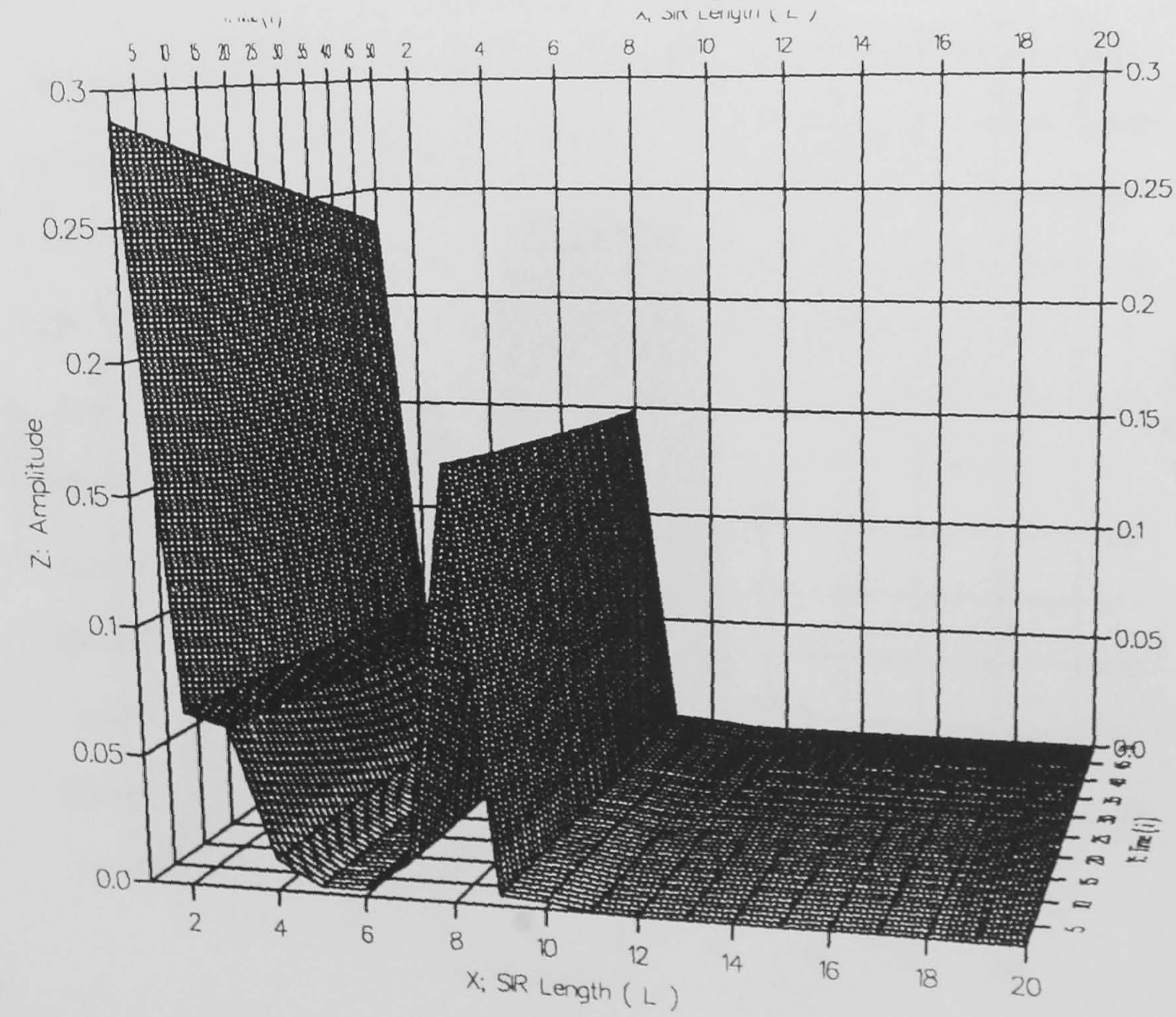
b : A Segment of the Amplitude Variation of Channel 1 SIR after MP at : (i = 6000-6050).



(i=6000-6050)

-Before Minimum Phase

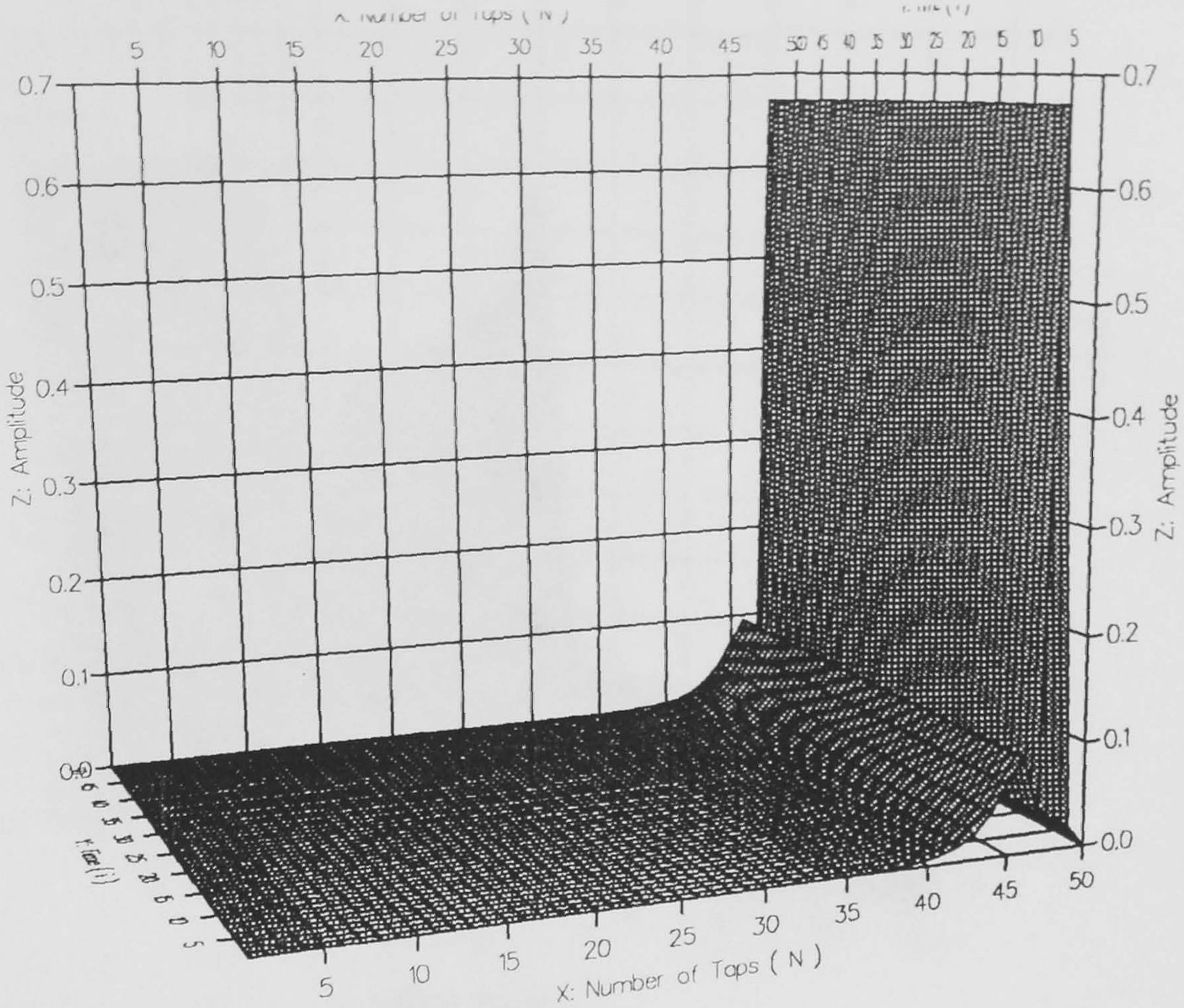
a : A Segment of the Amplitude Variation of Channel 1 SIR before MP at : (i = 6000-6050).



(i=6000-6050)

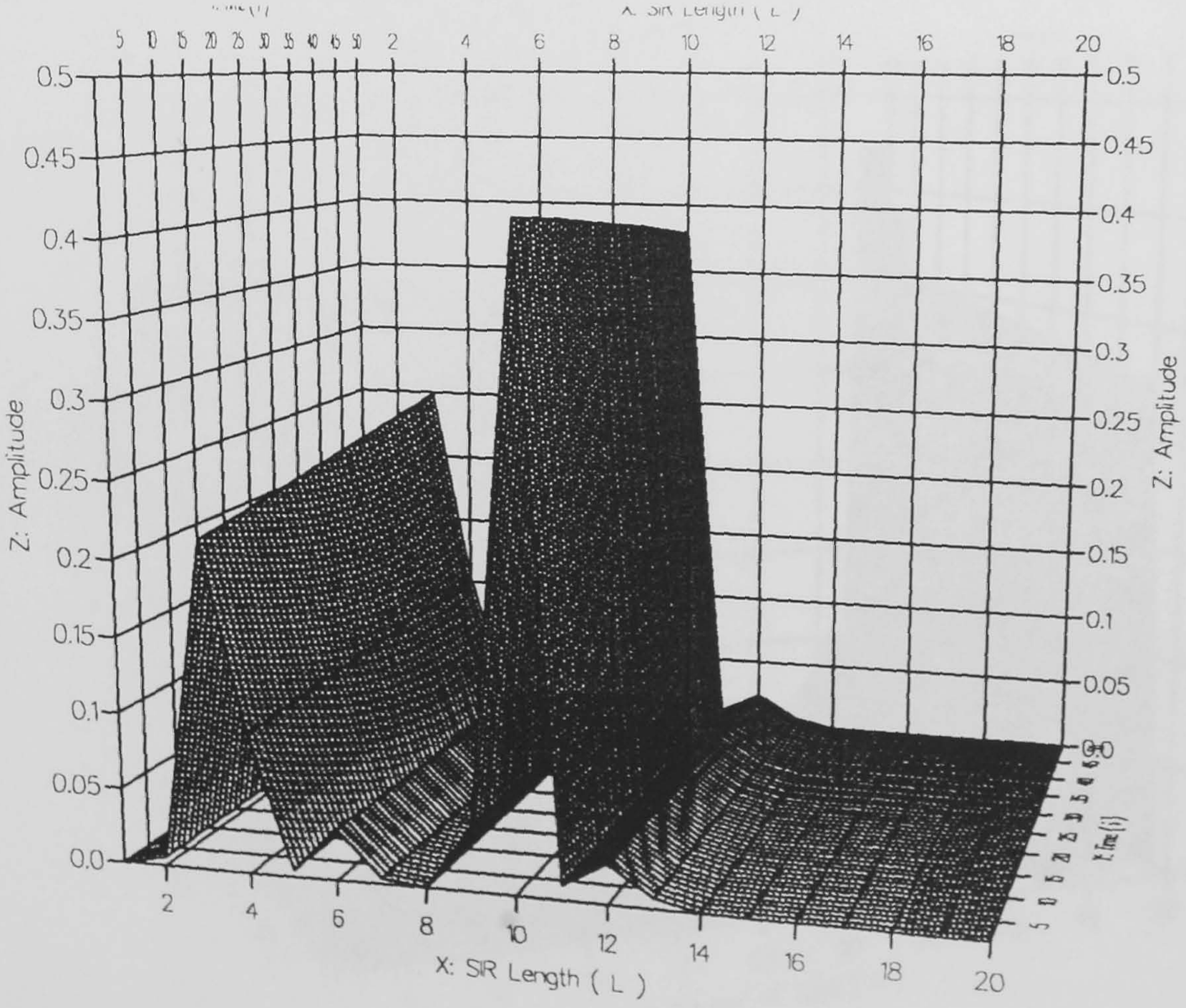
-After Minimum Phase.

b : A Segment of the Amplitude Variation of Channel 1 SIR after MP at : (i = 6000-6050).



(i=6000-6050)

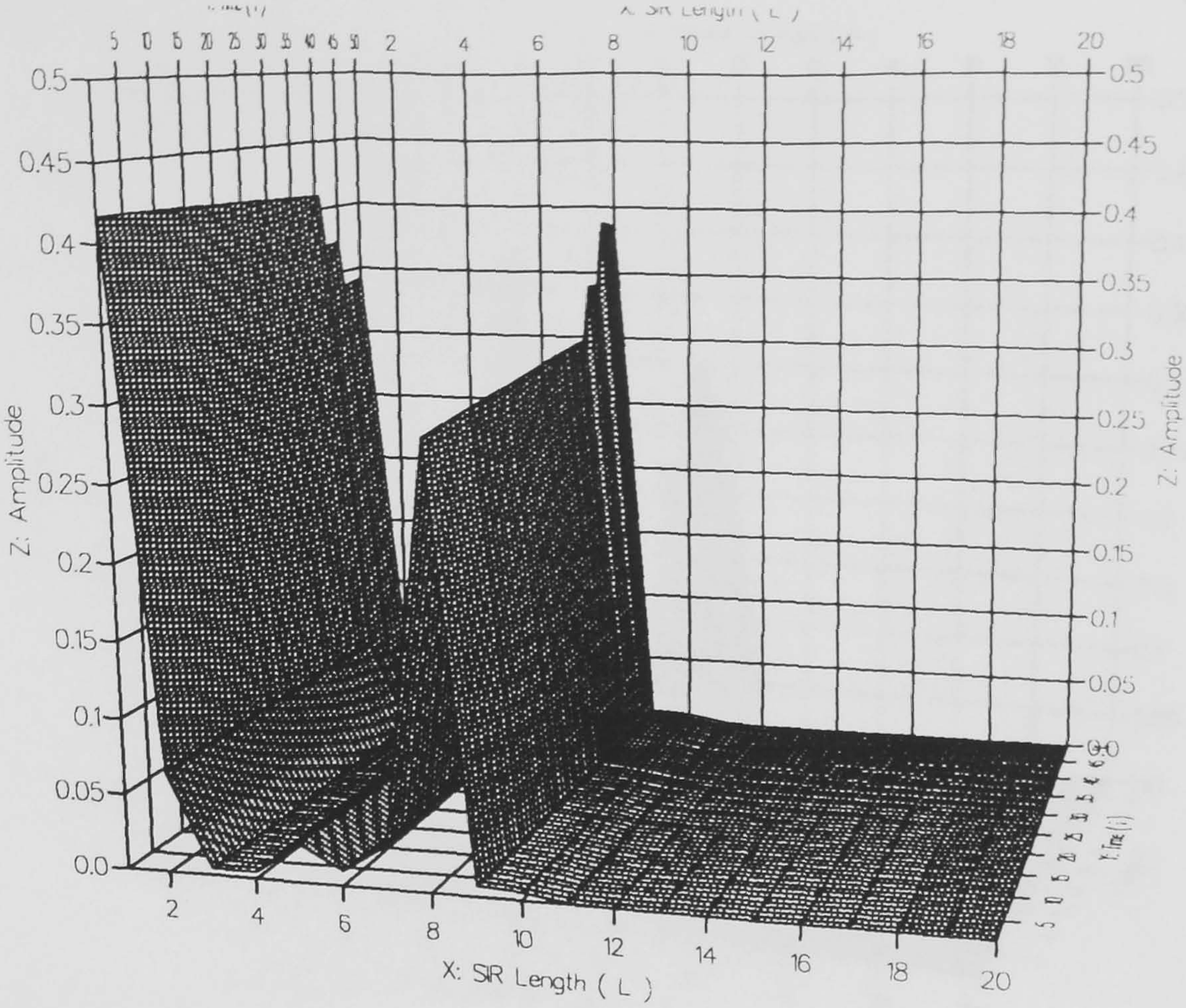
c: A Segment of the Adaptive Filter Tap-gains setting for Channel 1 at : (i = 6000-6050).



(i=9600-9650)

-Before Minimum Phase

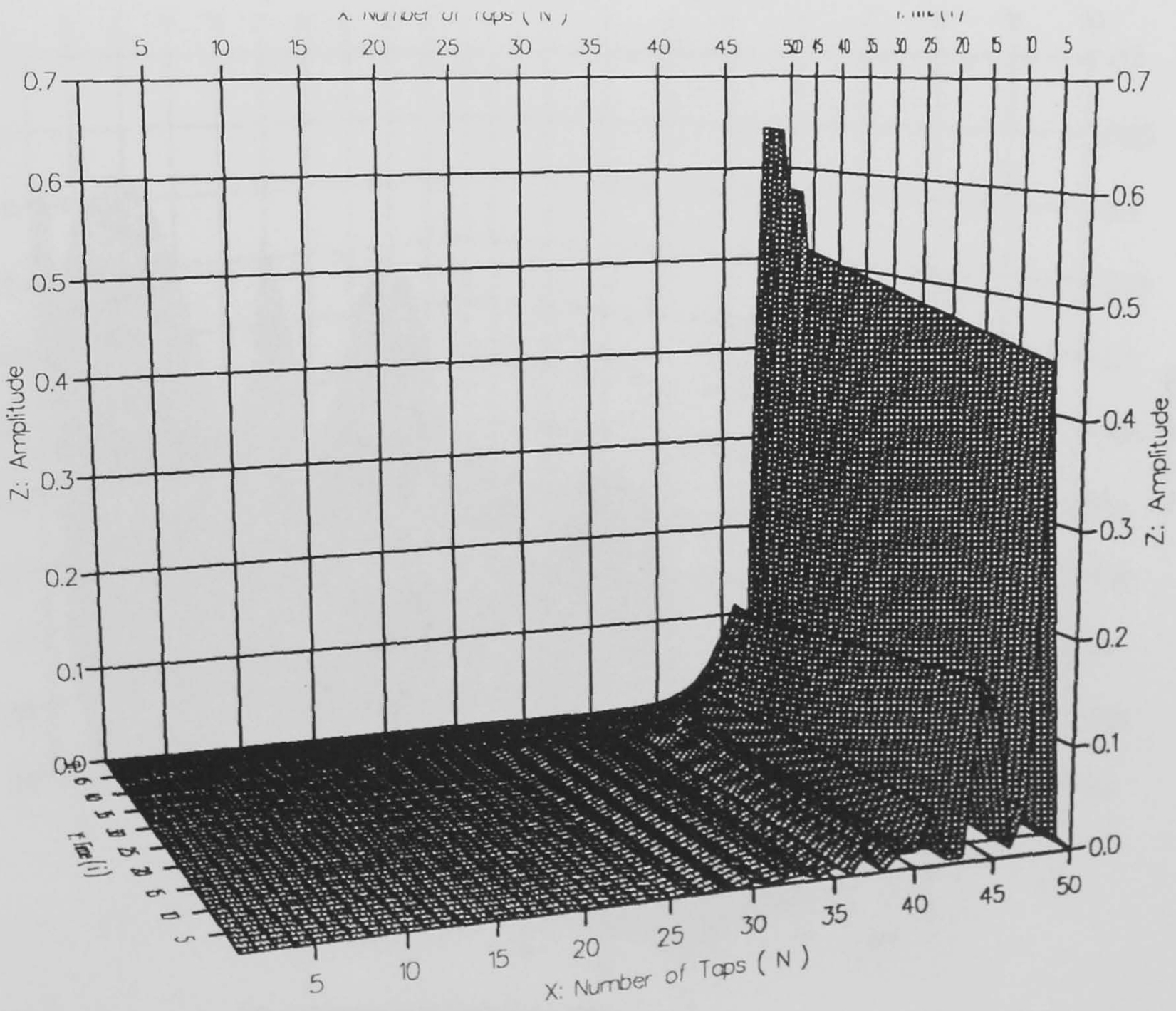
a : A Segment of the Amplitude Variation of Channel 1 SIR before MP at : (i = 9600-9650).



(i=9600-9650)

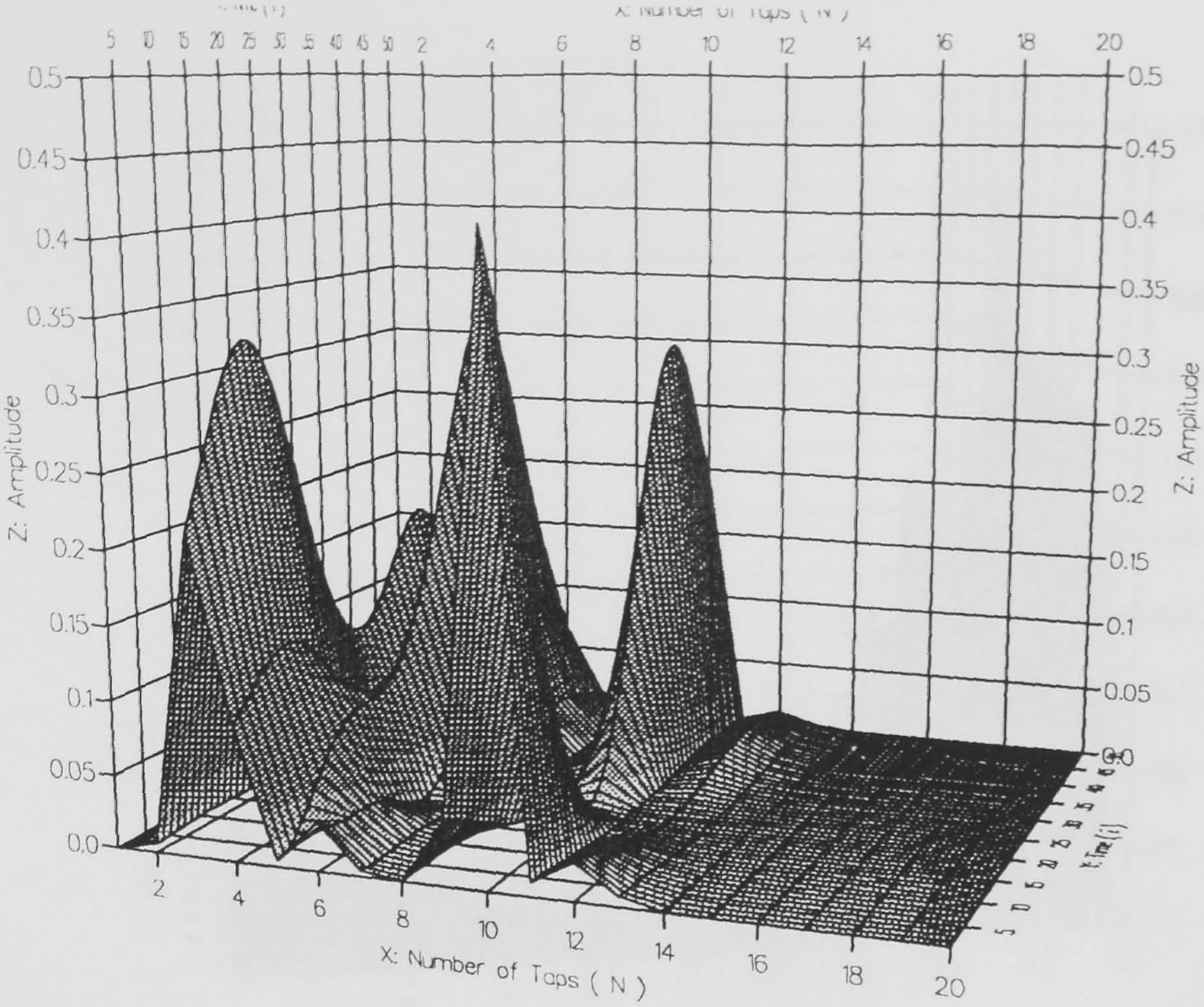
-After Minimum Phase

b : A Segment of the Amplitude Variation of Channel 1 SIR after MP at : (i = 9600-9650).



(i=9600-9650)

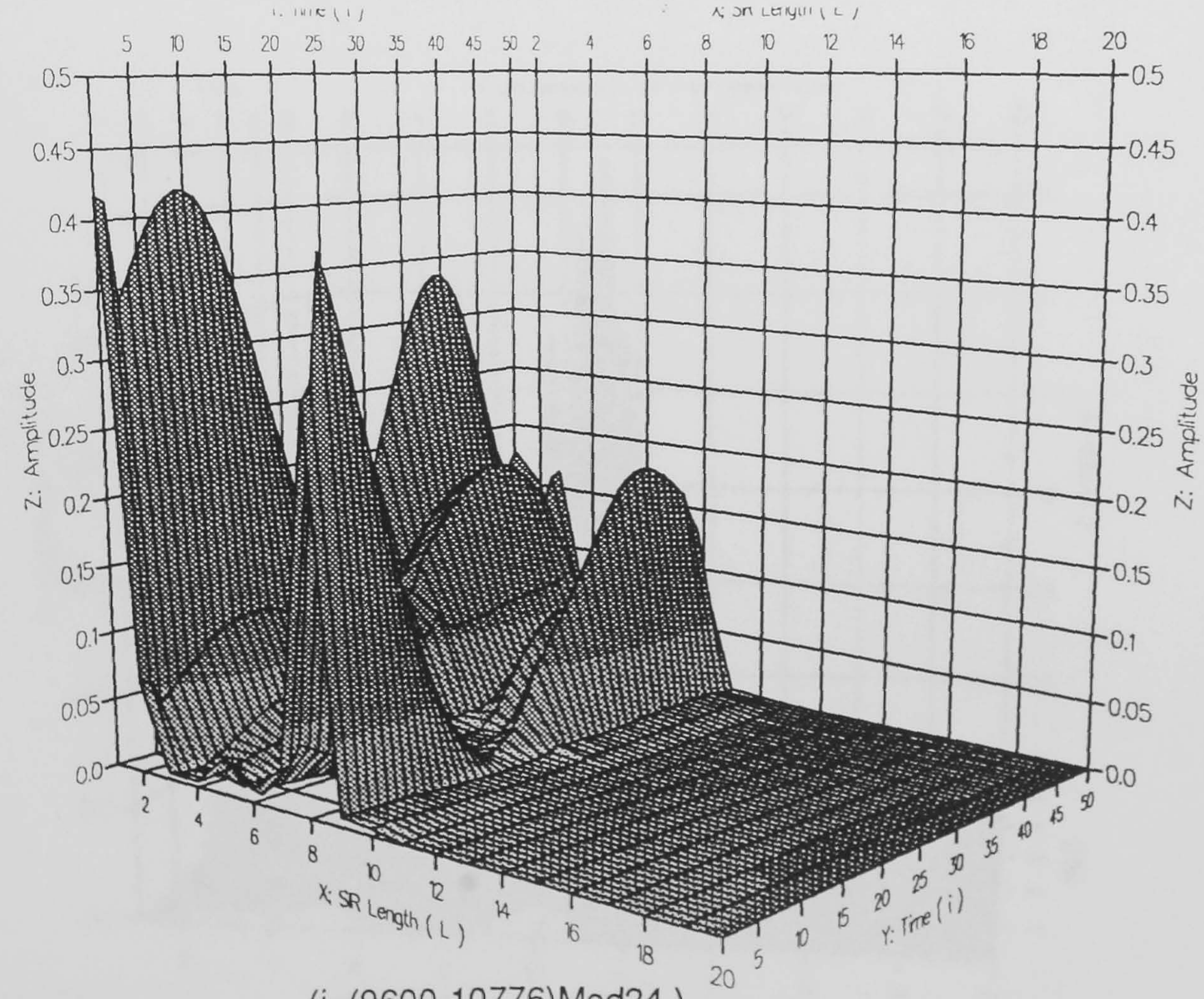
c: A Segment of the Adaptive Filter Tap-gains setting , for Channel 1 at : (i = 9600-9650).



(i=(9600-10776)Mod24)

-Before Minimum Phase

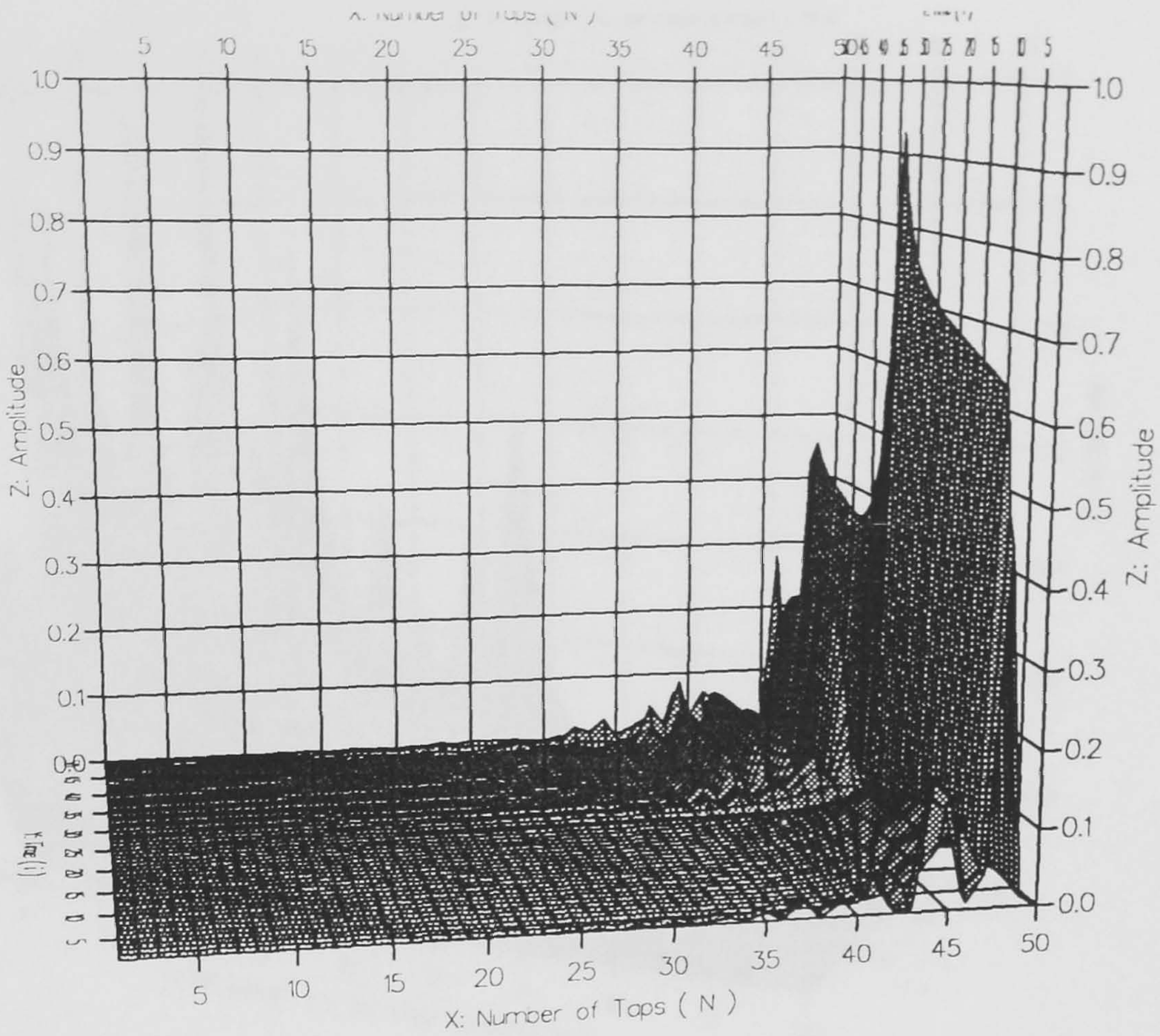
a : A Segment of the Amplitude Variation of Channel 1 SIR before MP at : (i = (9600-10776)mod 24).



(i=(9600-10776)Mod24)

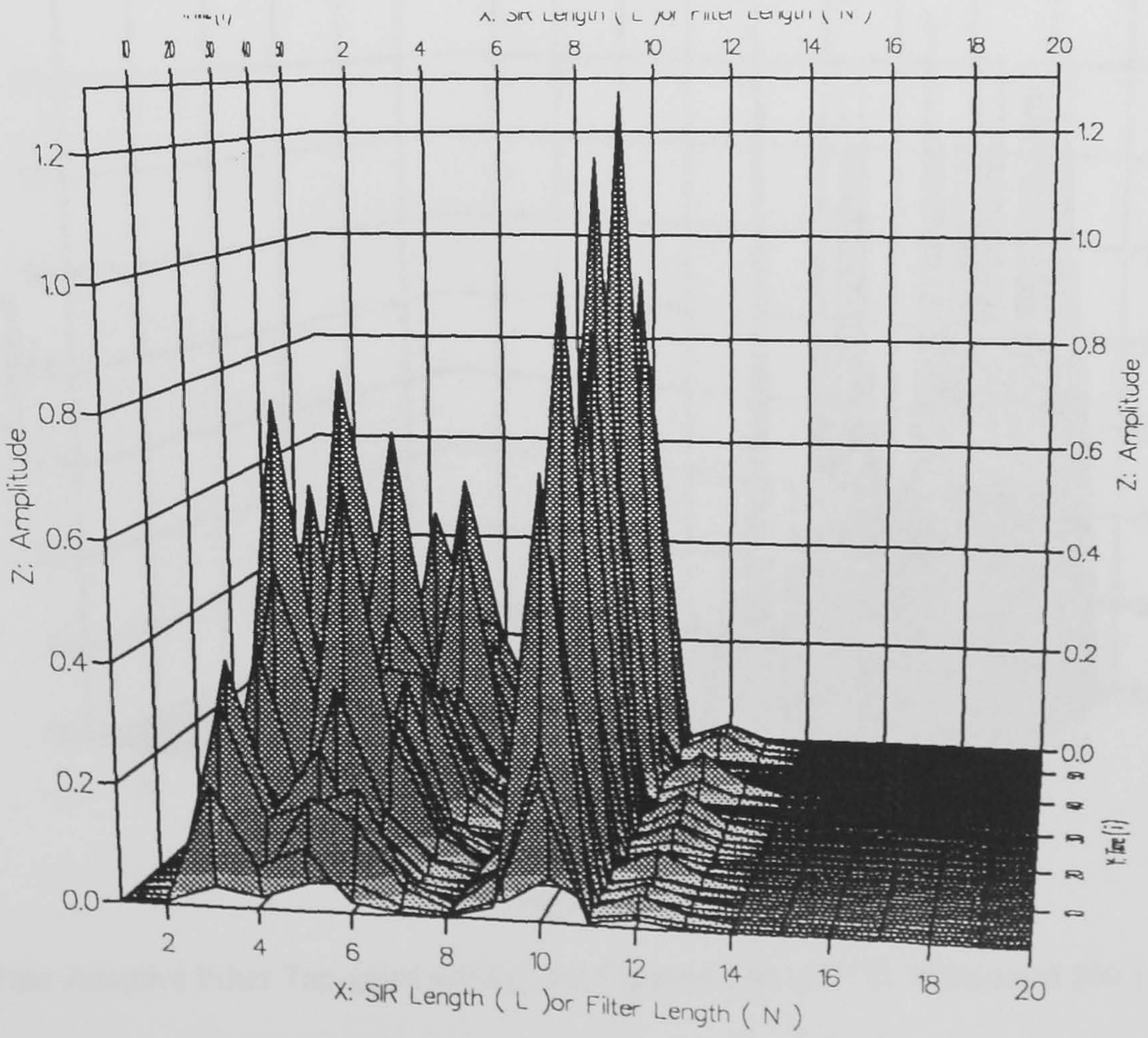
-After Minimum Phase.

b : A Segment of the Amplitude Variation of Channel 1 SIR after MP at : (i = (9600-10776)mod 24).

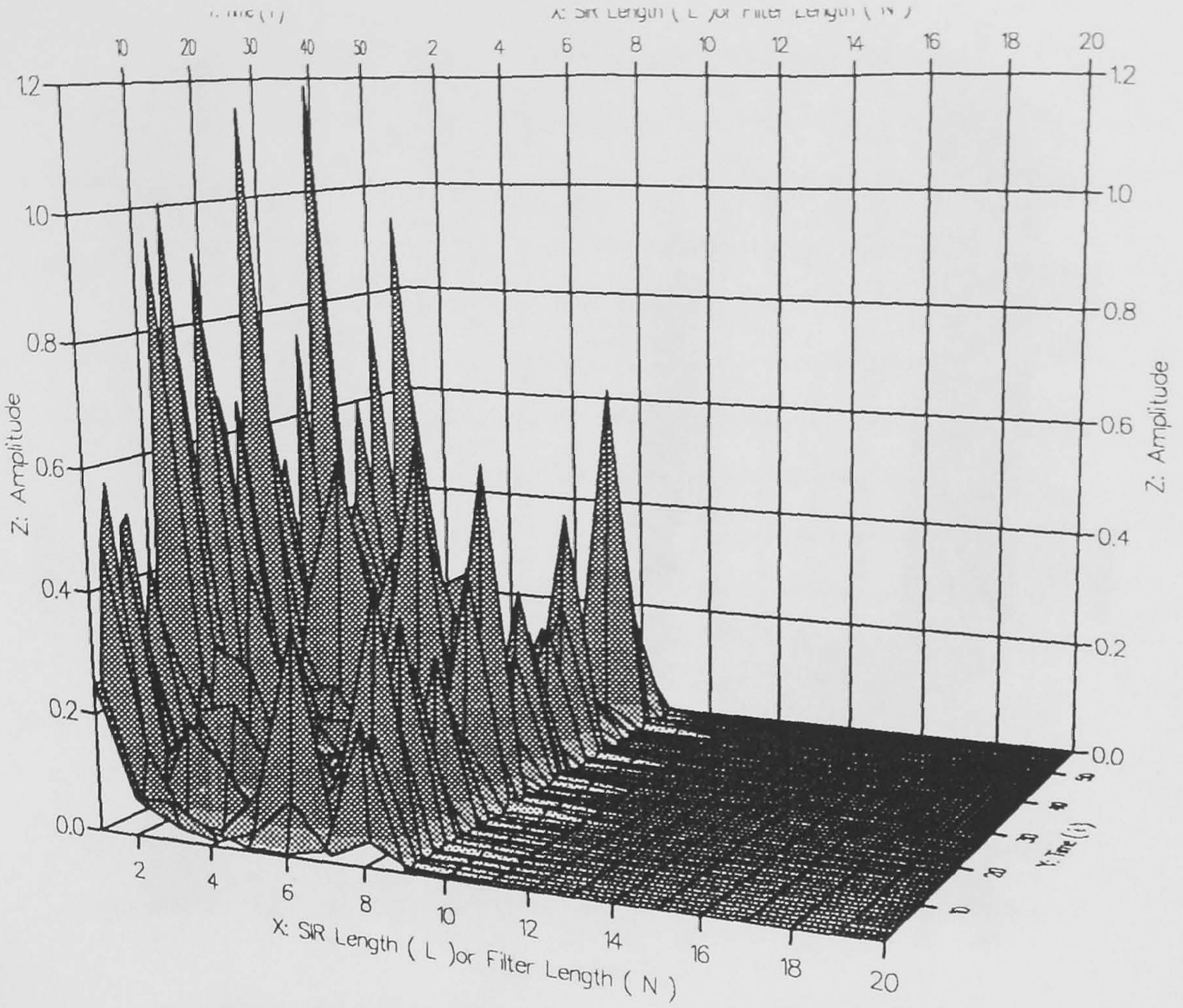


(i=(9600-10776)Mod24)

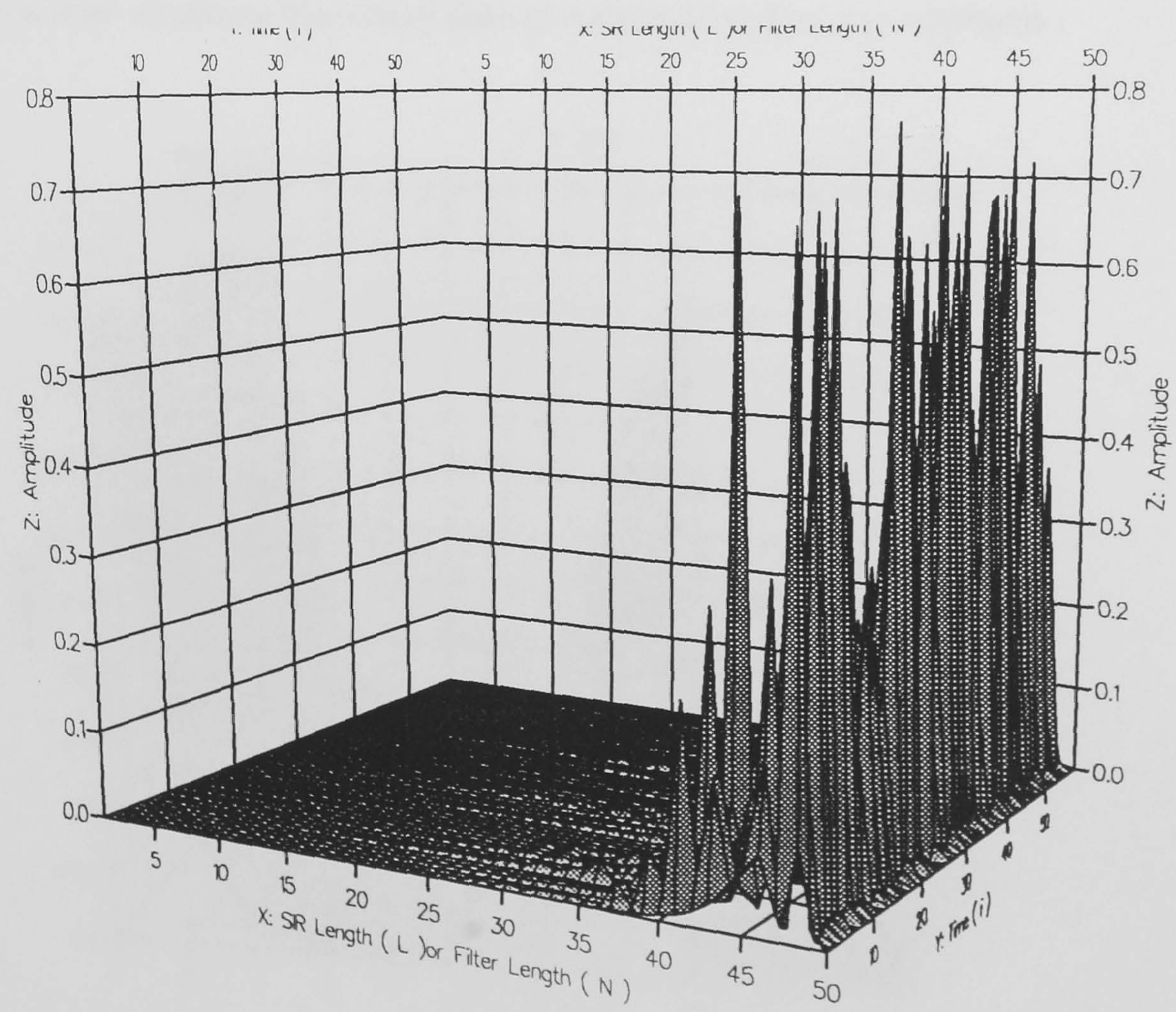
c: A Segment of the Adaptive Filter Tap-gains setting , for Channel 1 at : (i = (9600-10776)mod 24).



a : A Segment of the Amplitude Variation of Channel 1 SIR before MP at : (i = (1-55680)mod 960).

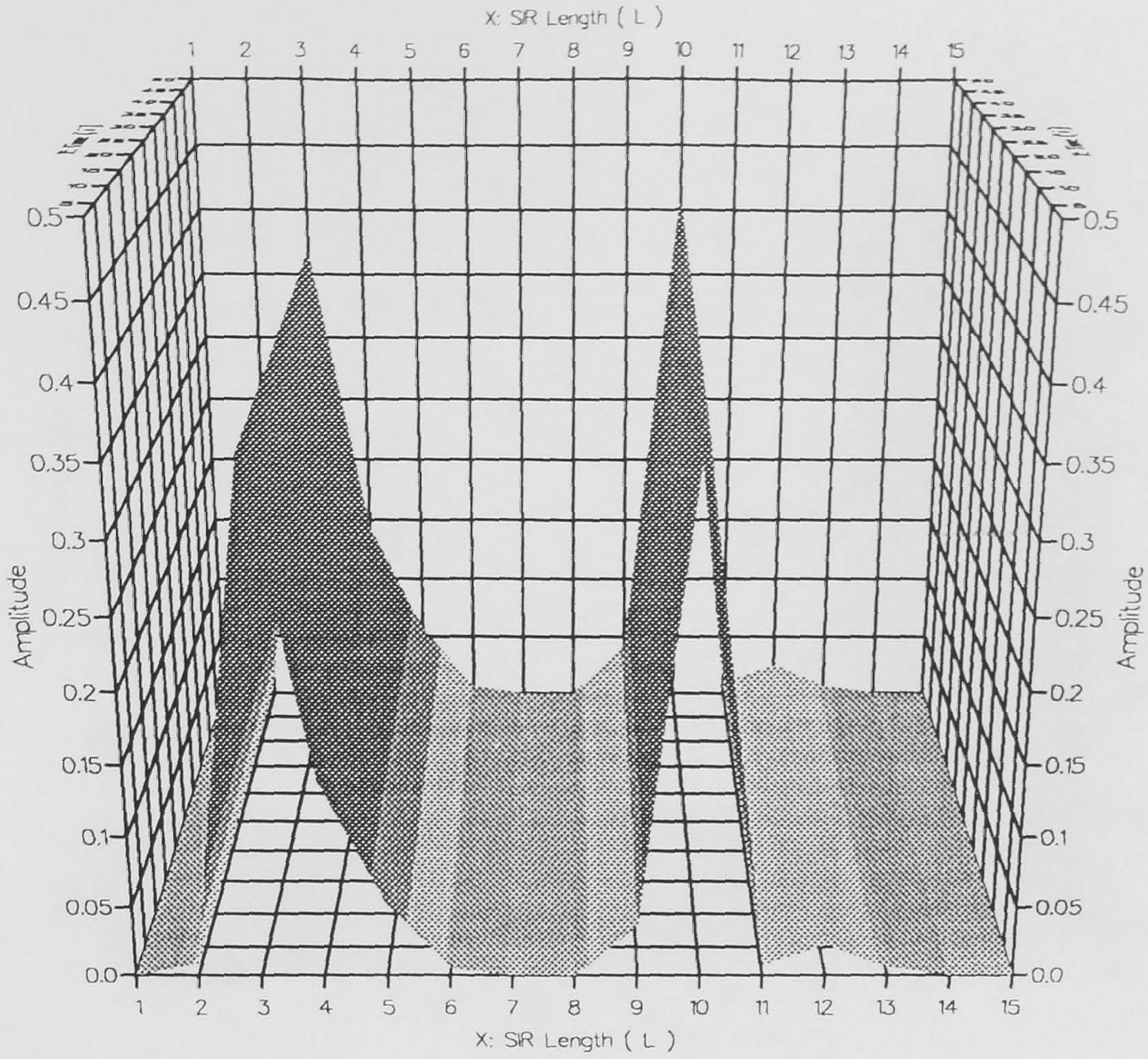


b : A Segment of the Amplitude Variation of Channel 1 SIR after MP at : (i = (1-55680)mod 960).



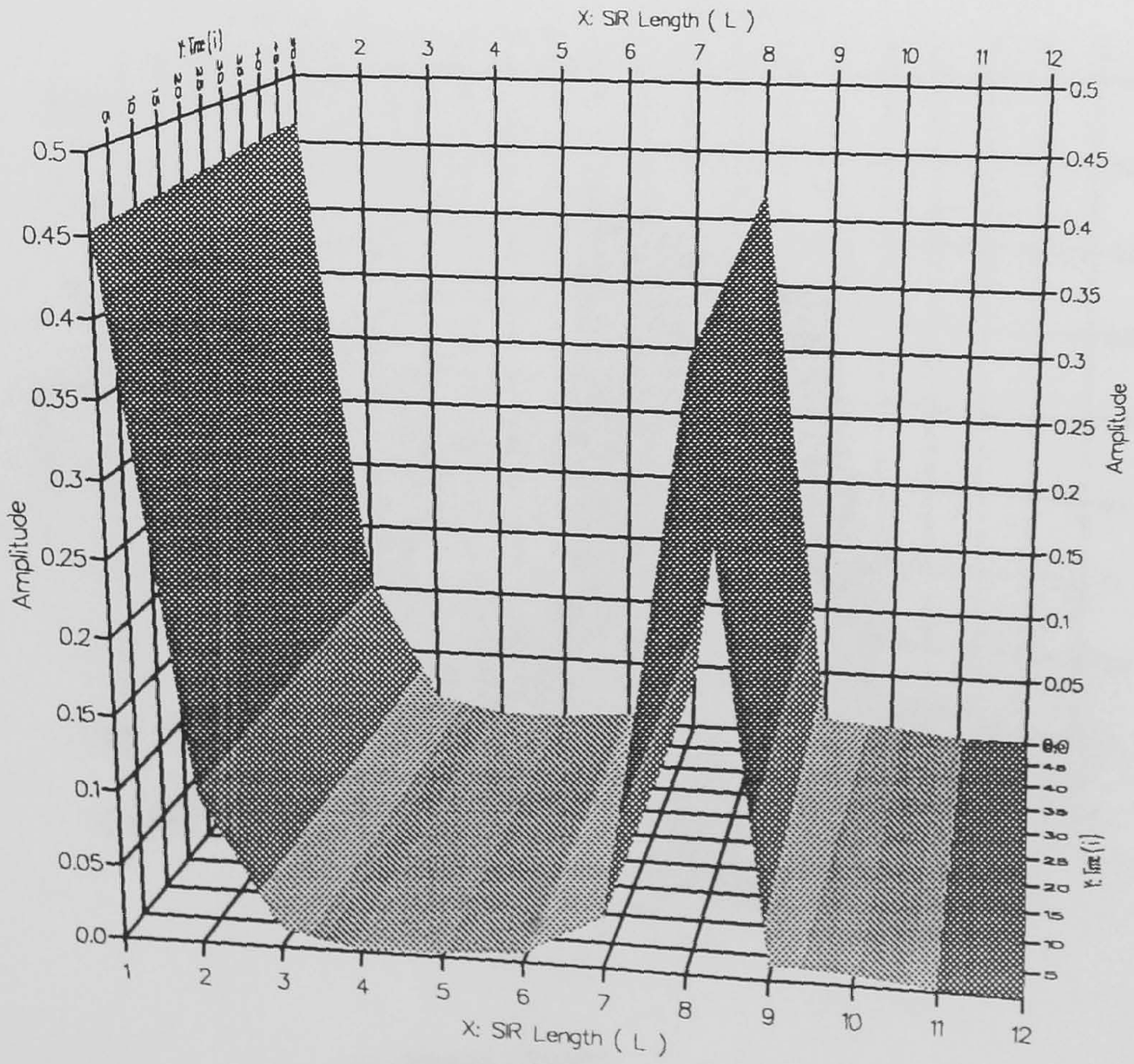
c: A Segment of the Adaptive Filter Tap-gains setting , for Channel 1 at : (i = (1-55680)mod 960).

Fig.4.5.1: Segments of the Amplitude Variation of Non-MP, MP SIR and Tap-gains setting , for Channel 1 at different durations.



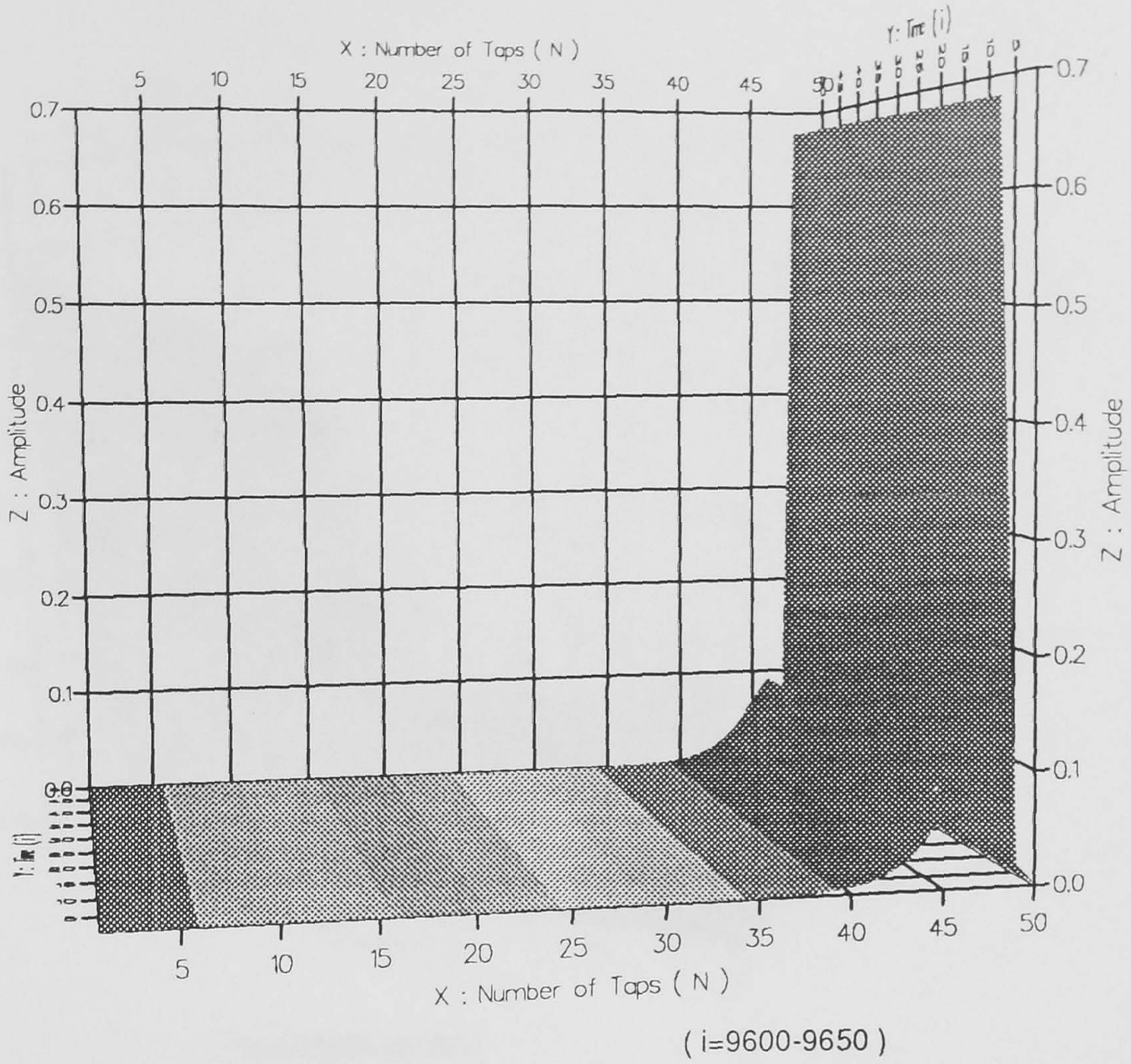
-Before Minimum Phase. (i=9600-9650)

a : A Segment of the Amplitude Variation of Channel 2 SIR before MP at : (i = 9600-9650).



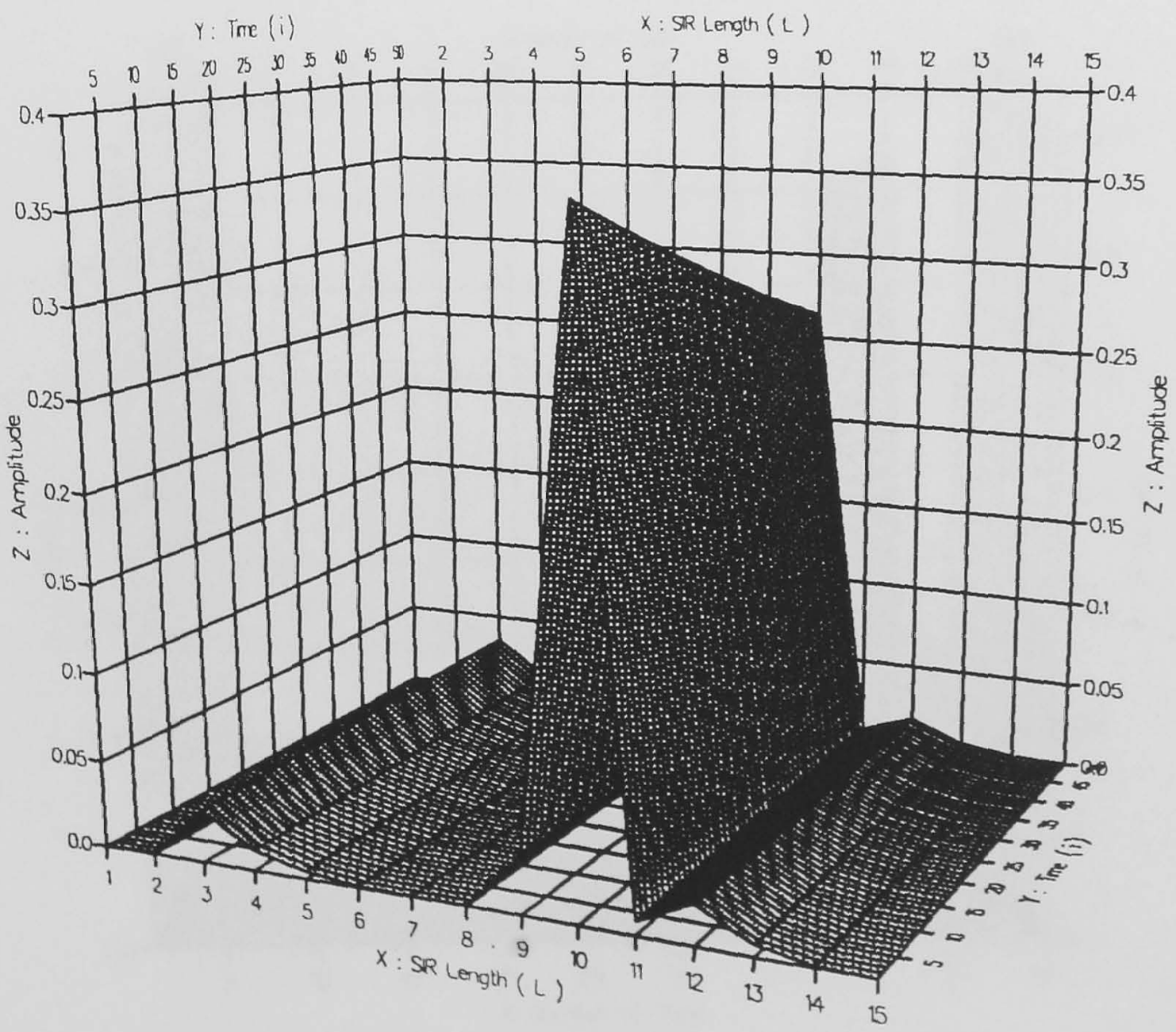
-After Minimum Phase. (i=9600-9650)

b : A Segment of the Amplitude Variation of Channel 2 SIR after MP at : (i = 9600-9650).



(i=9600-9650)

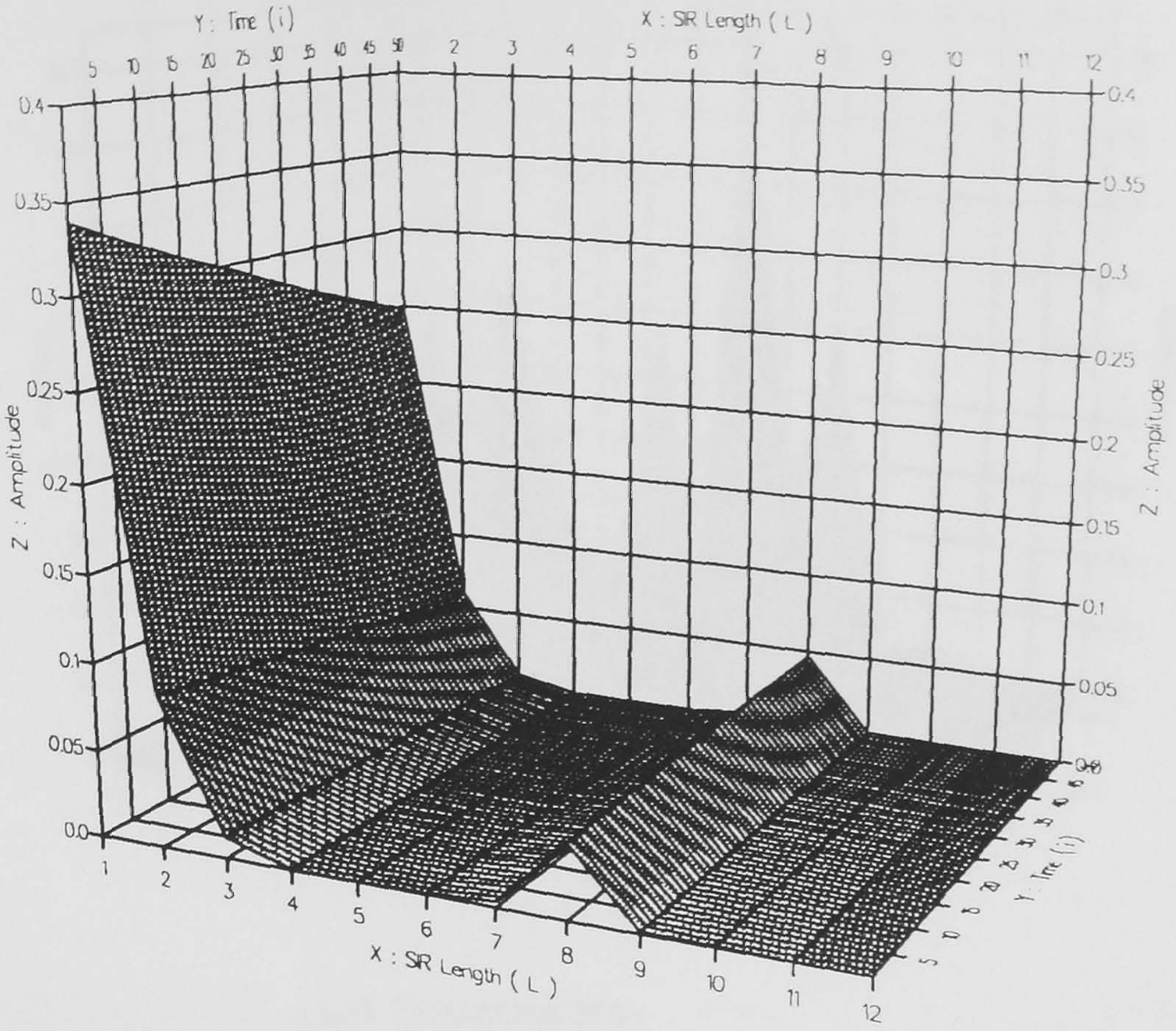
c: A Segment of the Adaptive Filter Tap-gains setting for Channel 2 at : (i = 9600-9650).



(i=30600-30650)

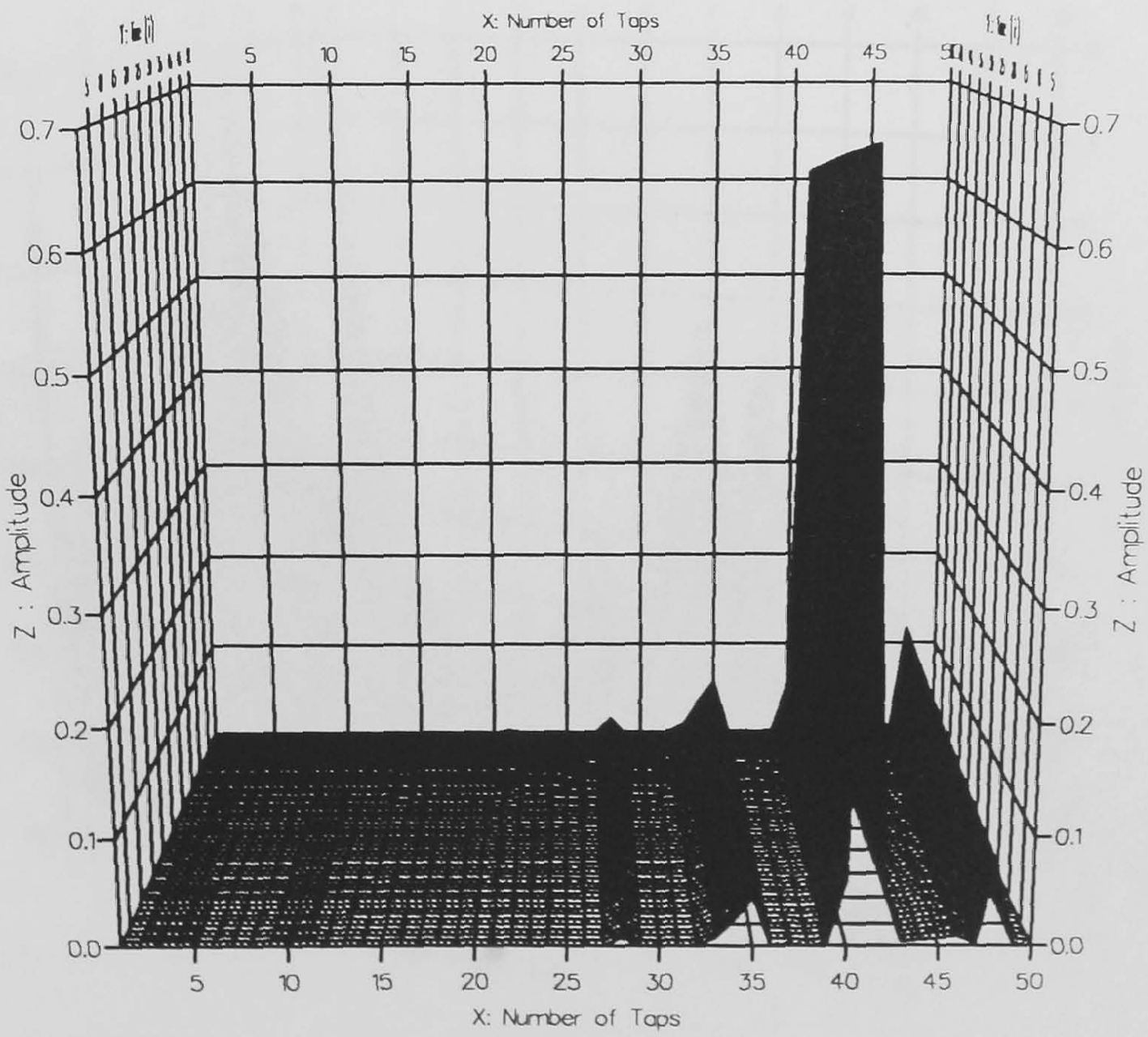
-Before Minimum Phase.

a : A Segment of the Amplitude Variation of Channel 2 SIR before MP at : (i = 30600-30650).



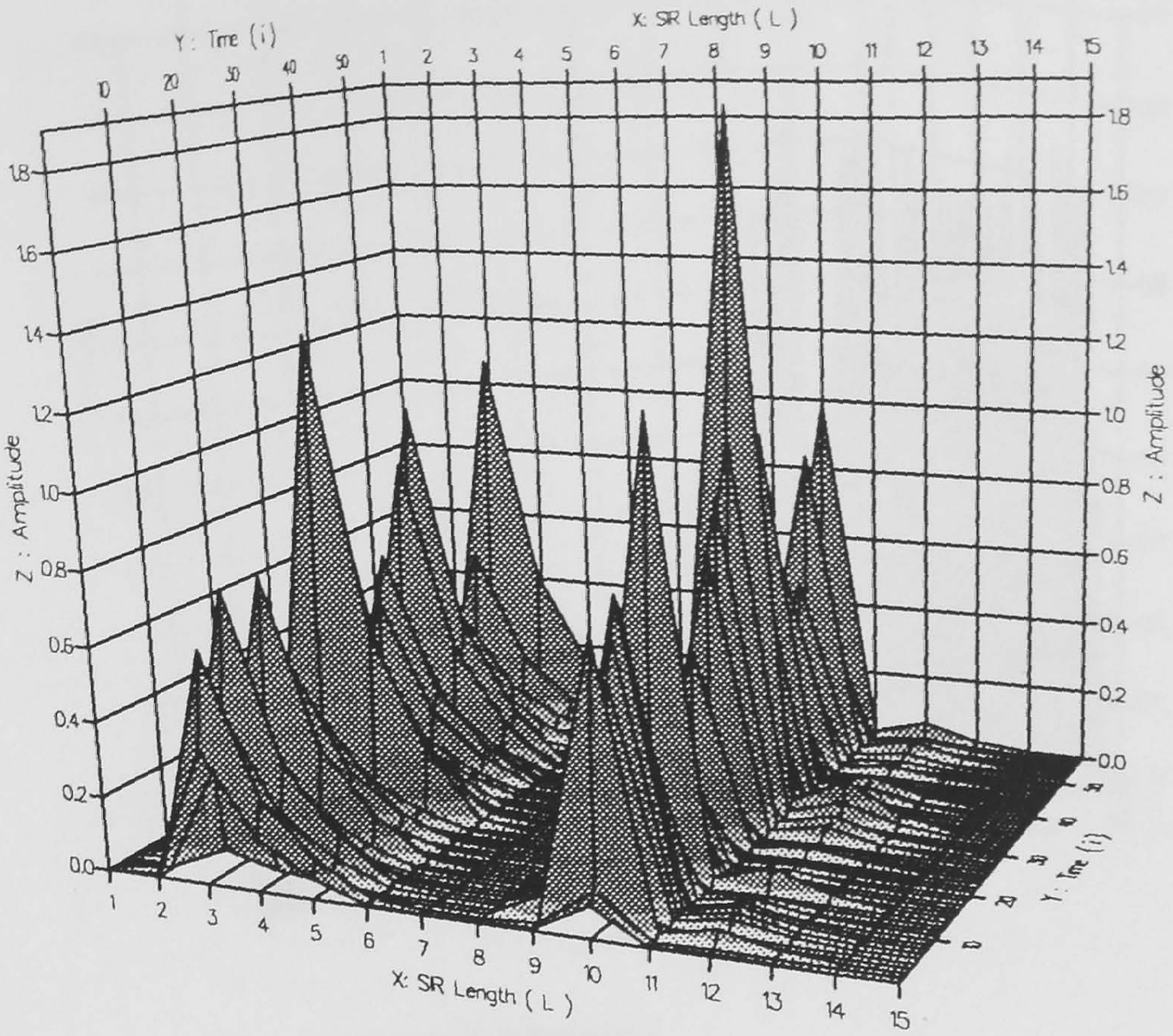
(i=30600-30650)

b : A Segment of the Amplitude Variation of Channel 2 SIR after MP at : (i = 30600-30650).



(i=30600-30650)

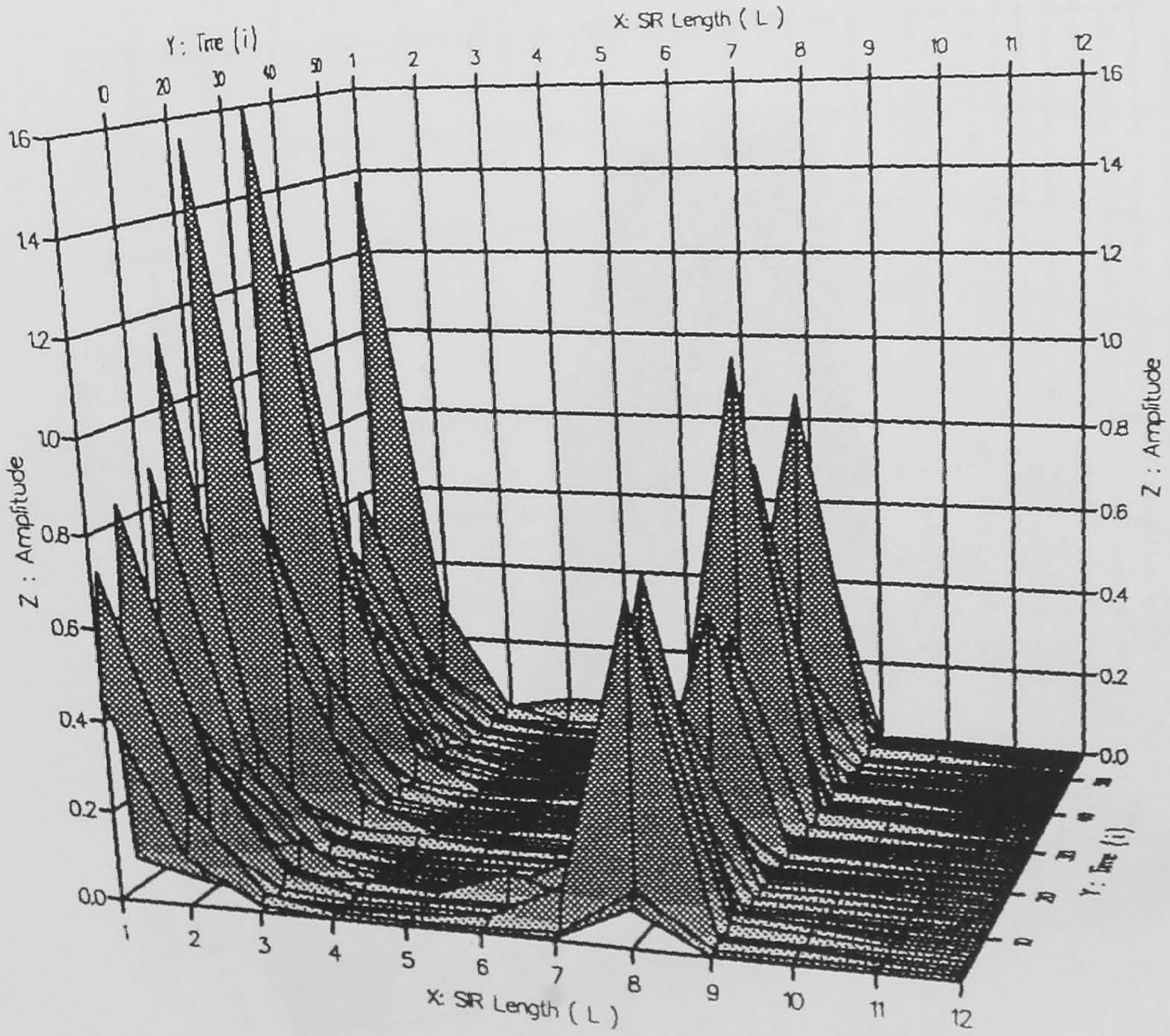
c: A Segment of the Adaptive Filter Tap-gains setting , for Channel 2 at : (i = 30600-30650).



(i=(1-55680)Mod.960)

-Before Minimum Phase.

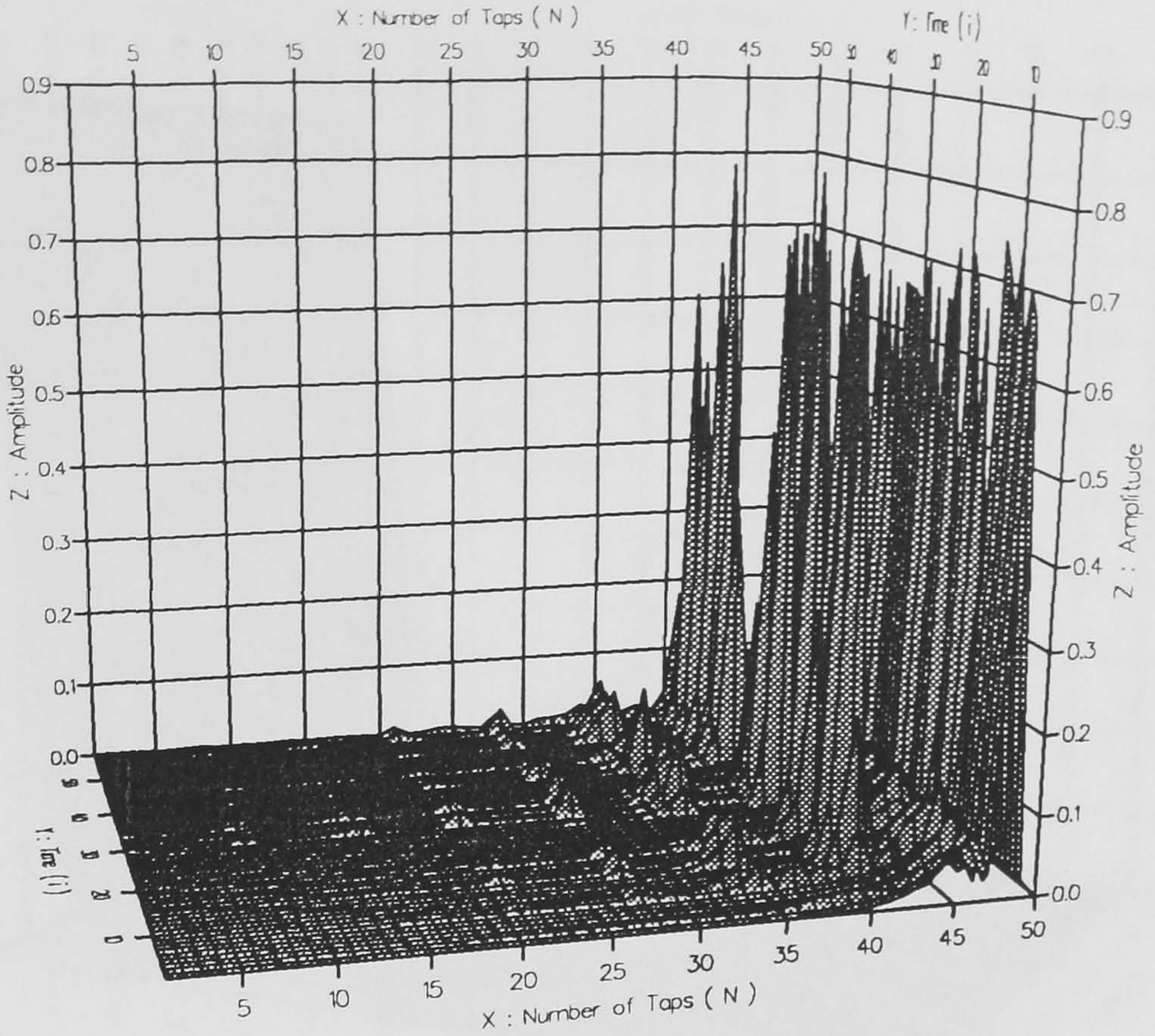
a : A Segment of the Amplitude Variation of Channel 2 SIR before MP at : (i = (1-55680)mod 960).



(i=(1-55680)Mod.960)

-After Minimum Phase.

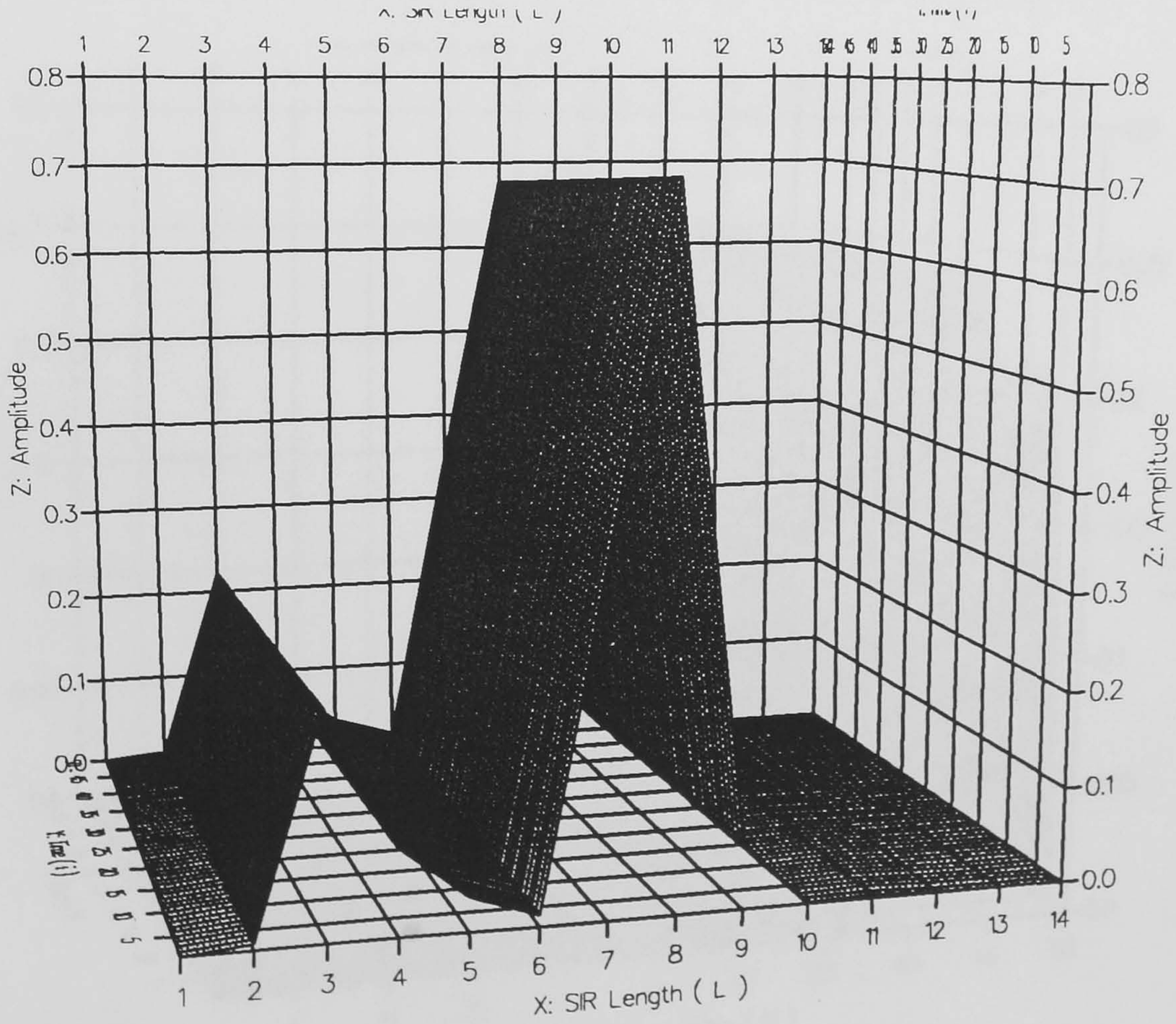
b : A Segment of the Amplitude Variation of Channel 2 SIR after MP at : (i = (1-55680)mod 960).



(i=(1-55680)Mod.960)

c: A Segment of the Adaptive Filter Tap-gains setting , for Channel 2 at : (i = (1-55680)mod 960).

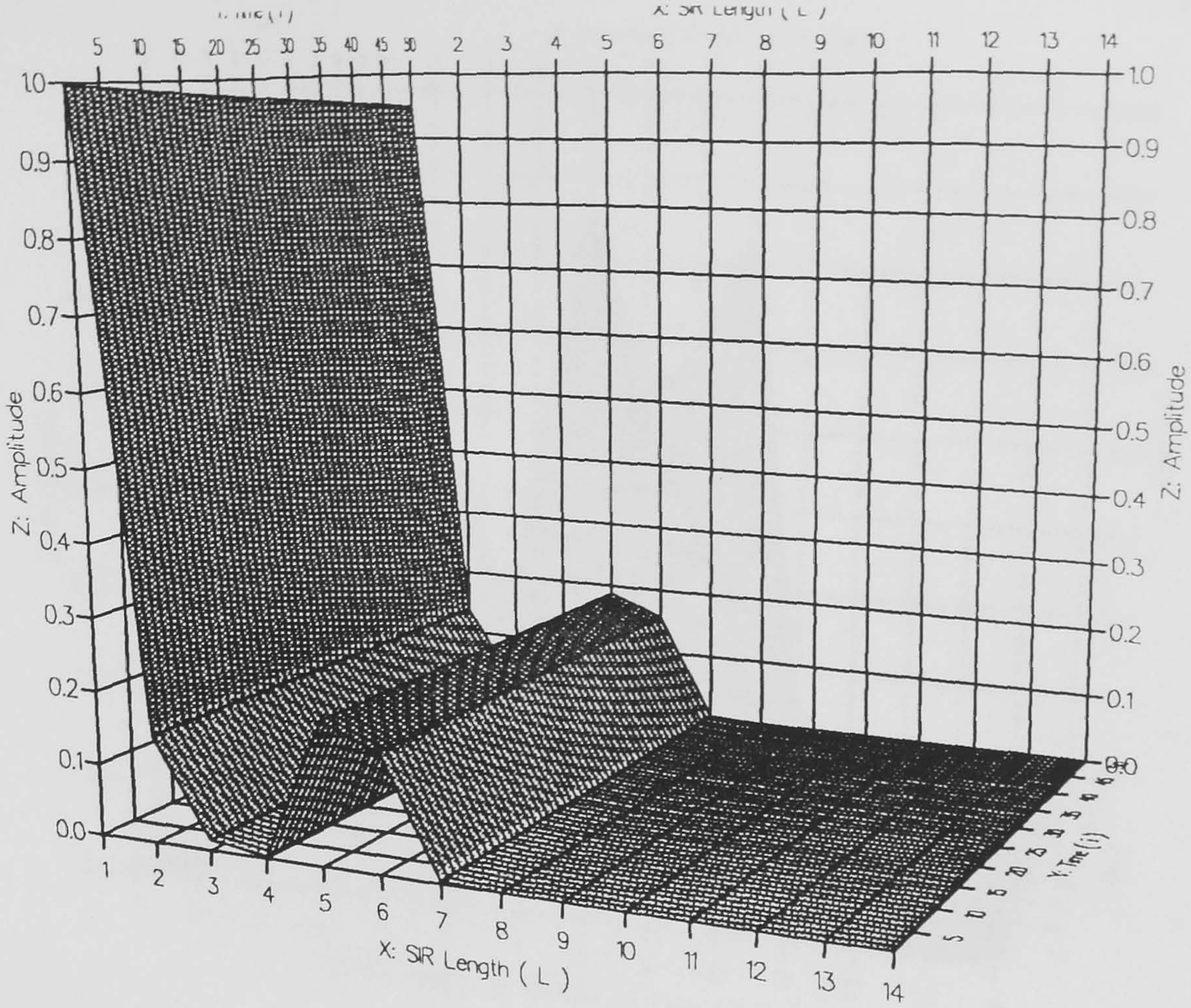
Fig.4.5.2 : Segments of the Amplitude Variation of Non-MP, MP SIR and Tap-gains setting , for Channel 2 at different duration



(i=9600-9650)

-Before Minimum Phase.

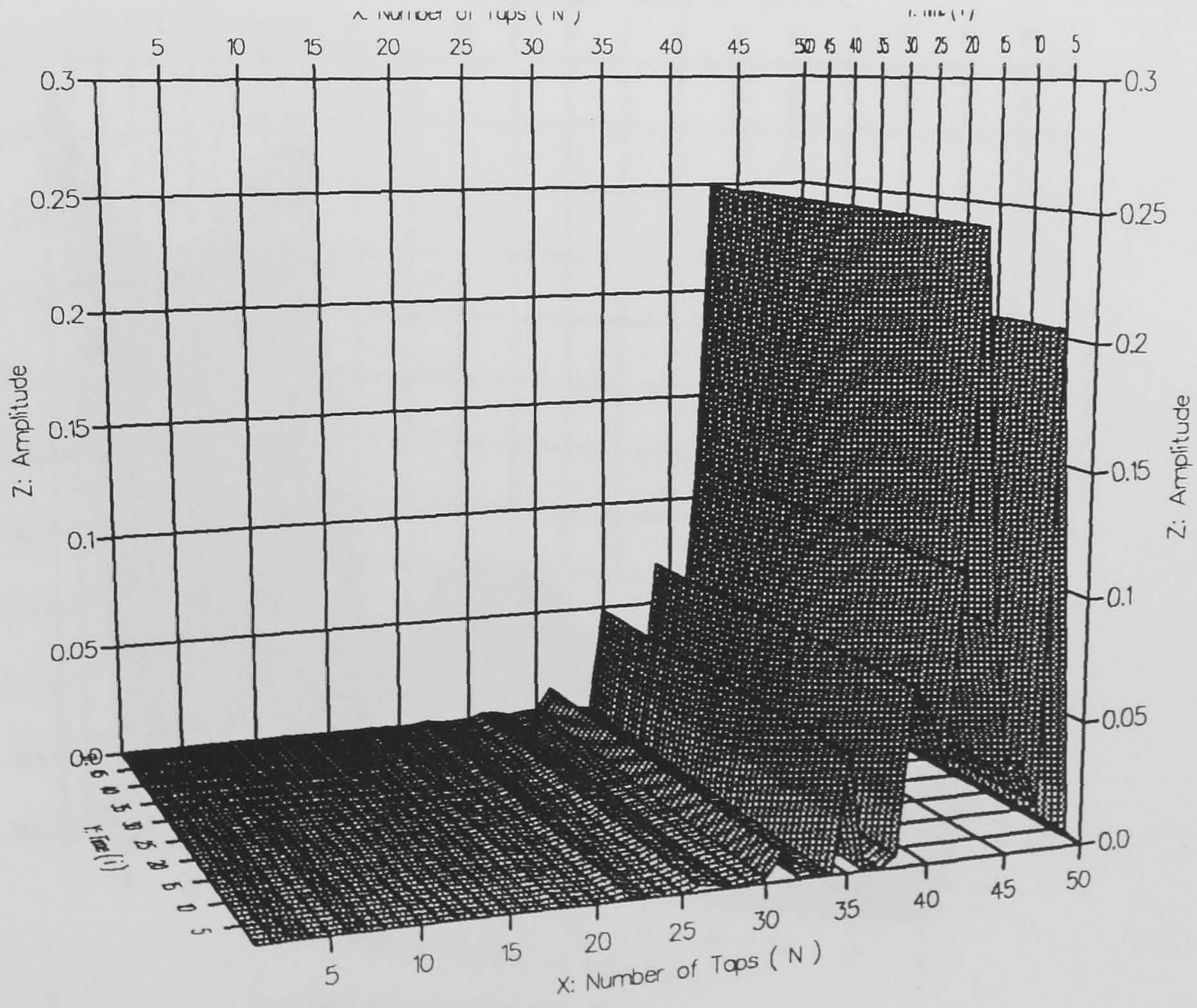
a : A Segment of the Amplitude Variation of Channel 3 SIR before MP at : (i = 9600-9650).



(i=9600-9650)

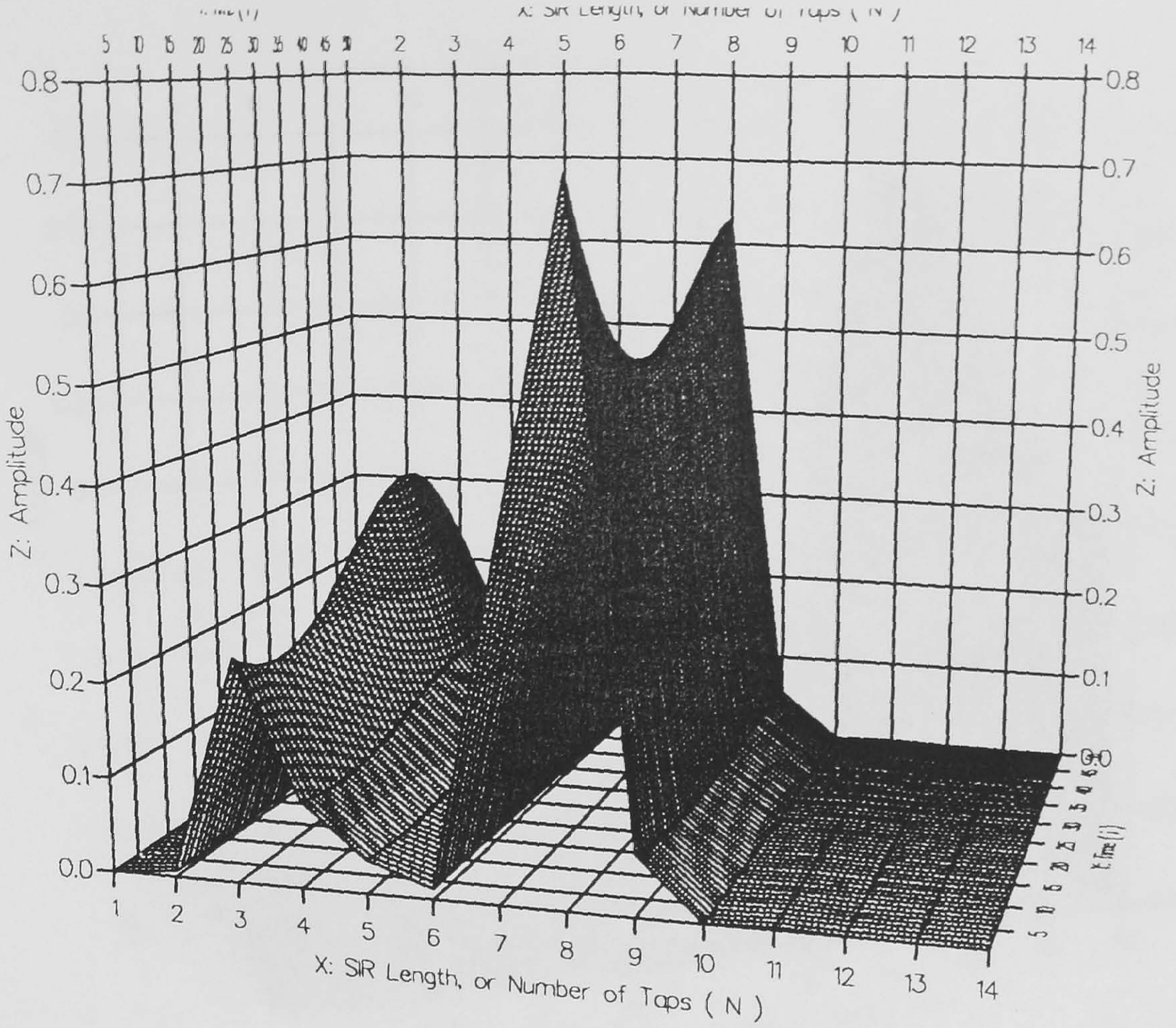
-After minimum Phase.

b : A Segment of the Amplitude Variation of Channel 3 SIR after MP at : (i = 9600-9650).



(i=9600-9650)

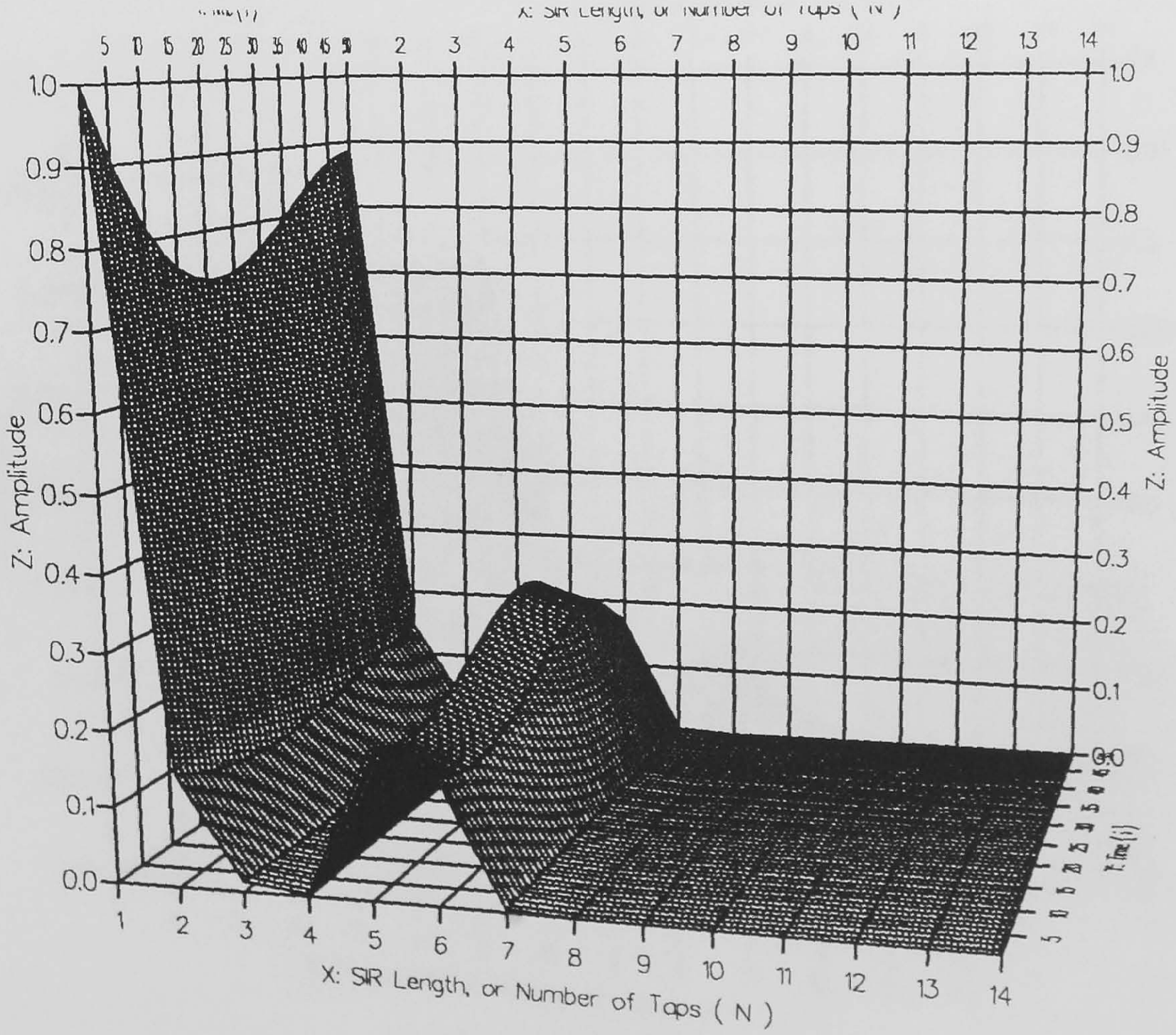
c: A Segment of the Adaptive Filter Tap-gains setting for Channel 3 at : (i = 9600-9650).



(i=(9600-10776)Mod24)

-Before Minimum Phase

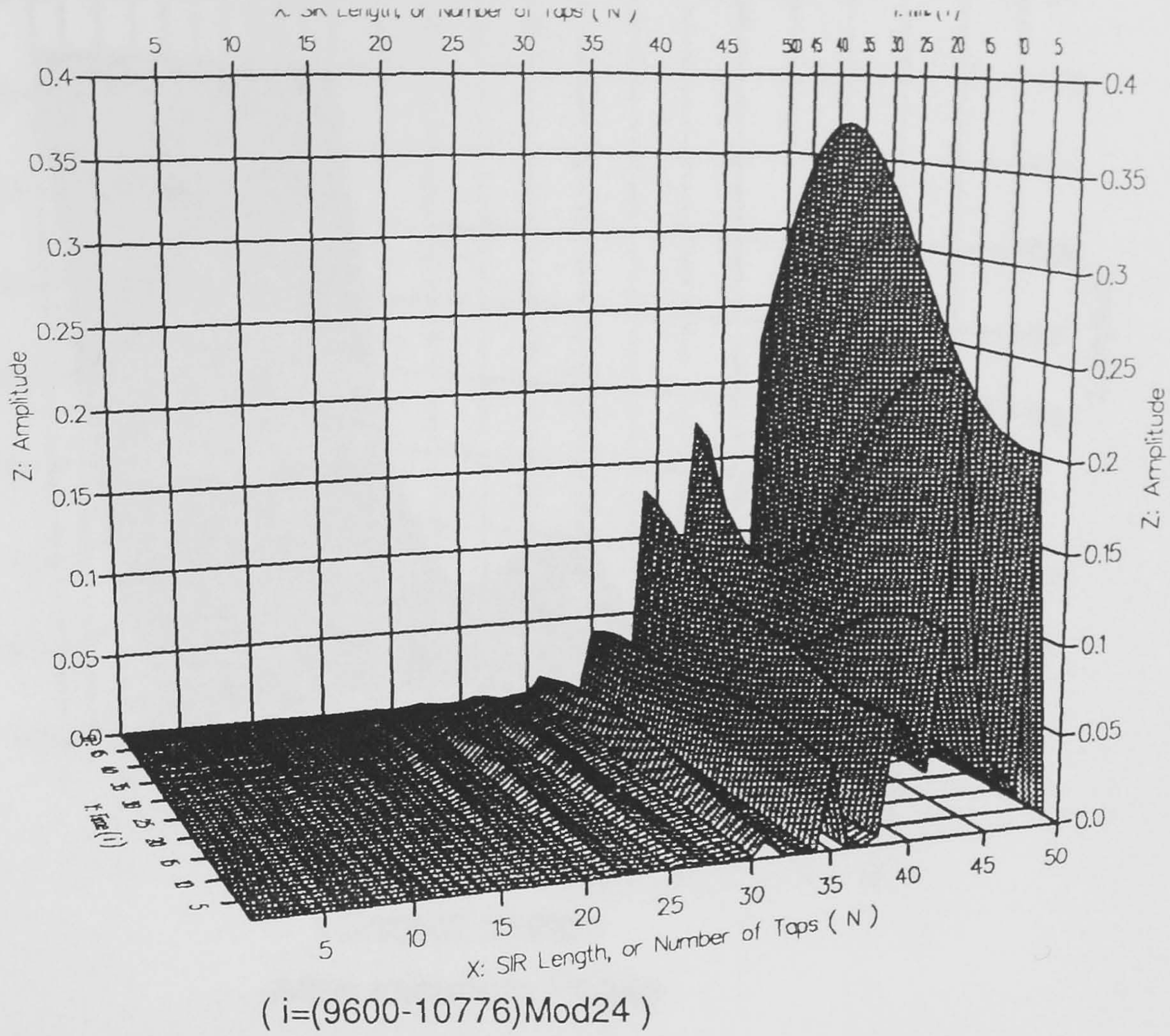
a : A Segment of the Amplitude Variation of Channel 3 SIR before MP at : (i = (9600-10776)mod 24).



(i=(9600-10776)Mod24)

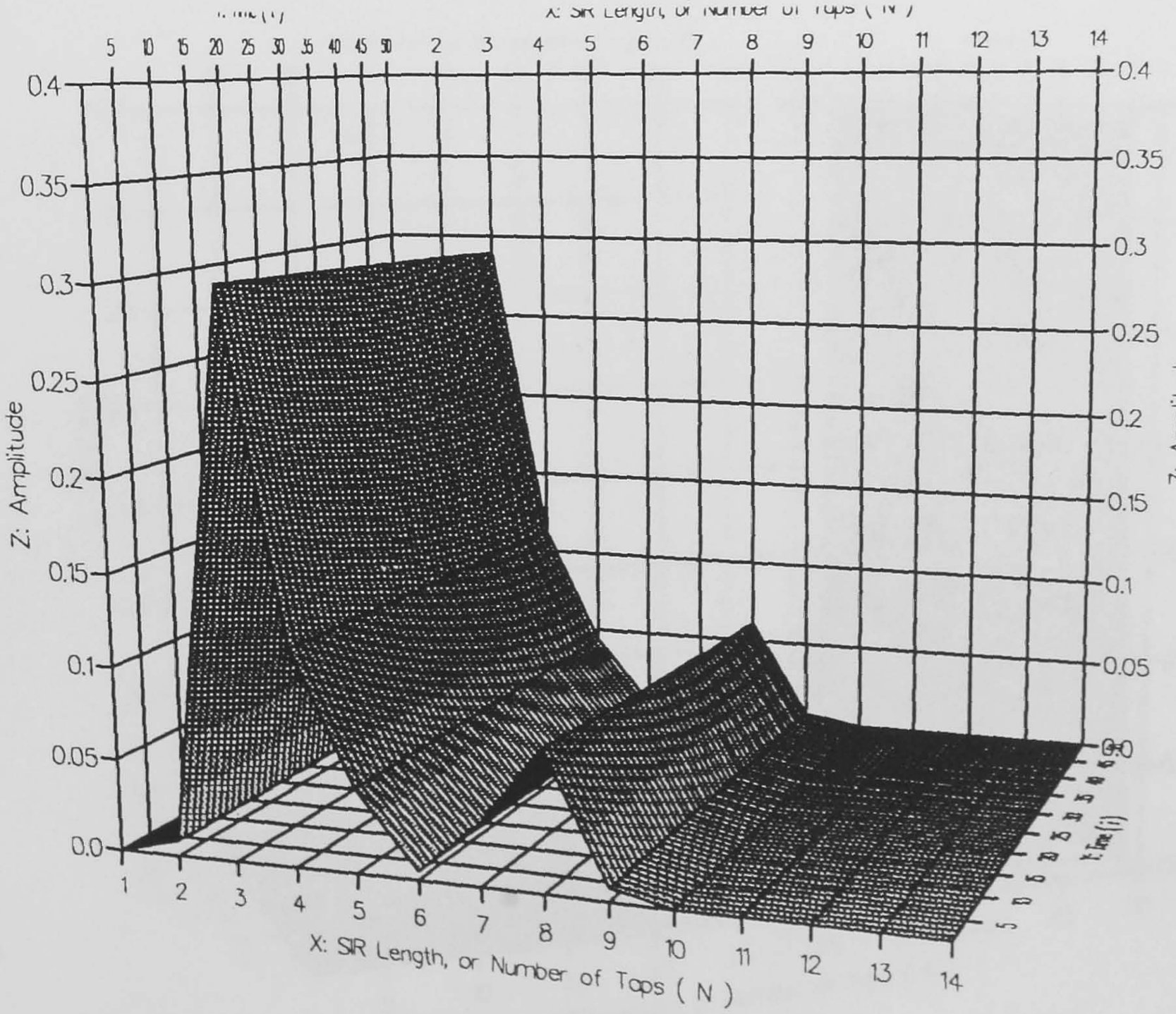
-After Minimum Phase

b : A Segment of the Amplitude Variation of Channel 3 SIR after MP at : (i = (9600-10776)mod 24).



(i=(9600-10776)Mod24)

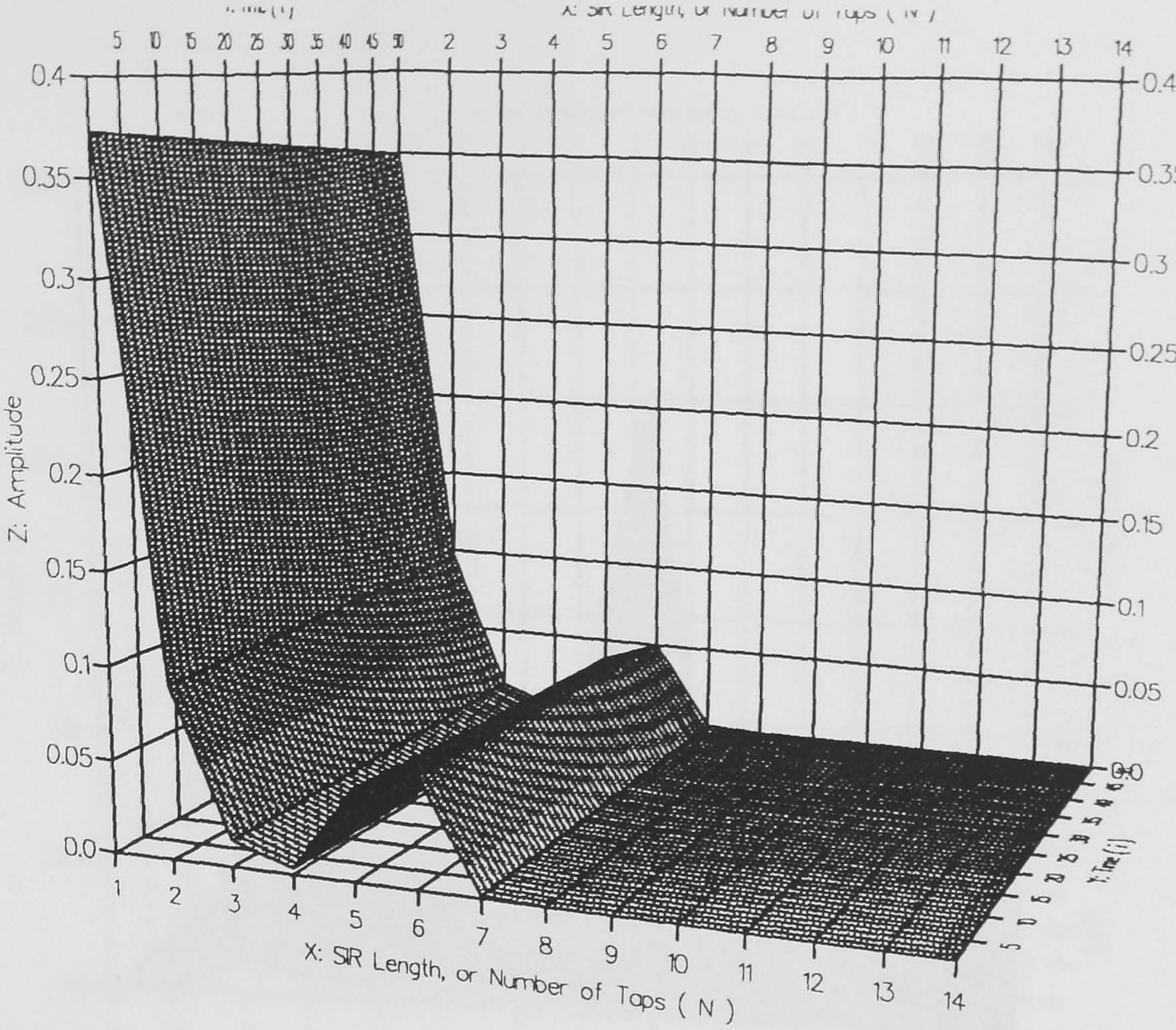
c: A Segment of the Adaptive Filter Tap-gains setting , for Channel 3 at : (i = (9600-10776)mod 24).



(i=30500-30550)

-Before Minimum Phase

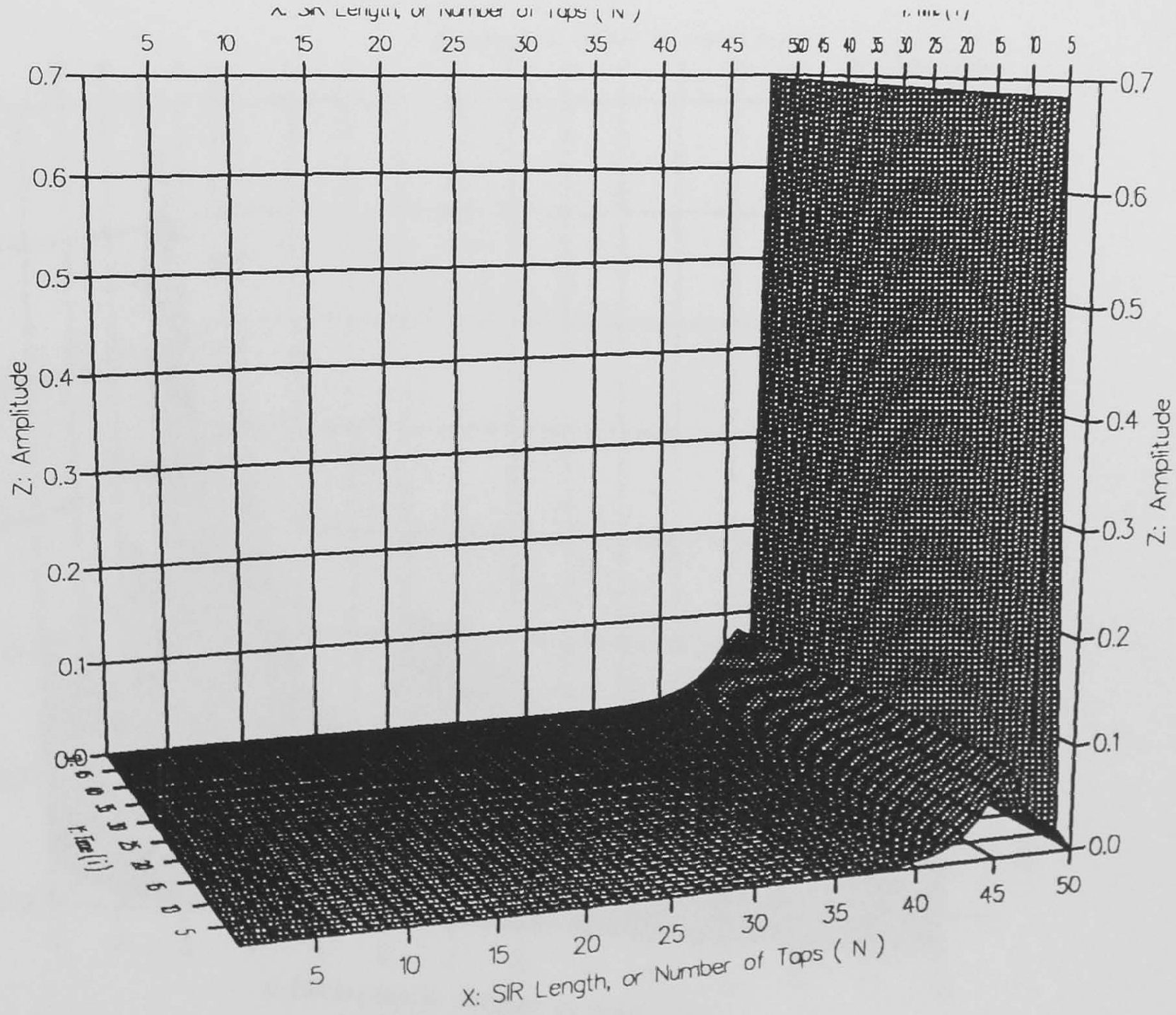
a : A Segment of the Amplitude Variation of Channel 3 SIR before MP at : (i = 30500-30550).



(i=30500-30550)

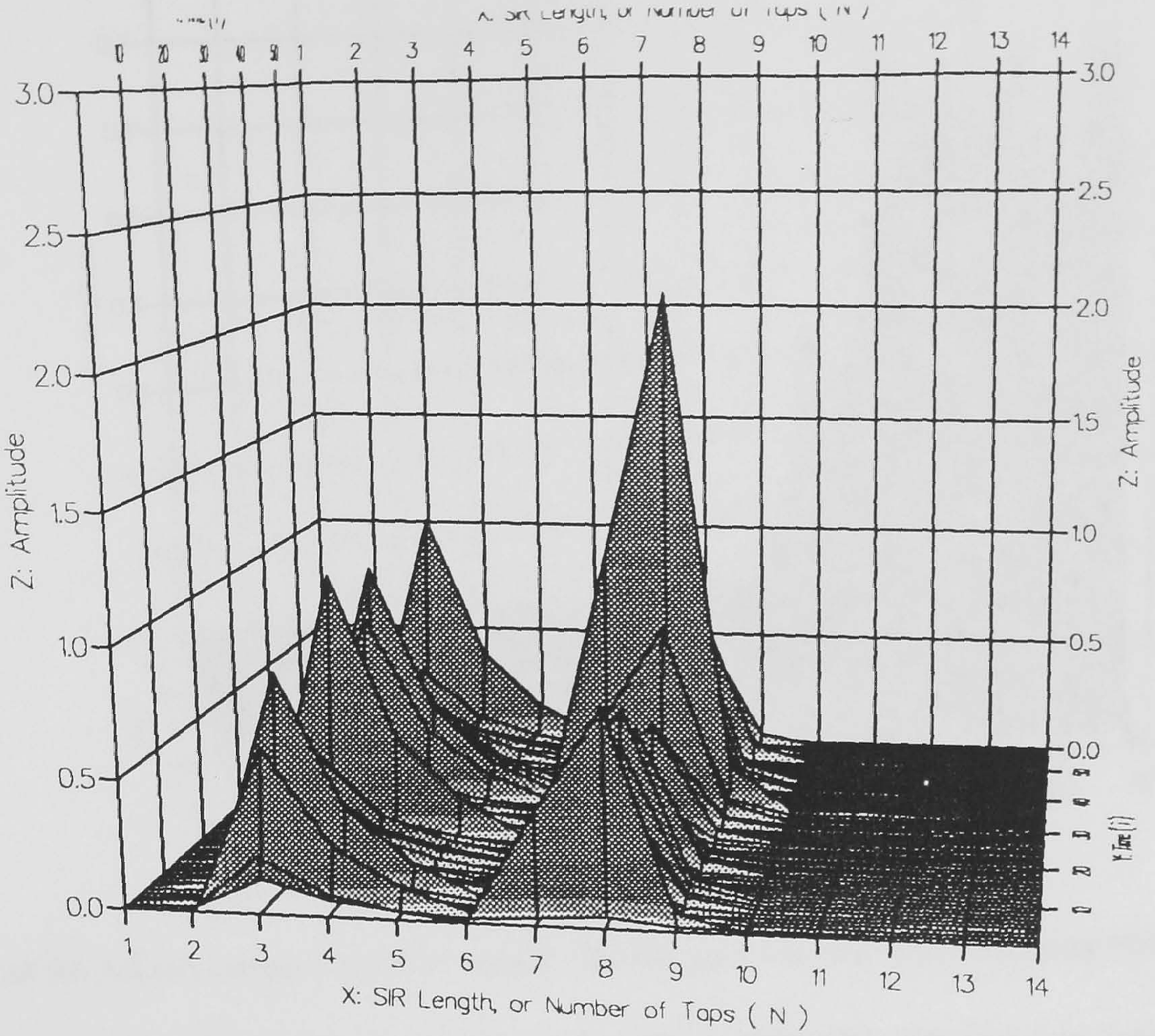
-After minimum Phase

b : A Segment of the Amplitude Variation of Channel 3 SIR after MP at : (i = 30500-30550).

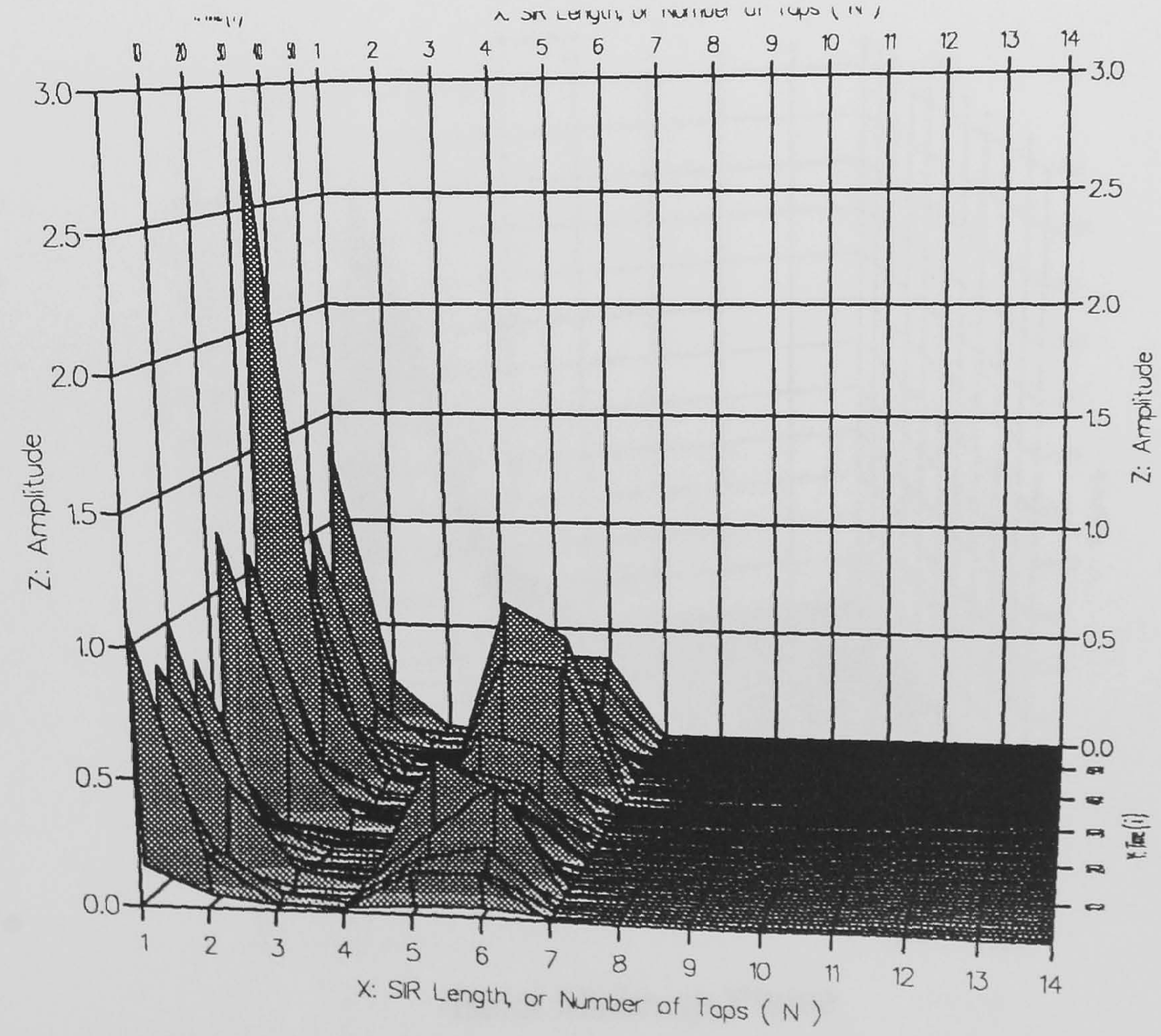


(i=30500-30550)

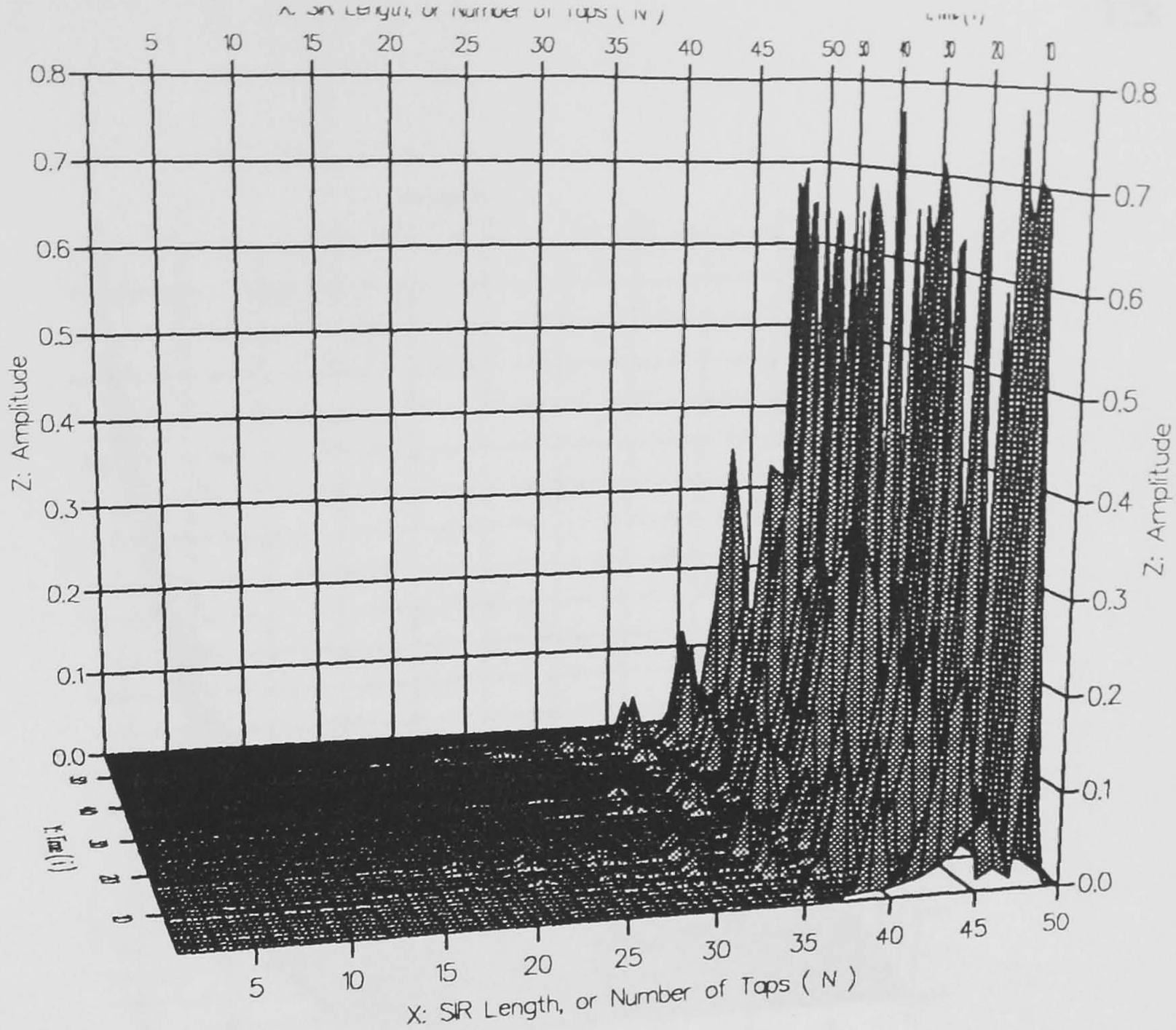
c: A Segment of the Adaptive Filter Tap-gains setting for Channel 3 at : (i = 30500-30550).



a : A Segment of the Amplitude Variation of Channel 3 SIR before MP at : (i = (1-55680)mod 960).

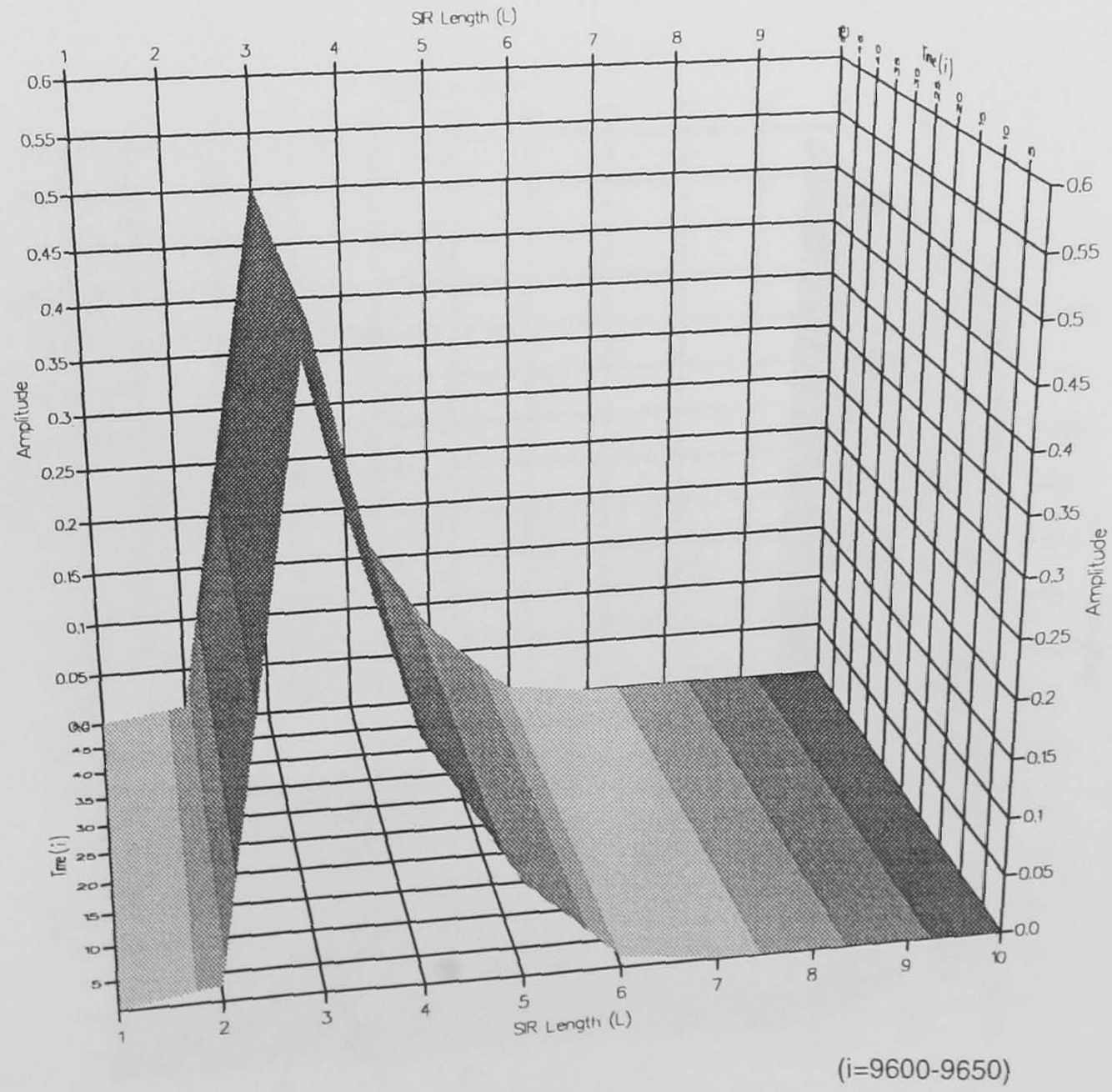


b : A Segment of the Amplitude Variation of Channel 3 SIR after MP at : (i = (1-55680)mod 960).



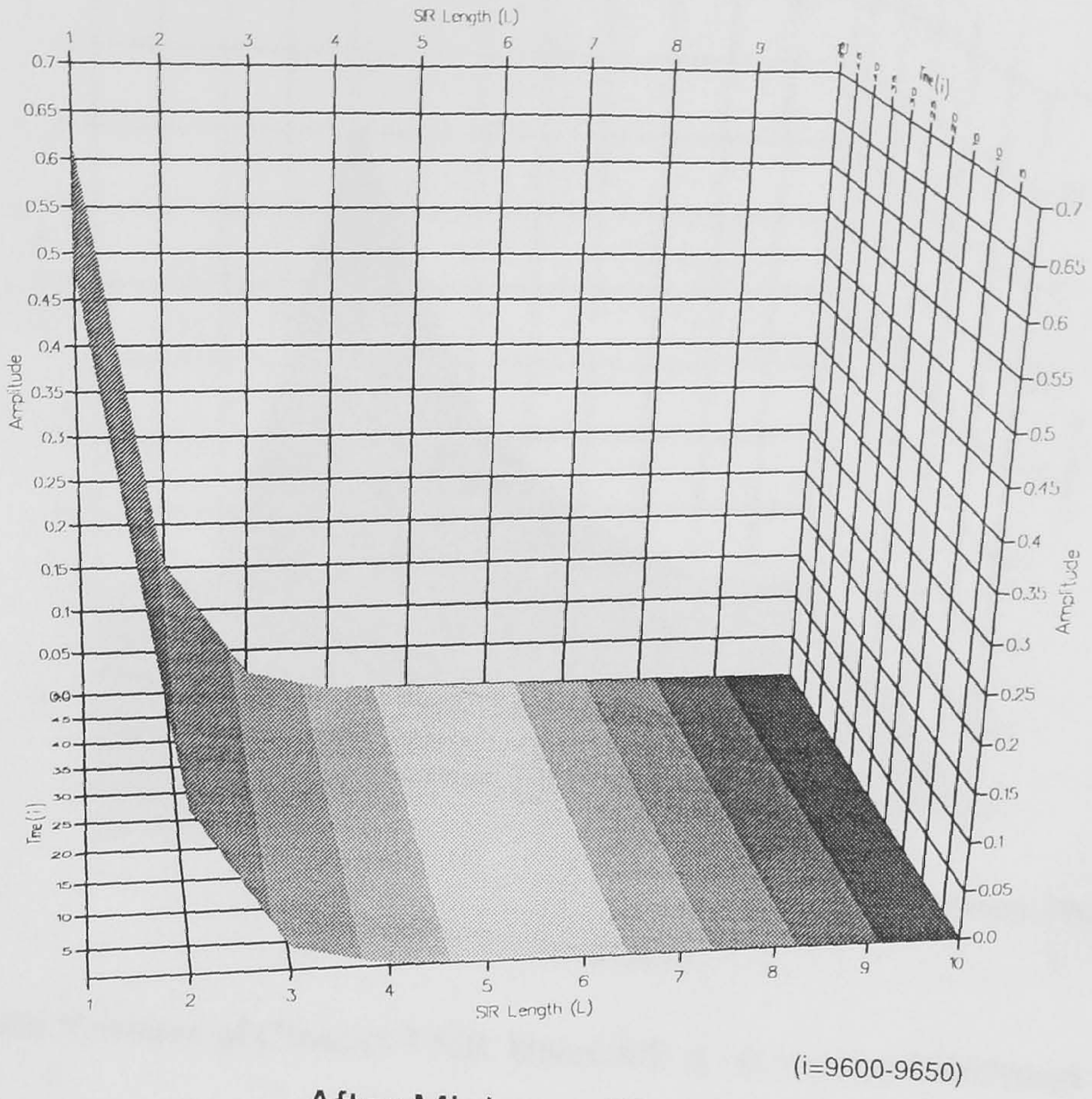
c: A Segment of the Adaptive Filter Tap-gains setting , for Channel 3 at : (i = (1-55680)mod 960).

Fig.4.5.3 : Segments of the Amplitude Variation of Non-MP, MP SIR and Tap-gains setting , for Channel 3 at different duration.



-Befor Minimum Phase

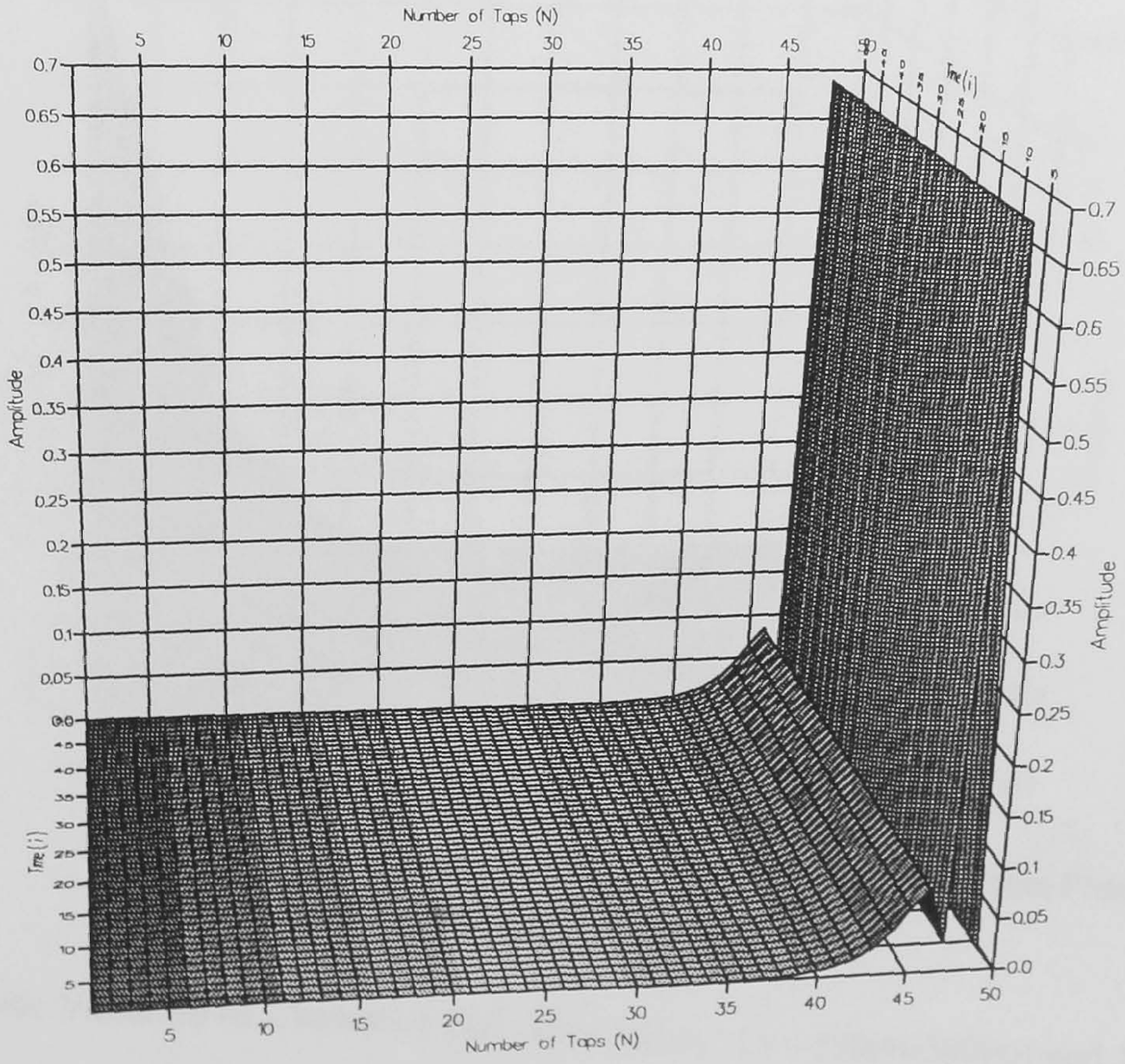
a : A Segment of the Amplitude Variation of Channel 4 SIR before MP at : (i = 9600-9650).



(i=9600-9650)

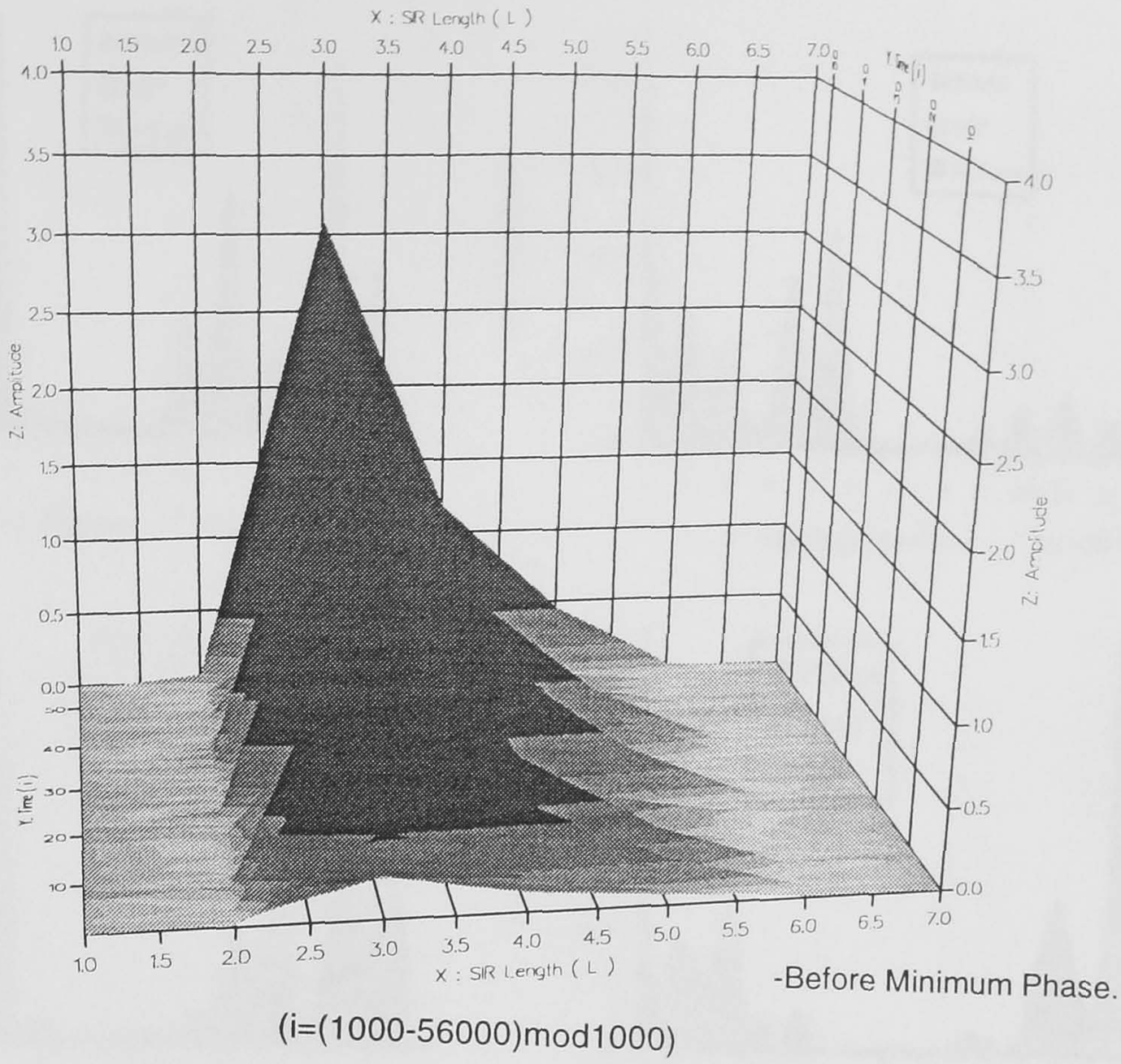
-After Minimum Phase

b : A Segment of the Amplitude Variation of Channel 4 SIR after MP at : (i = 9600-9650).

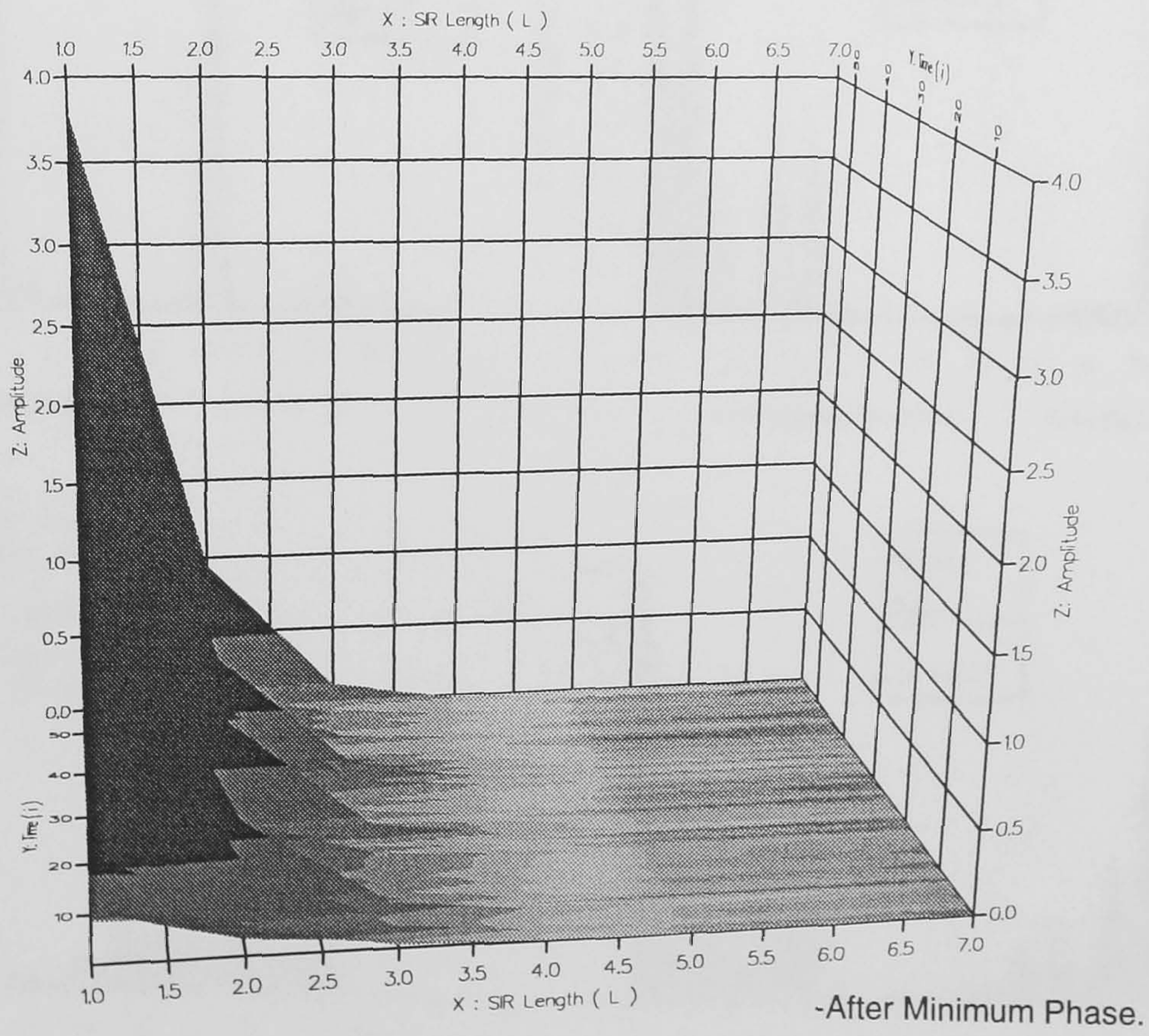


(i=9600-9650)

c: A Segment of the Adaptive Filter Tap-gains setting for Channel 4 at : (i = 9600-9650).

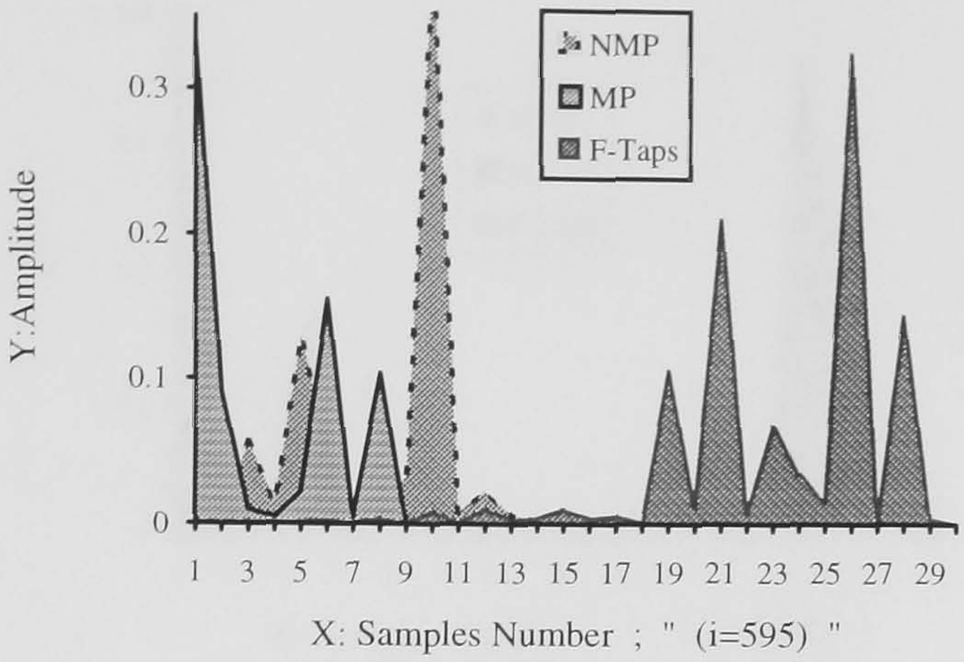


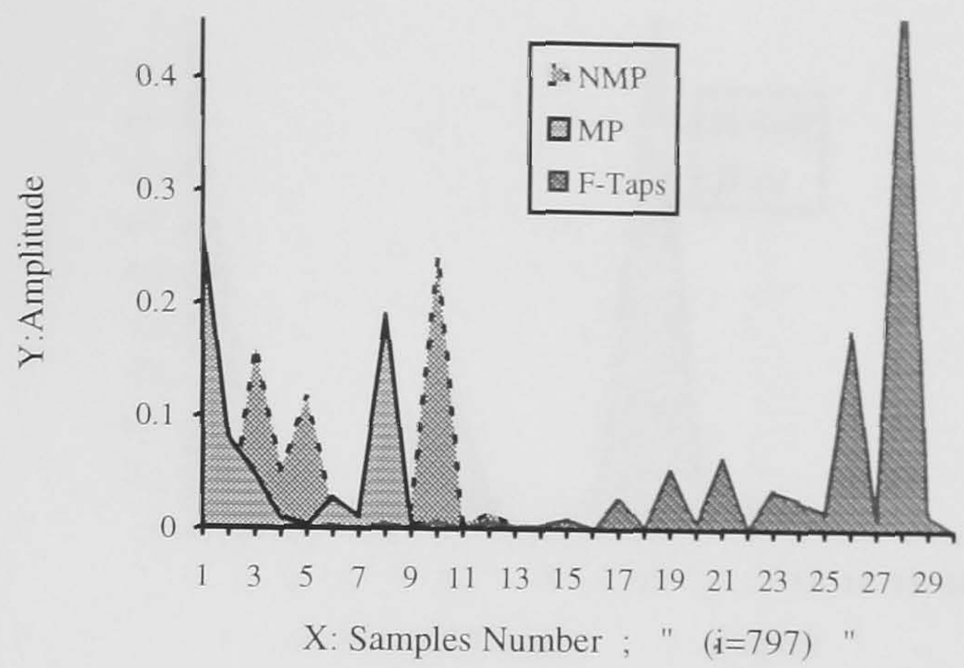
a : A Segment of the Amplitude Variation of Channel 4 SIR before MP at : (i = (1000-56000)mod 1000).

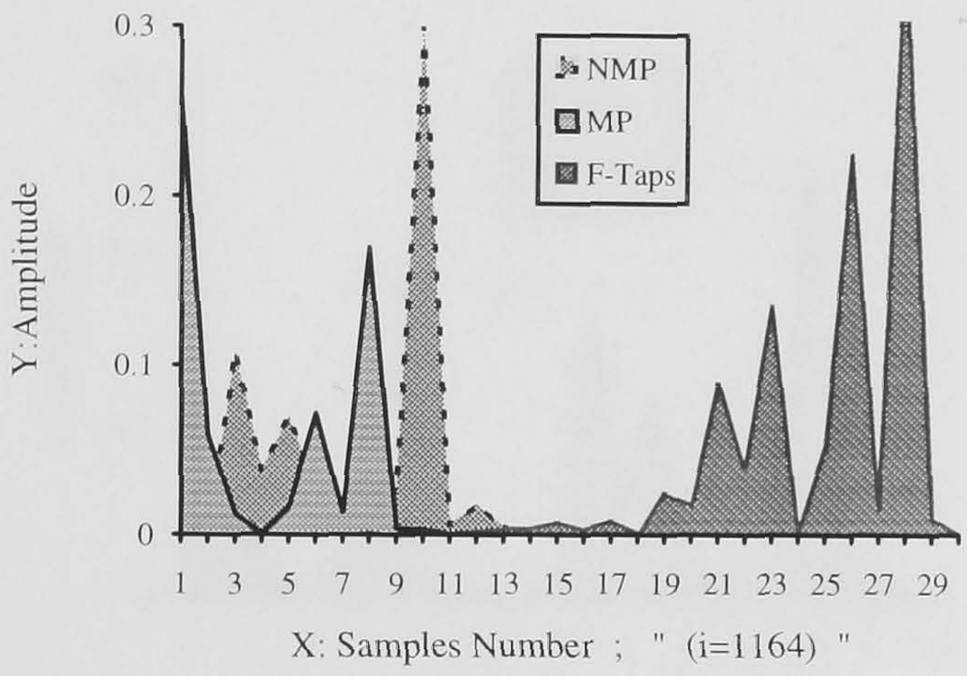


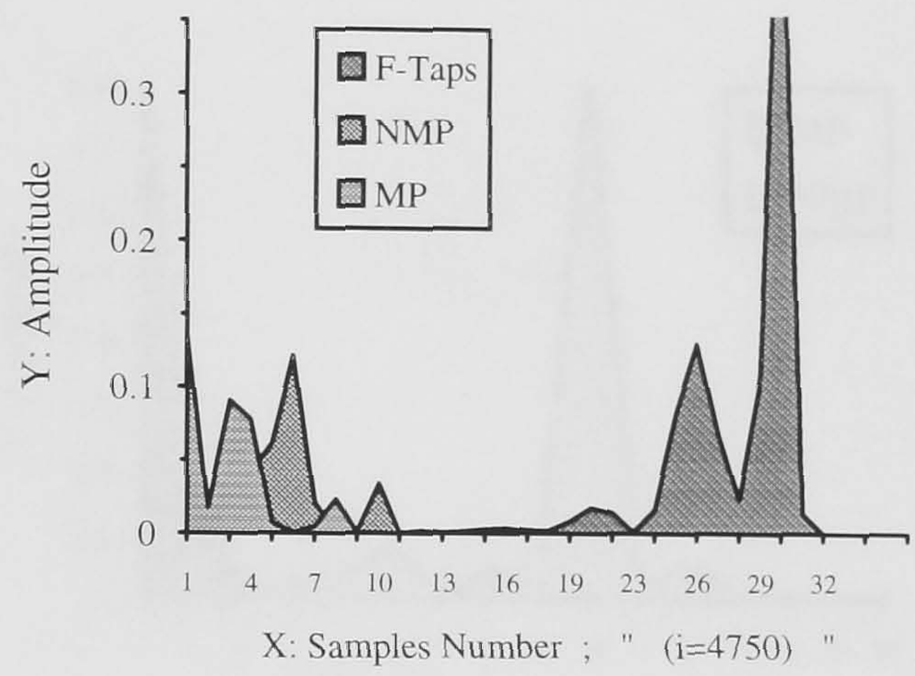
b : A Segment of the Amplitude Variation of Channel 4 SIR after MP at : (i = (1000-56000)mod 1000).

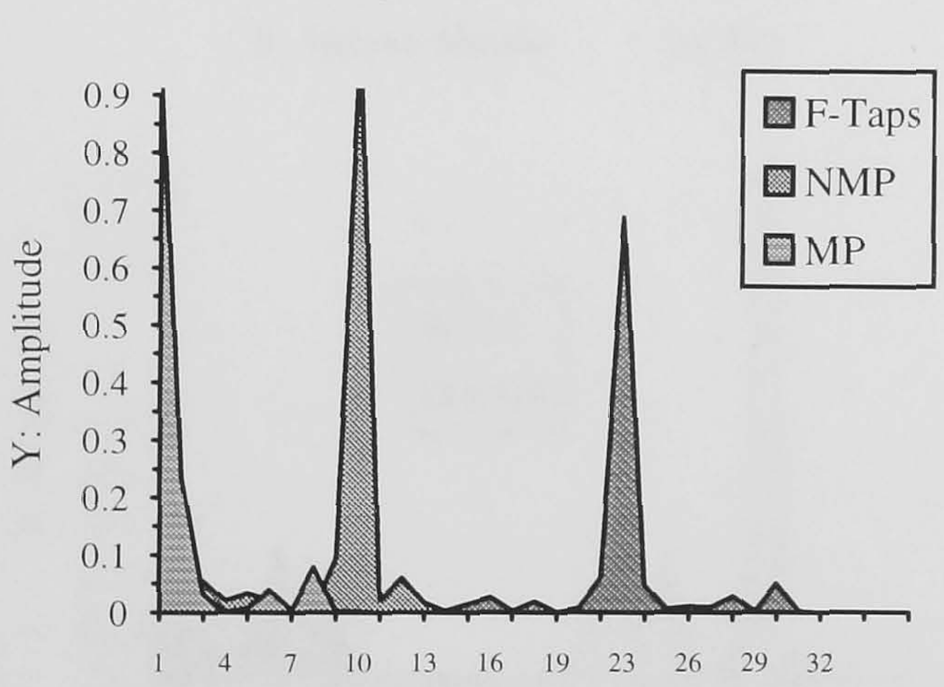
Fig.4.5.4 : Segments of the Amplitude Variation of Non-MP, MP SIR for Channel 4 at different duration

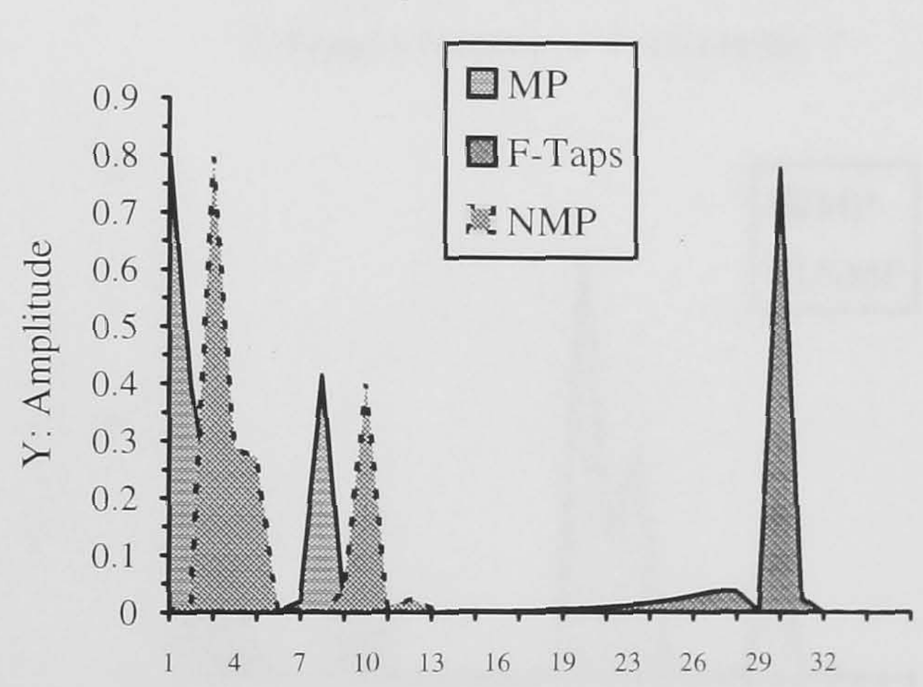

NMP
MP
F-Taps
Y:Amplitude
0.3
0.2
0.1
0
1 3 5 7 9 11 13 15 17 19 21 23 25 27 29
X: Samples Number ; " (i=595) "

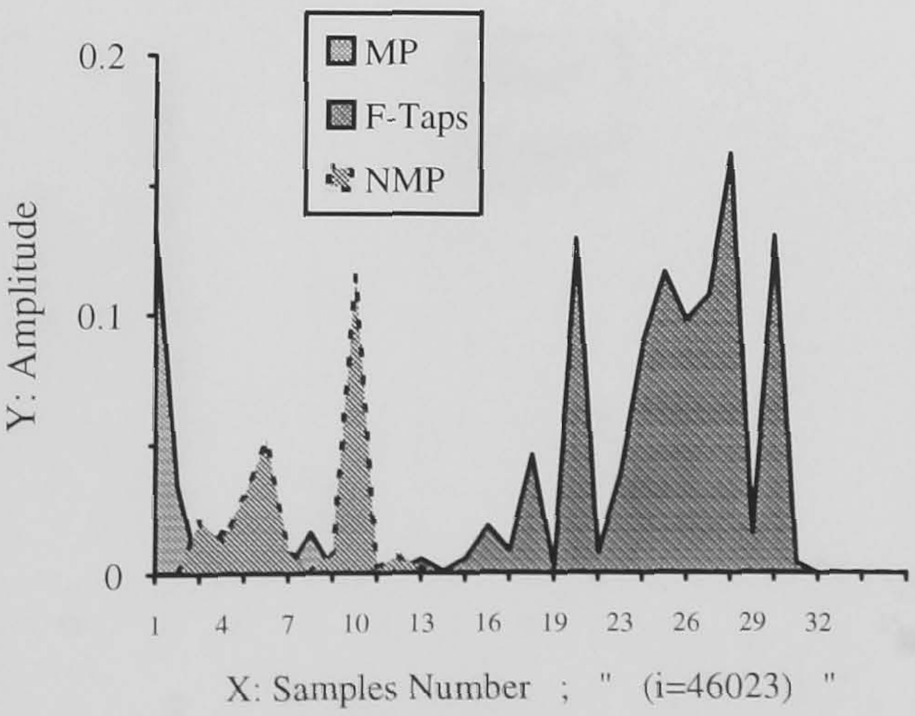

NMP
MP
F-Taps
Y:Amplitude
0.4
0.3
0.2
0.1
0
1 3 5 7 9 11 13 15 17 19 21 23 25 27 29
X: Samples Number ; " (i=797) "

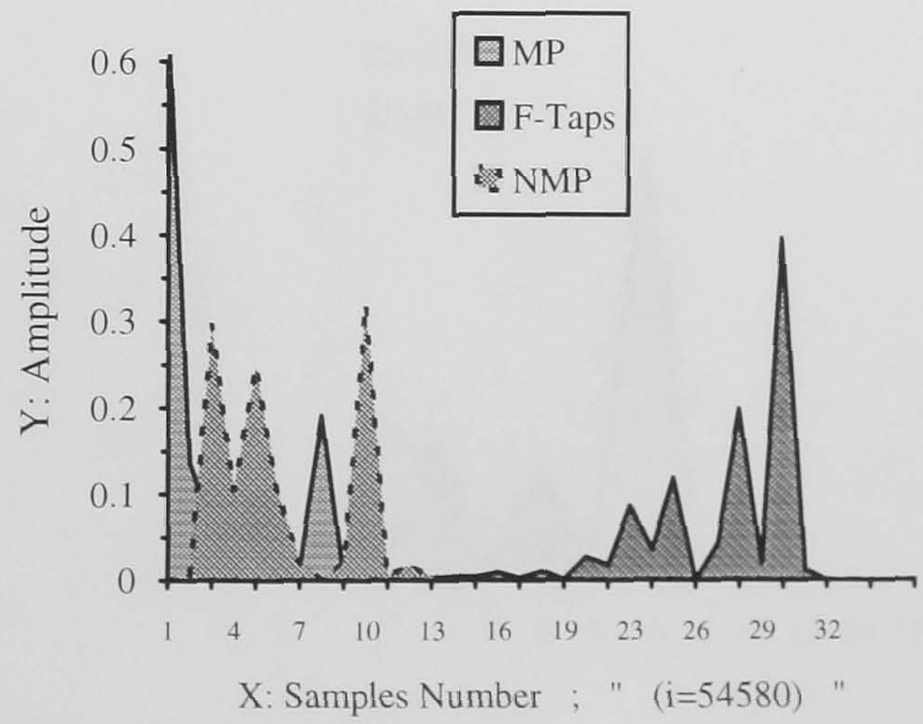

NMP
MP
F-Taps
Y:Amplitude
0.3
0.2
0.1
0
1 3 5 7 9 11 13 15 17 19 21 23 25 27 29
X: Samples Number ; " (i=1164) "

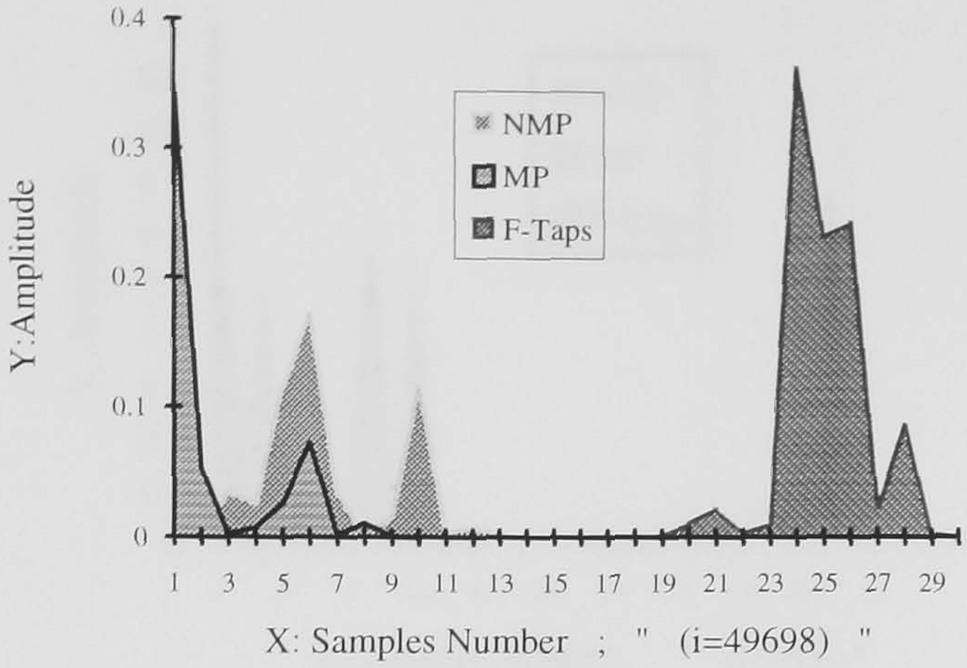

F-Taps
NMP
MP
Y: Amplitude
0.3
0.2
0.1
0
1 4 7 10 13 16 19 23 26 29 32
X: Samples Number ; " (i=4750) "

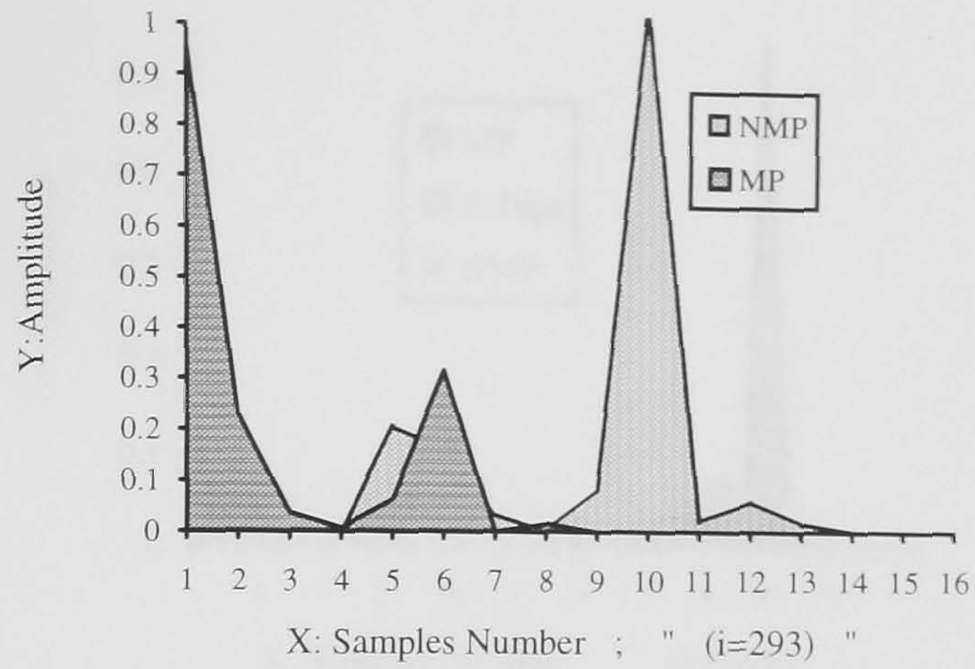

F-Taps
NMP
MP
Y: Amplitude
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
1 4 7 10 13 16 19 23 26 29 32
X: Samples Number ; " (i=11475) "

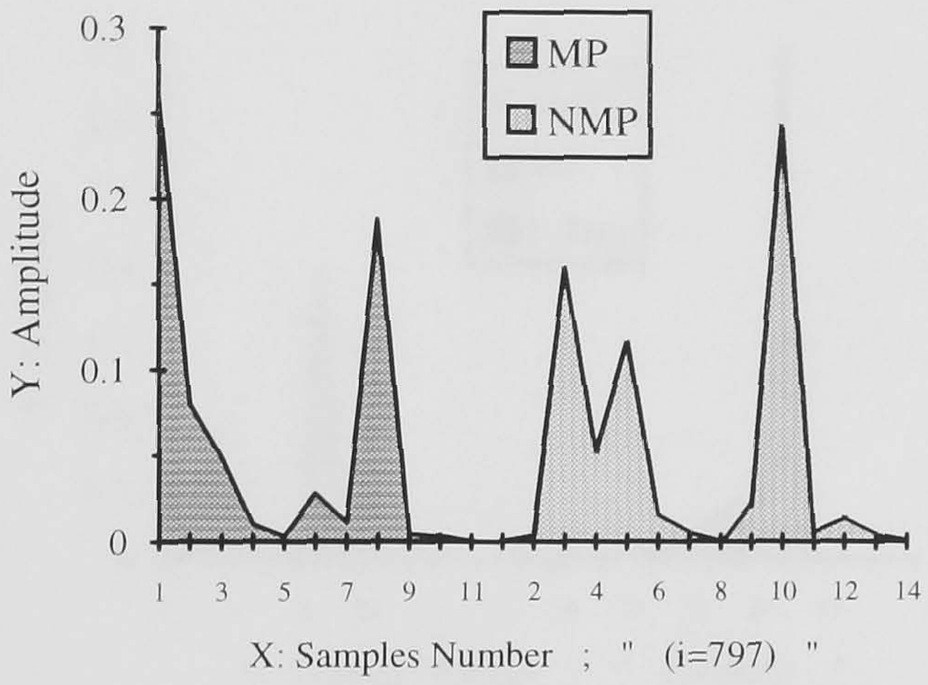

MP
F-Taps
NMP
Y: Amplitude
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
1 4 7 10 13 16 19 23 26 29 32
X: Samples Number ; " (i=19221) "

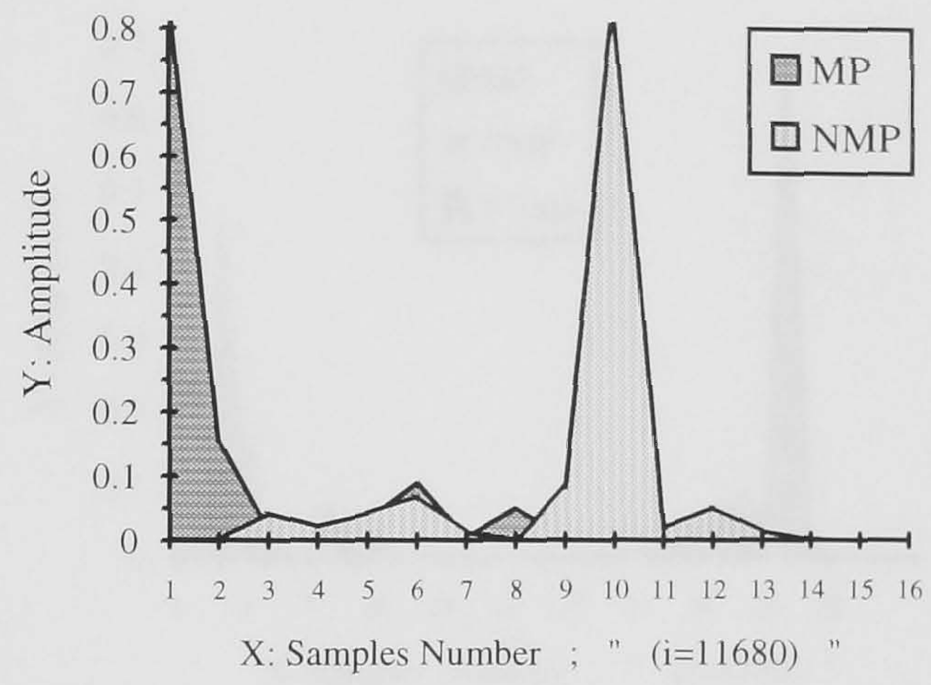

MP
F-Taps
NMP
Y: Amplitude
0.2
0.1
0
1 4 7 10 13 16 19 23 26 29 32
X: Samples Number ; " (i=46023) "

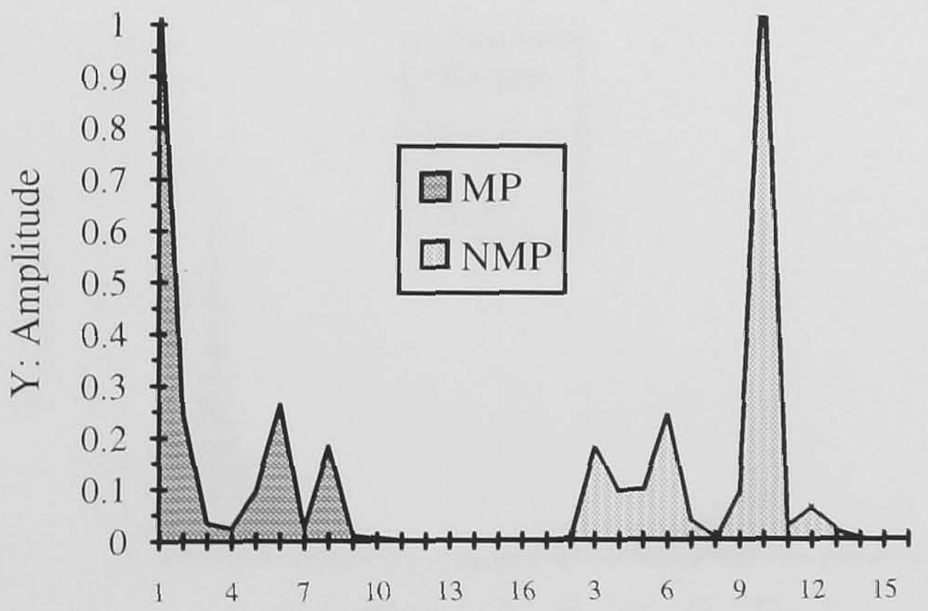

MP
F-Taps
NMP
Y: Amplitude
0.6
0.5
0.4
0.3
0.2
0.1
0
1 4 7 10 13 16 19 23 26 29 32
X: Samples Number ; " (i=54580) "

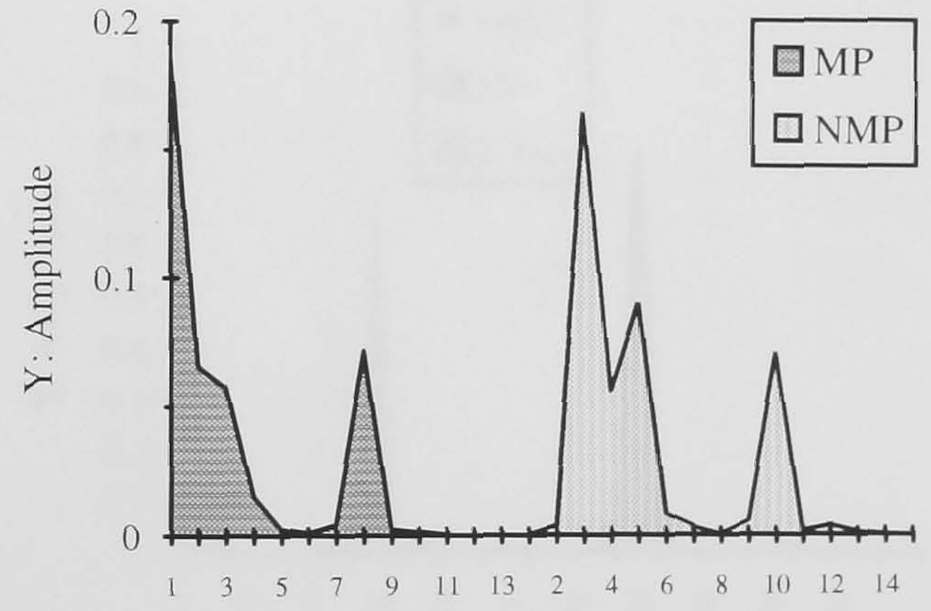


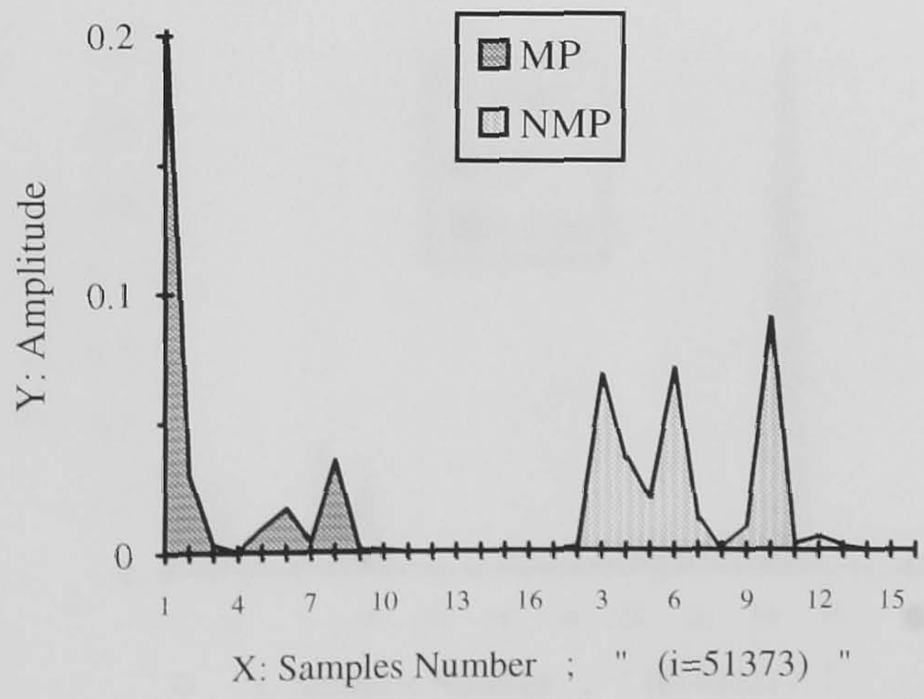












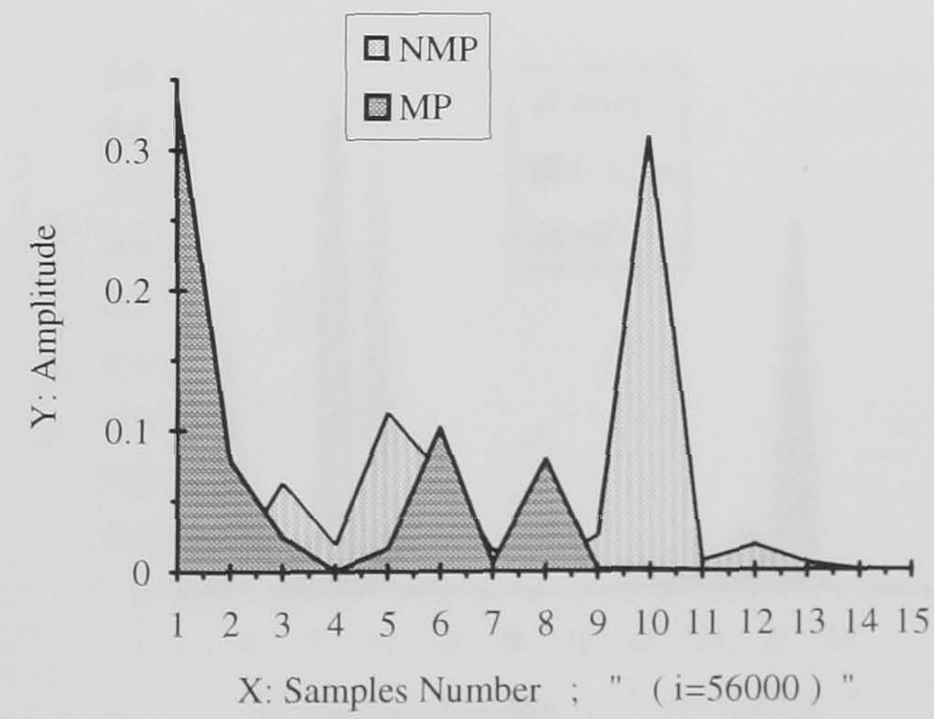
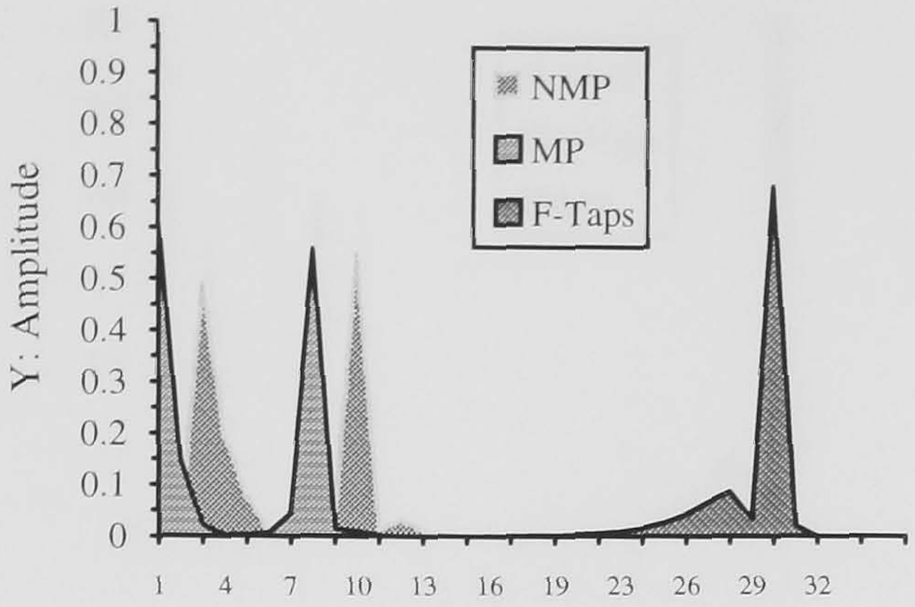
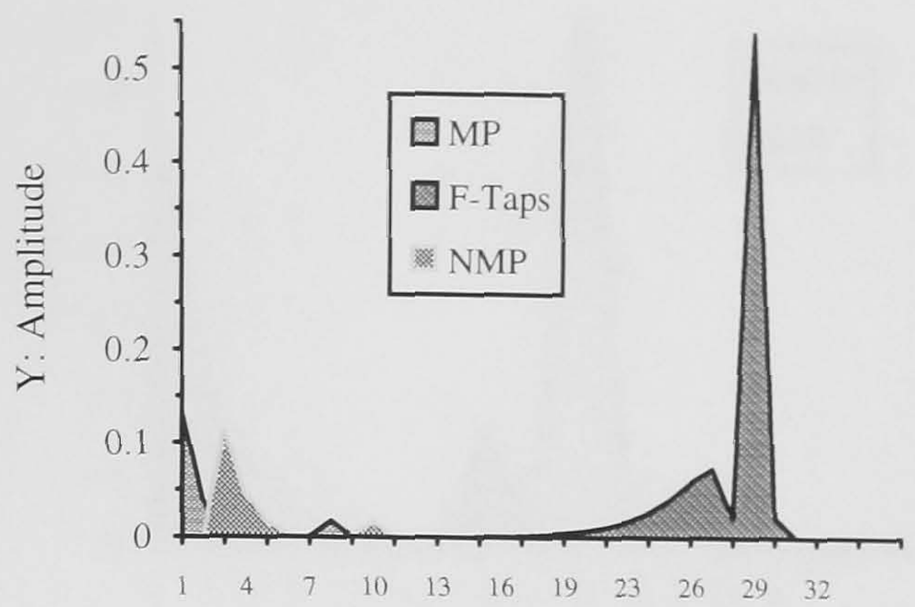
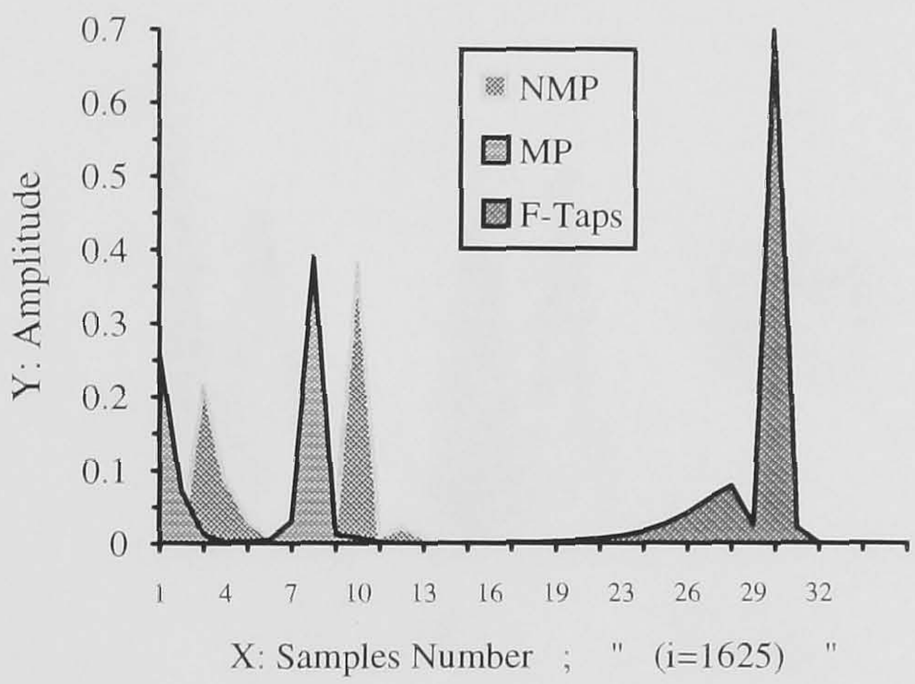
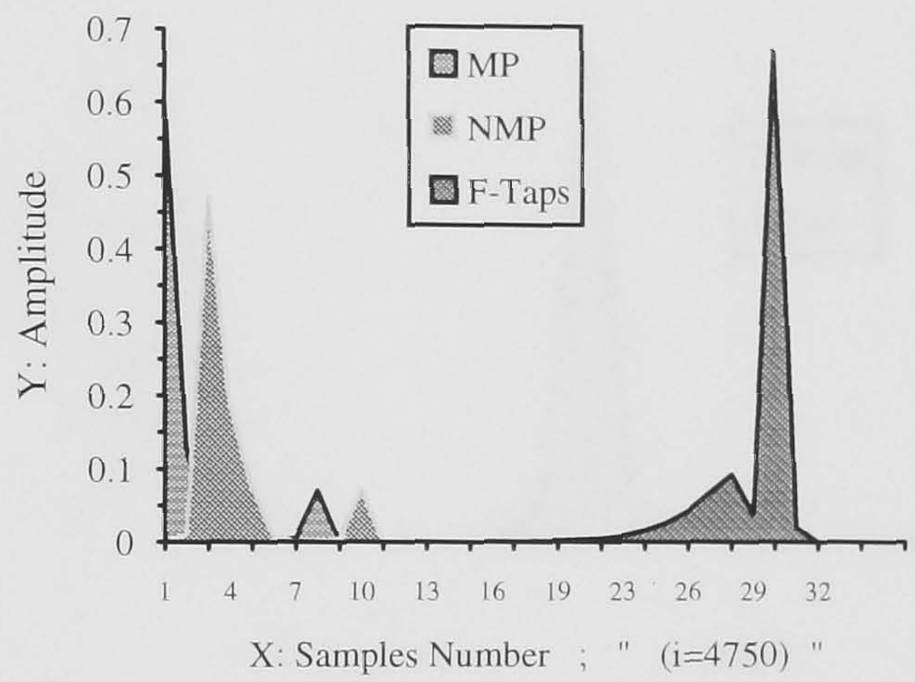


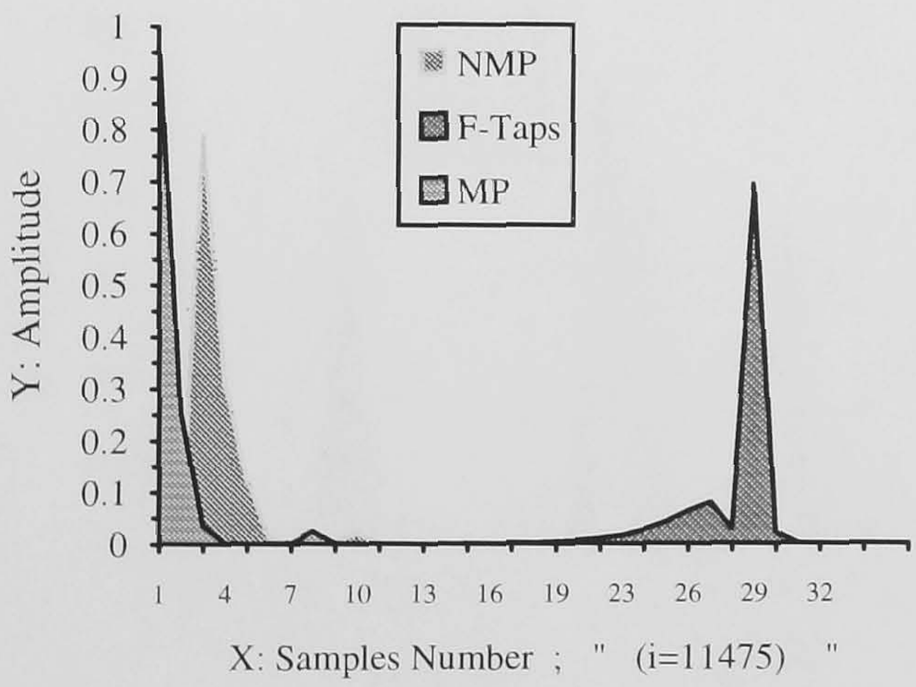
Fig.4.5.5 Non-MP, MP SIR and Filter Coefficients for Channel1.

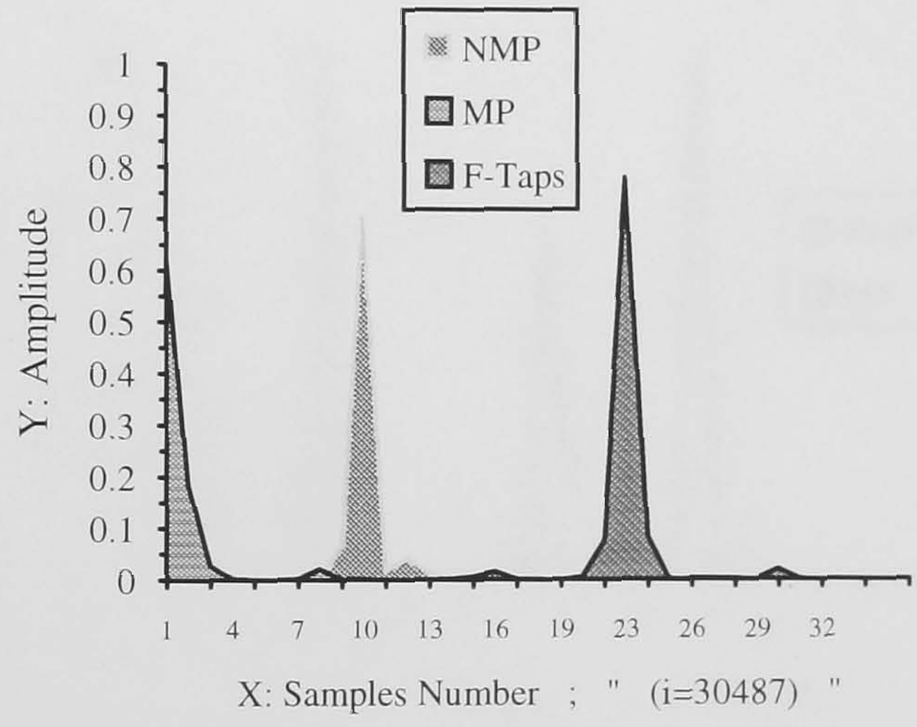

Y: Amplitude
NMP
MP
F-Taps
X: Samples Number ; " (i=595) "

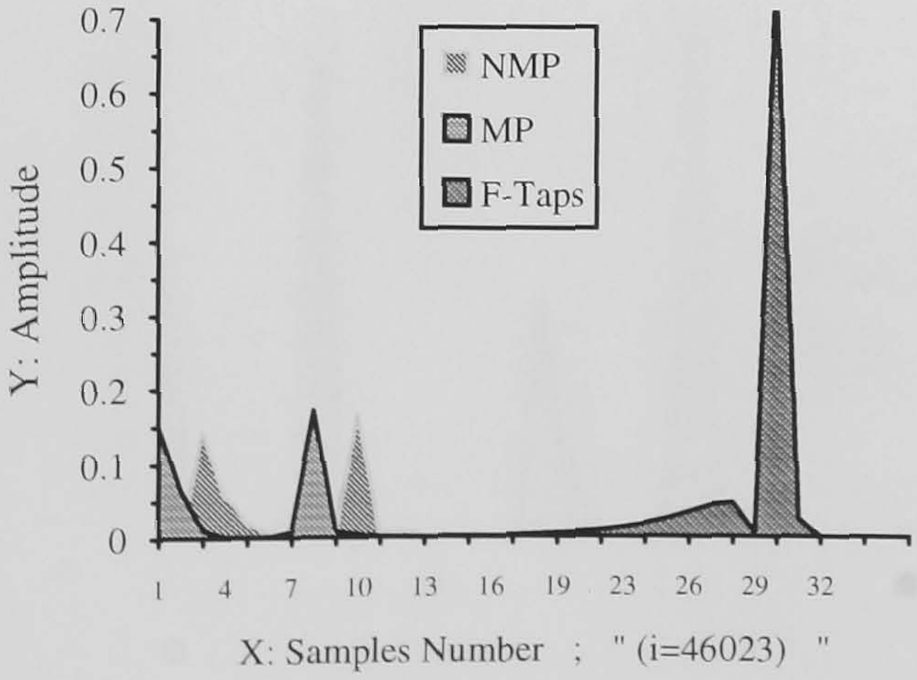

Y: Amplitude
MP
F-Taps
NMP
X: Samples Number ; " (i=1164) "

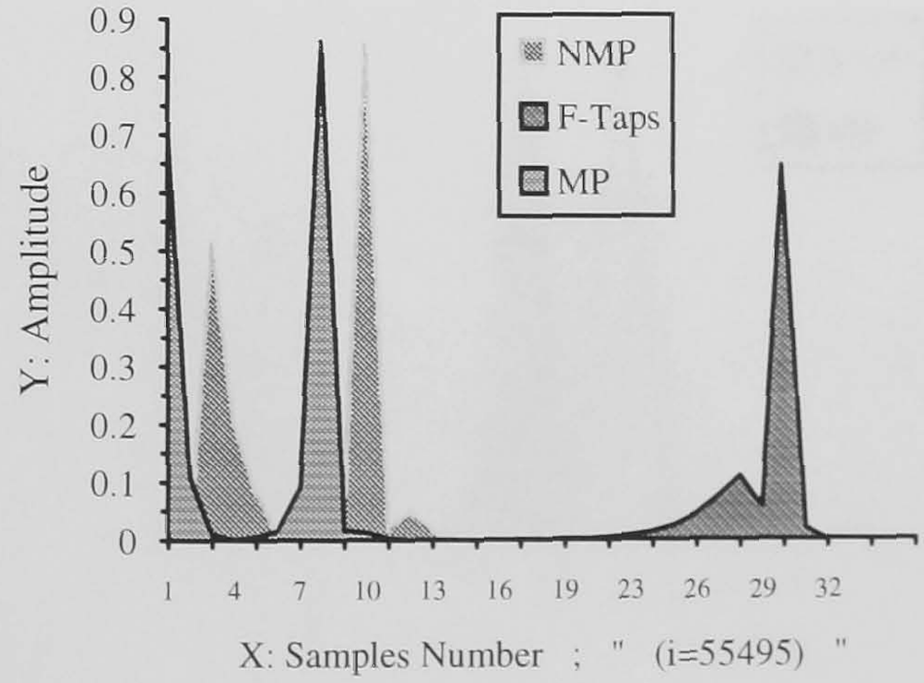

Y: Amplitude
NMP
MP
F-Taps
X: Samples Number ; " (i=1625) "

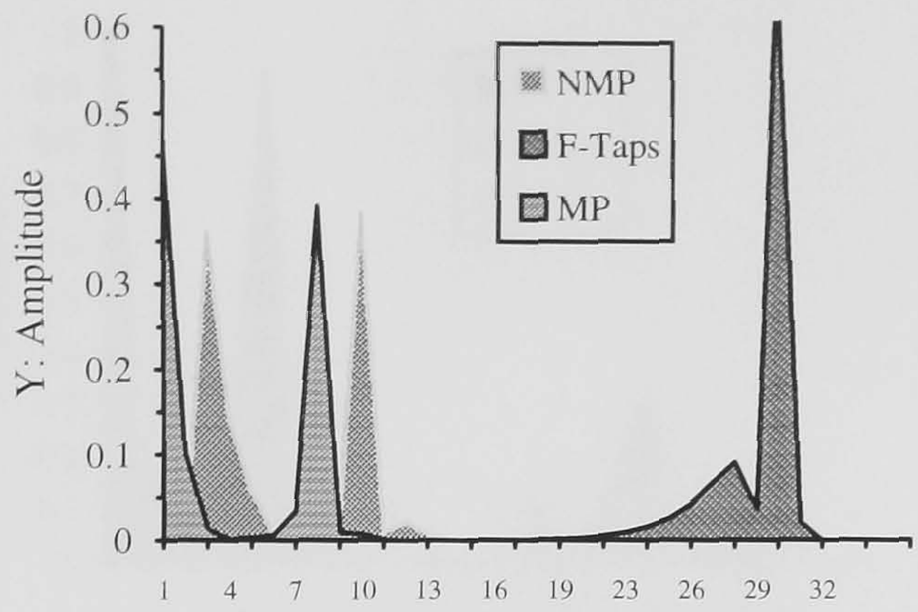

Y: Amplitude
MP
NMP
F-Taps
X: Samples Number ; " (i=4750) "

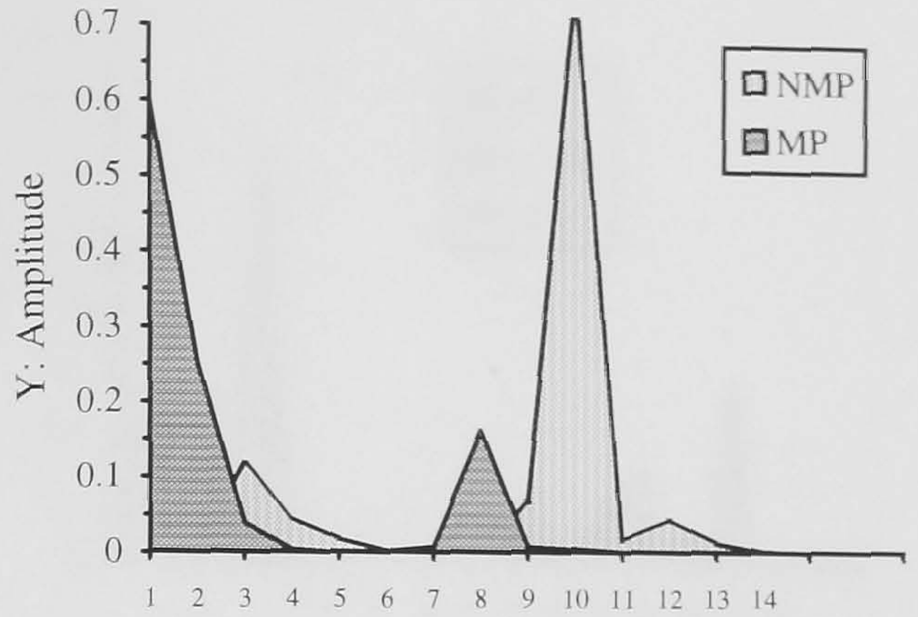

Y: Amplitude
NMP
F-Taps
MP
X: Samples Number ; " (i=11475) "

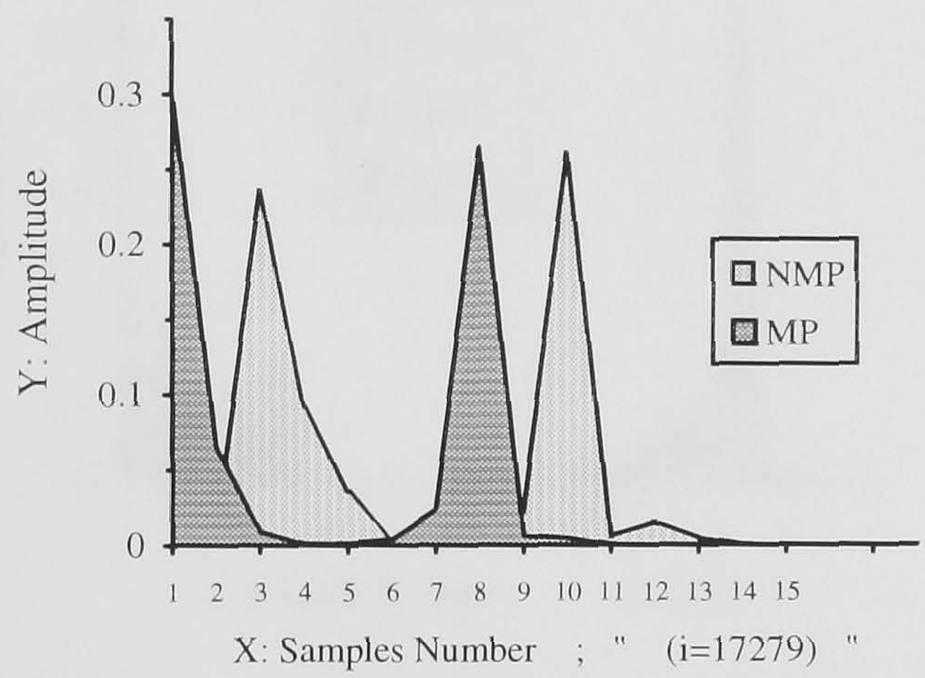

Y: Amplitude
NMP
MP
F-Taps
X: Samples Number ; " (i=30487) "

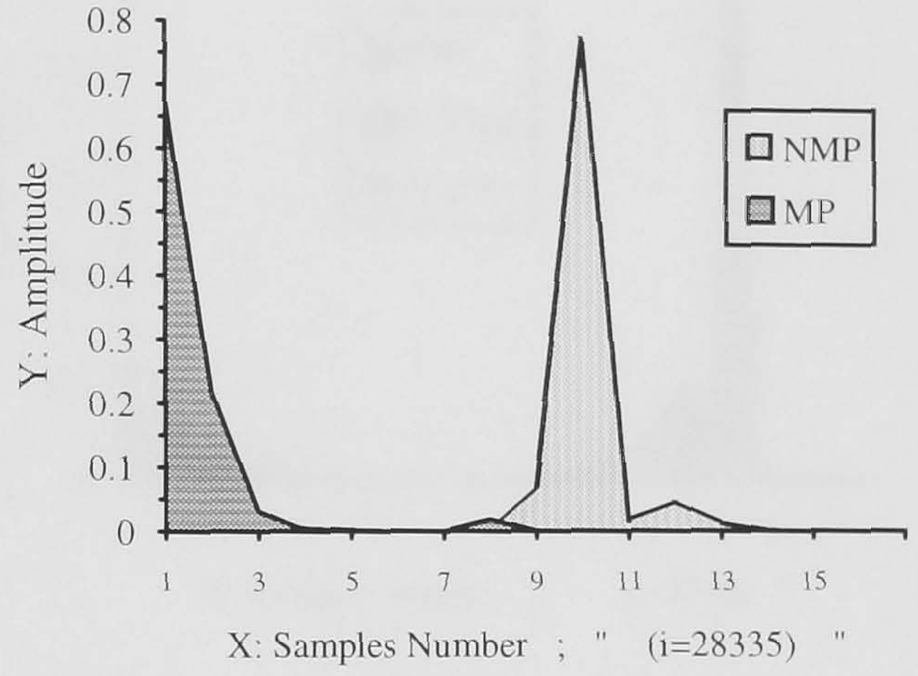

Y: Amplitude
NMP
MP
F-Taps
X: Samples Number ; " (i=46023) "

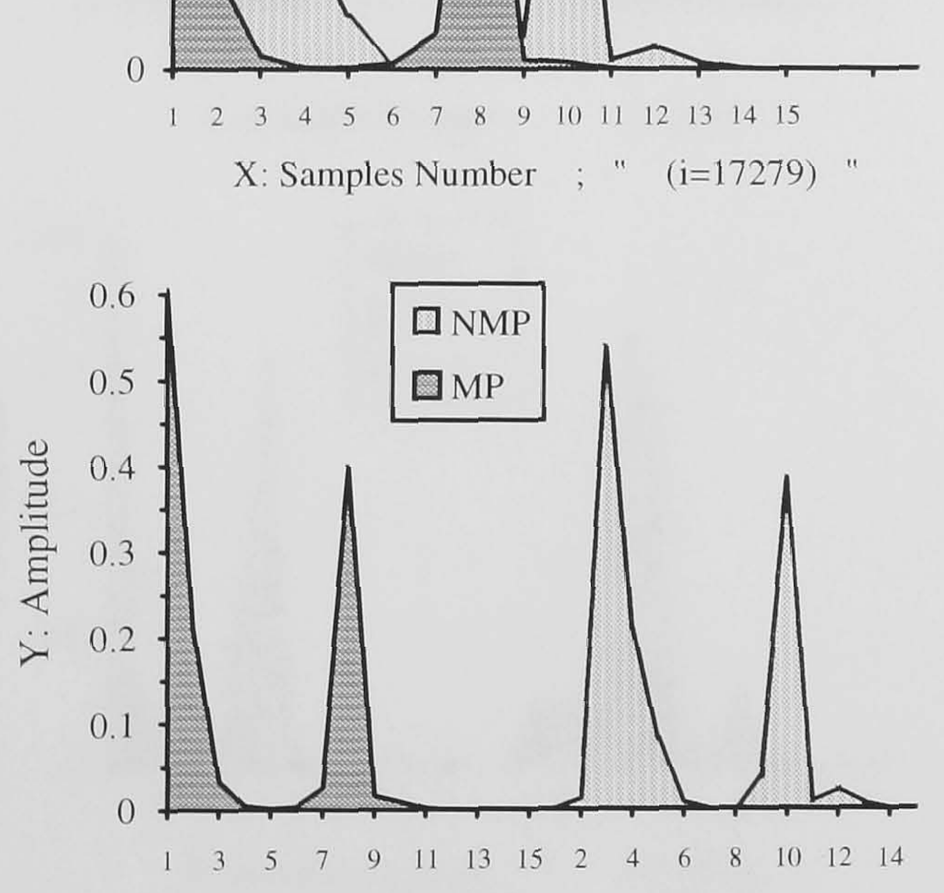

Y: Amplitude
NMP
F-Taps
MP
X: Samples Number ; " (i=55495) "











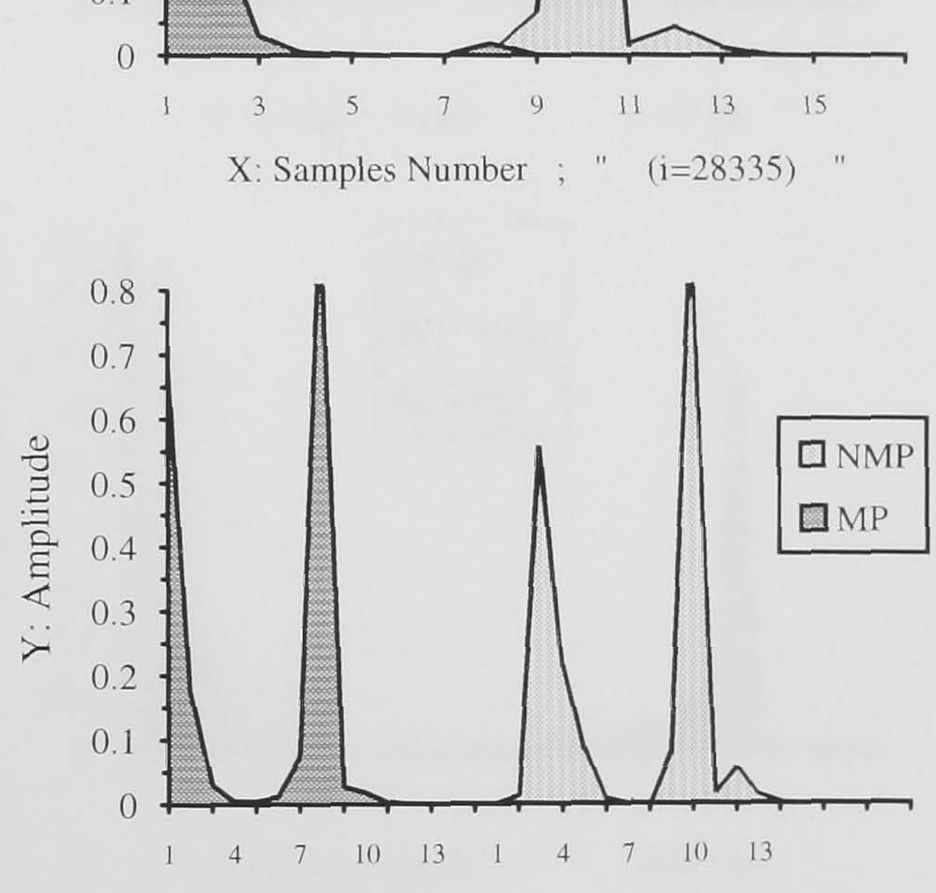
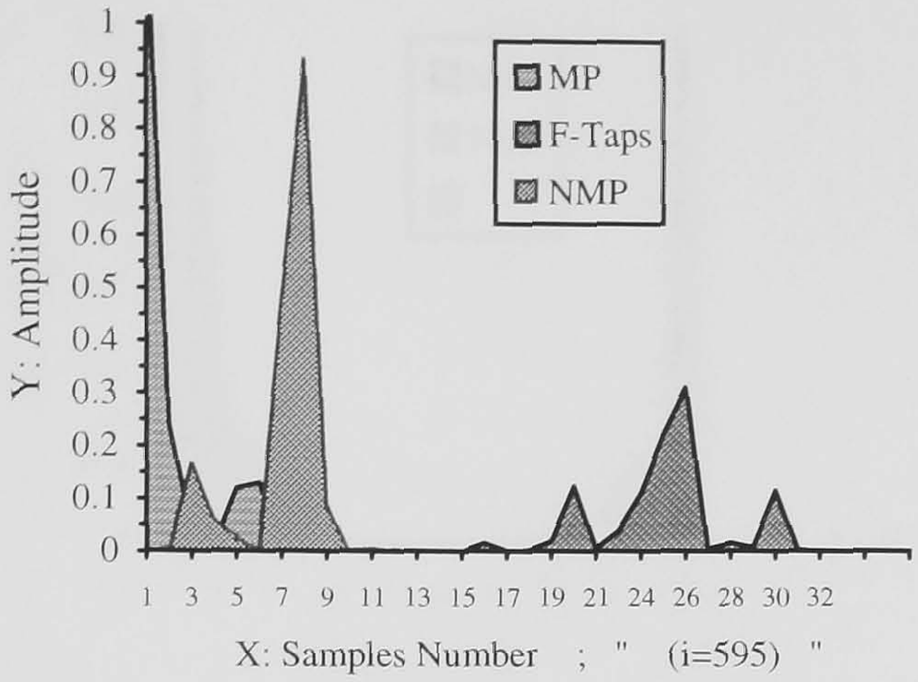
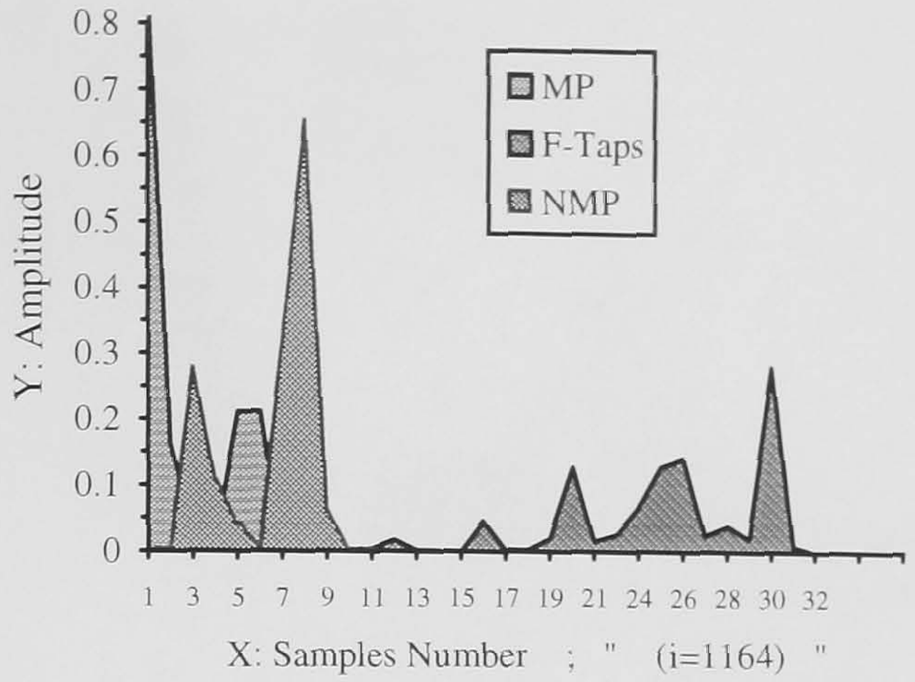
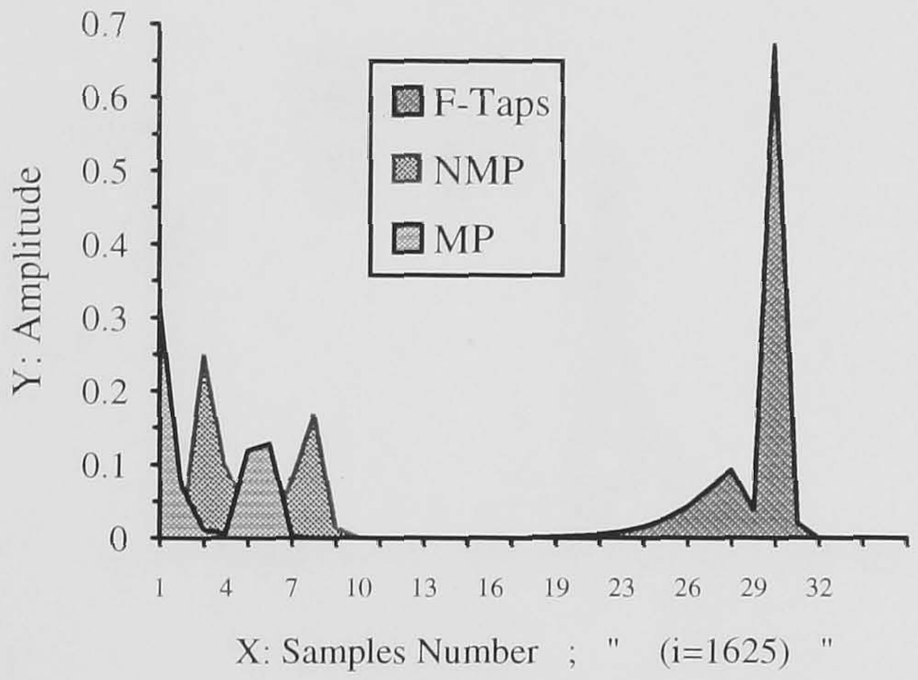
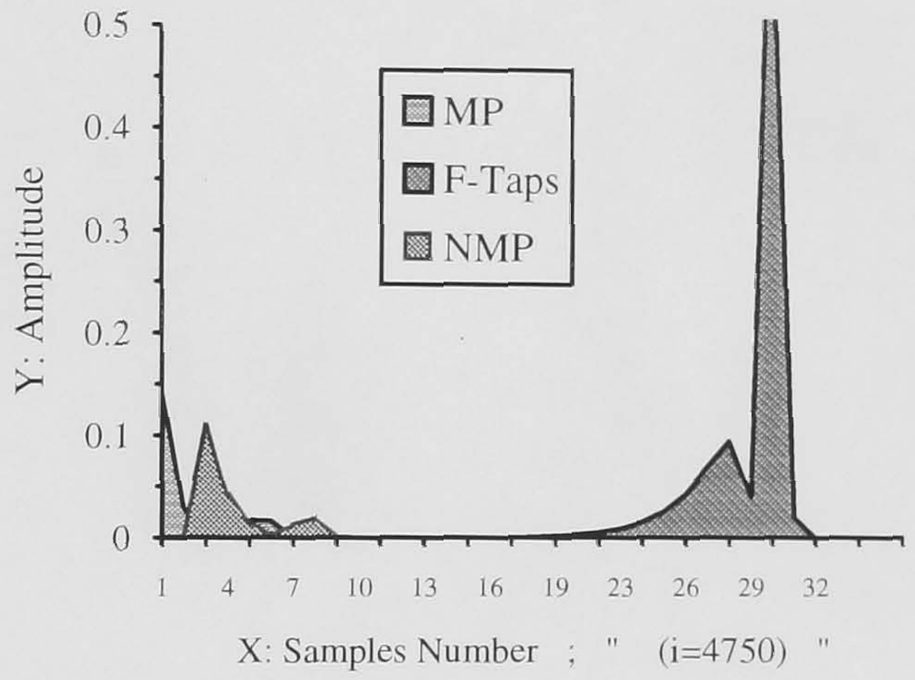


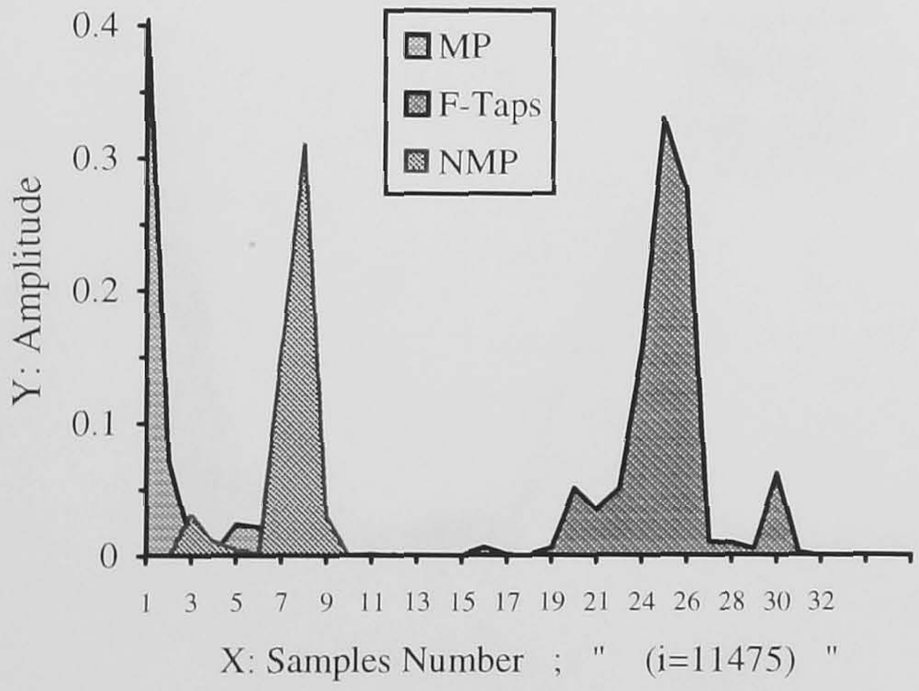
Fig. 4.5.6 Non-MP, MP SIR and Filter Coefficients for Channel2.

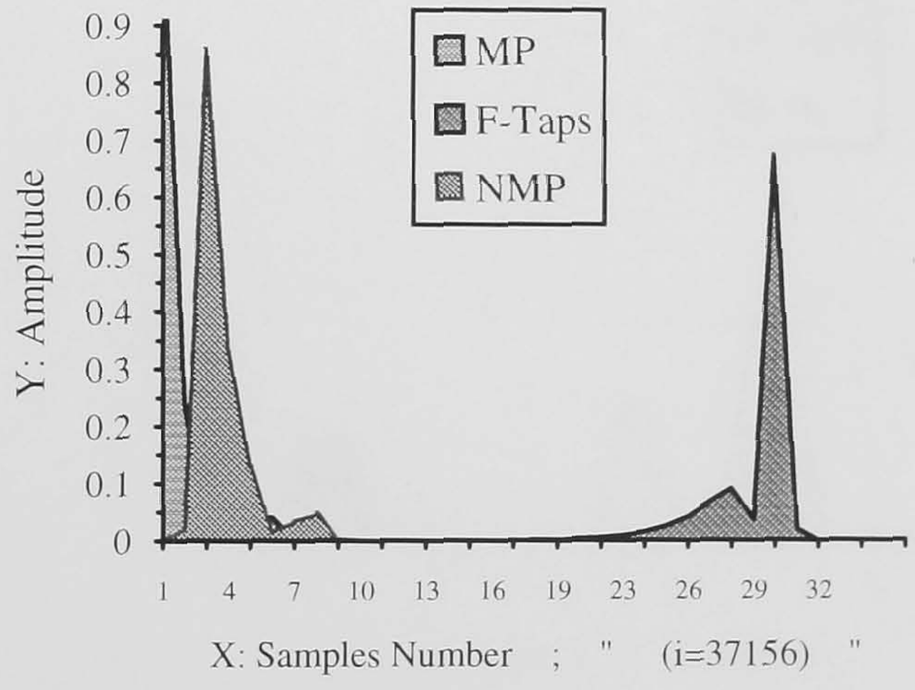

Y: Amplitude
MP
F-Taps
NMP
X: Samples Number ; " (i=595) "

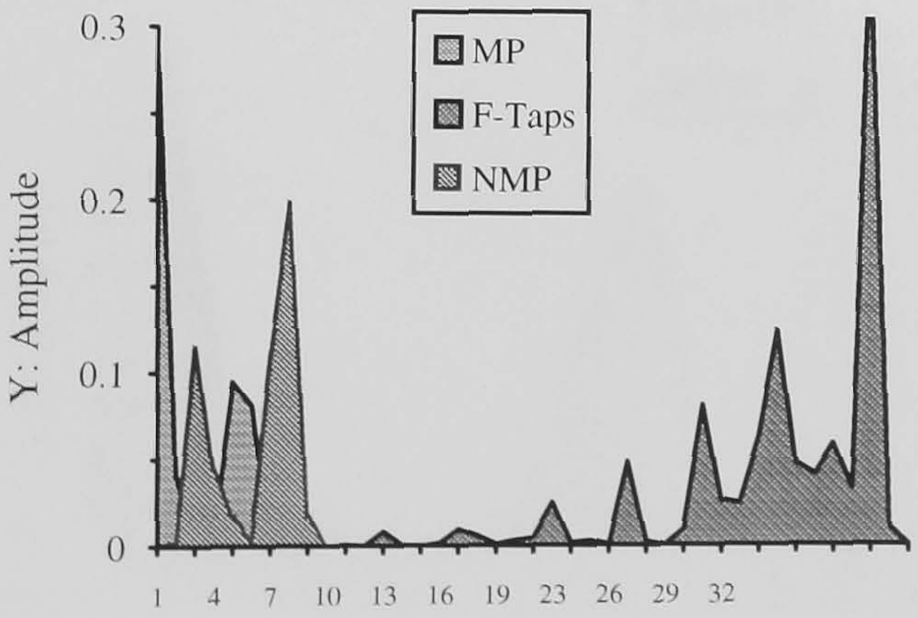

Y: Amplitude
MP
F-Taps
NMP
X: Samples Number ; " (i=1164) "

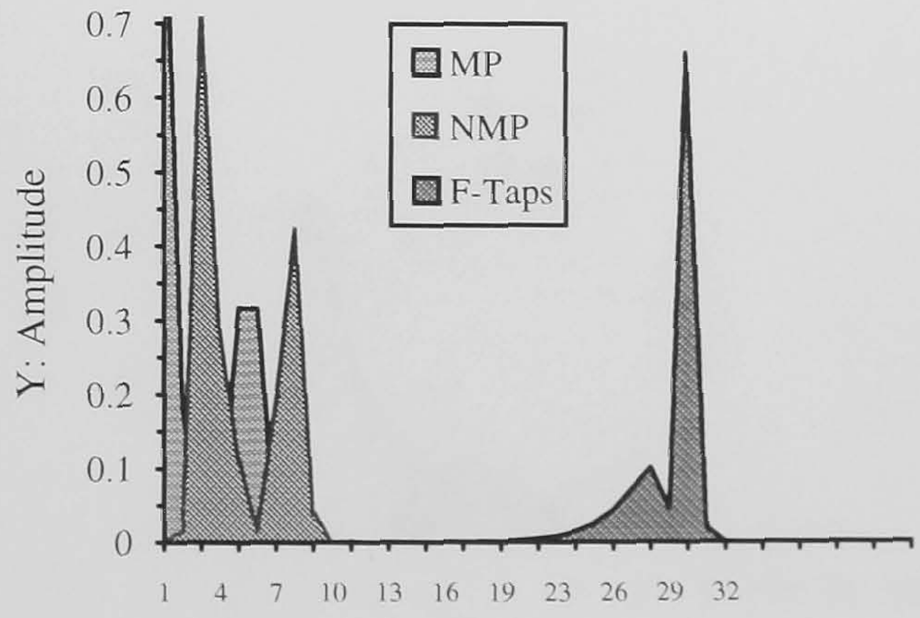

Y: Amplitude
F-Taps
NMP
MP
X: Samples Number ; " (i=1625) "

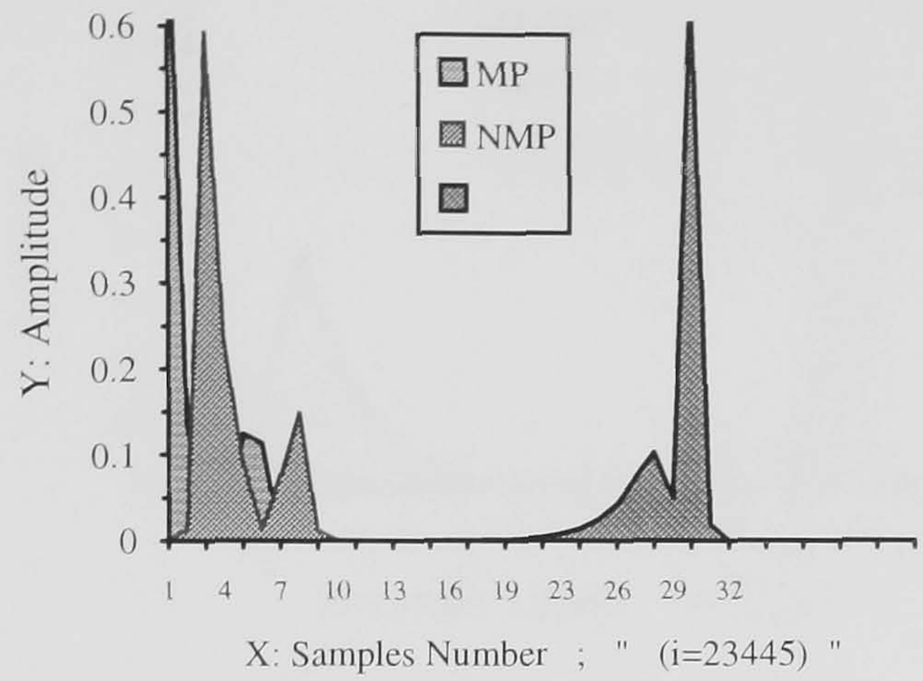

Y: Amplitude
MP
F-Taps
NMP
X: Samples Number ; " (i=4750) "

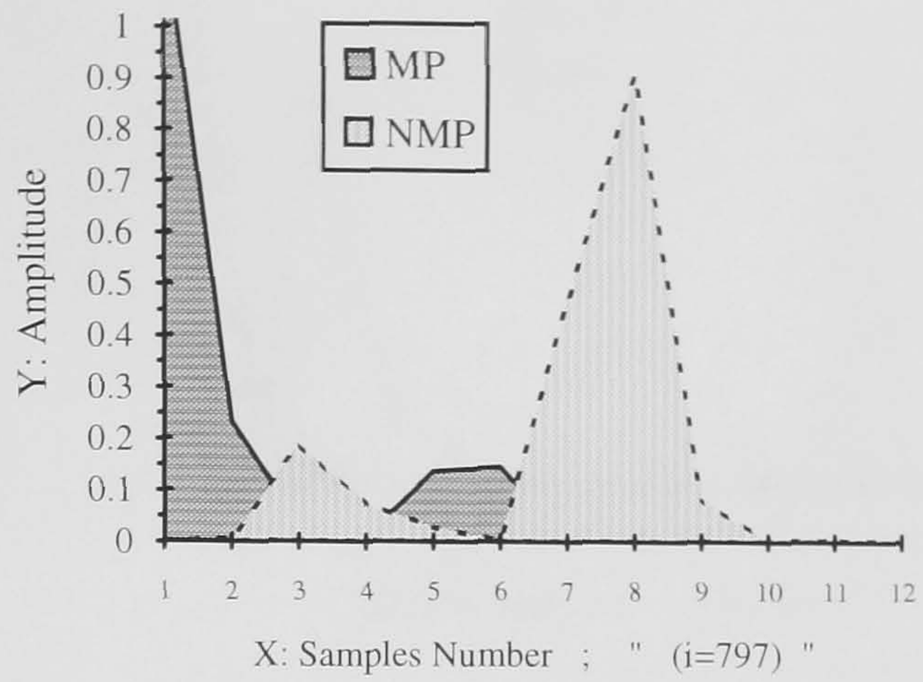

Y: Amplitude
MP
F-Taps
NMP
X: Samples Number ; " (i=11475) "

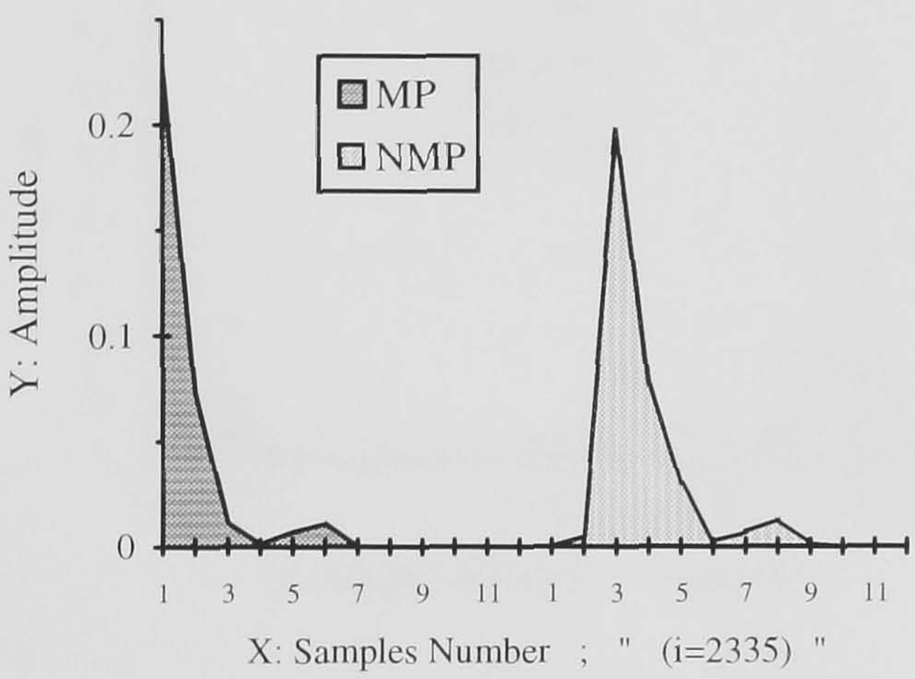

Y: Amplitude
MP
F-Taps
NMP
X: Samples Number ; " (i=37156) "

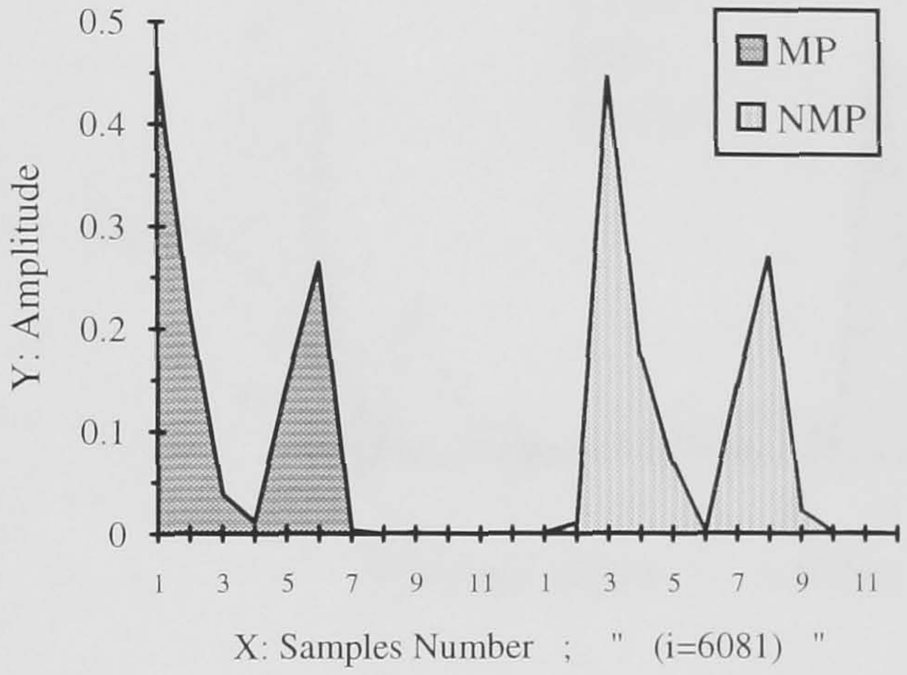

Y: Amplitude
MP
F-Taps
NMP
X: Samples Number ; " (i=46023) "

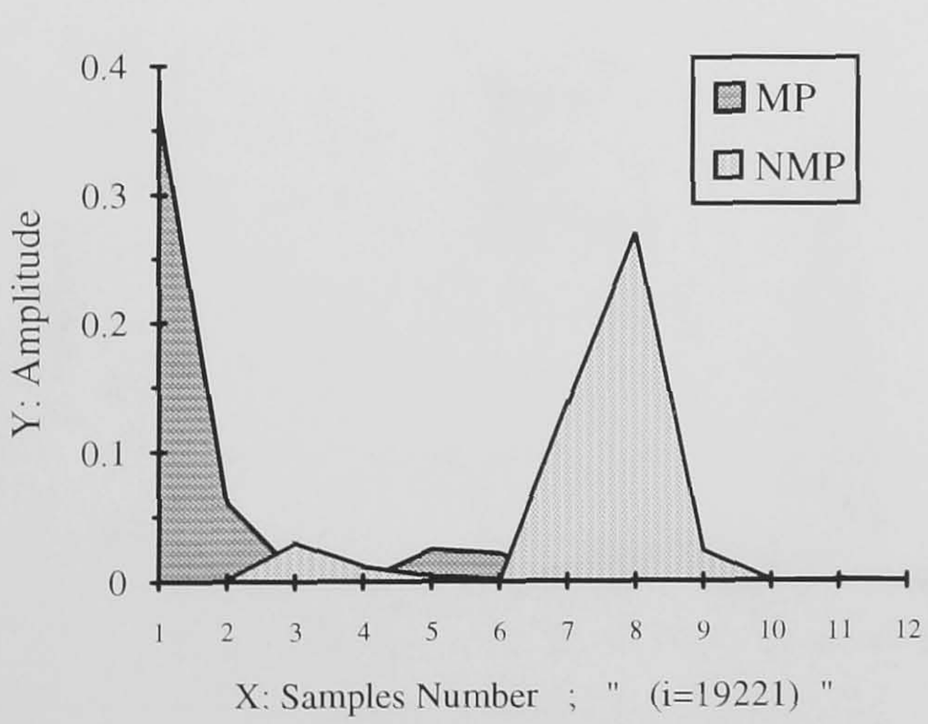

Y: Amplitude
MP
NMP
F-Taps
X: Samples Number ; " (i=54580) "

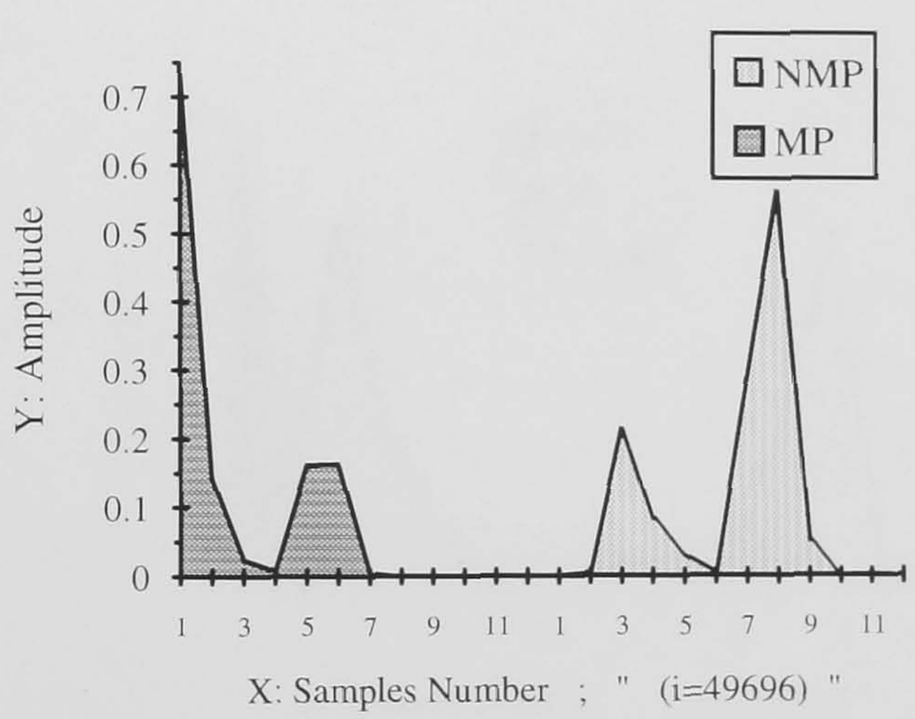


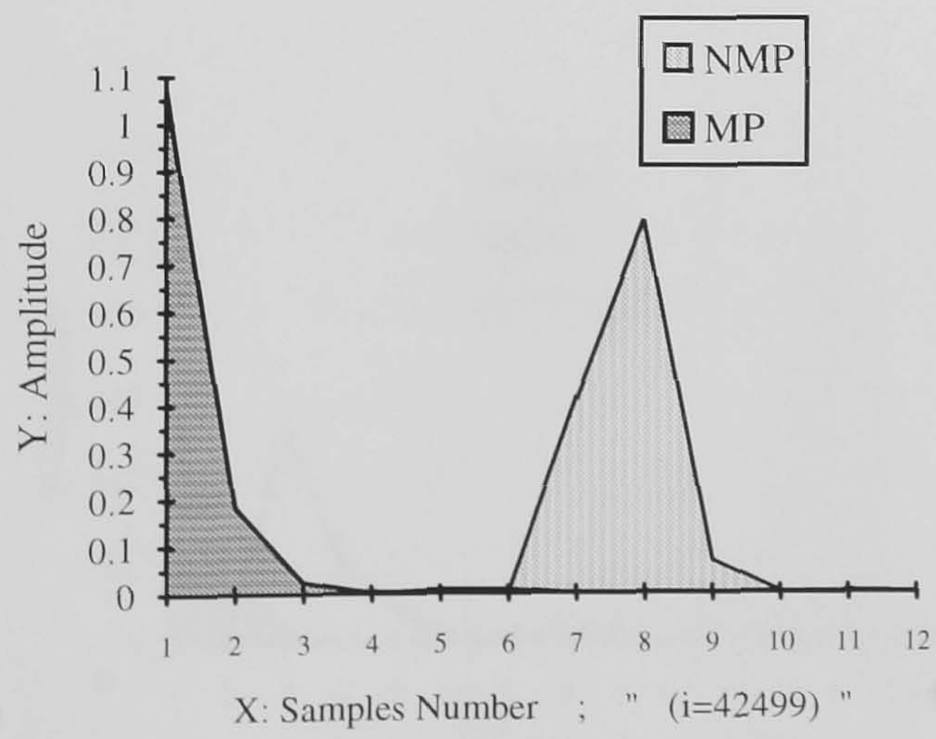












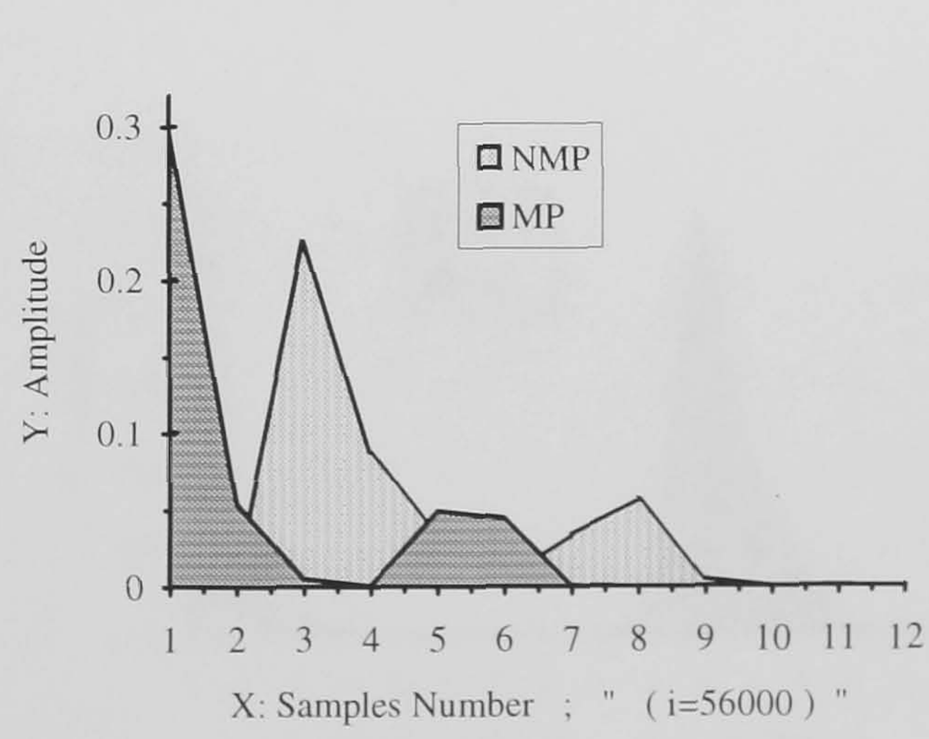
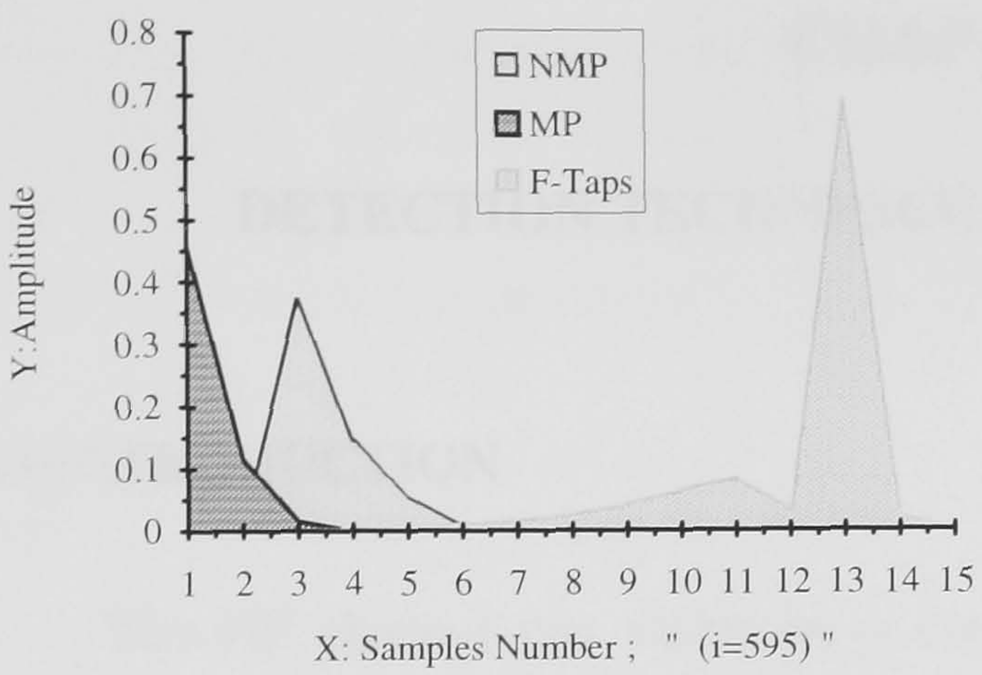
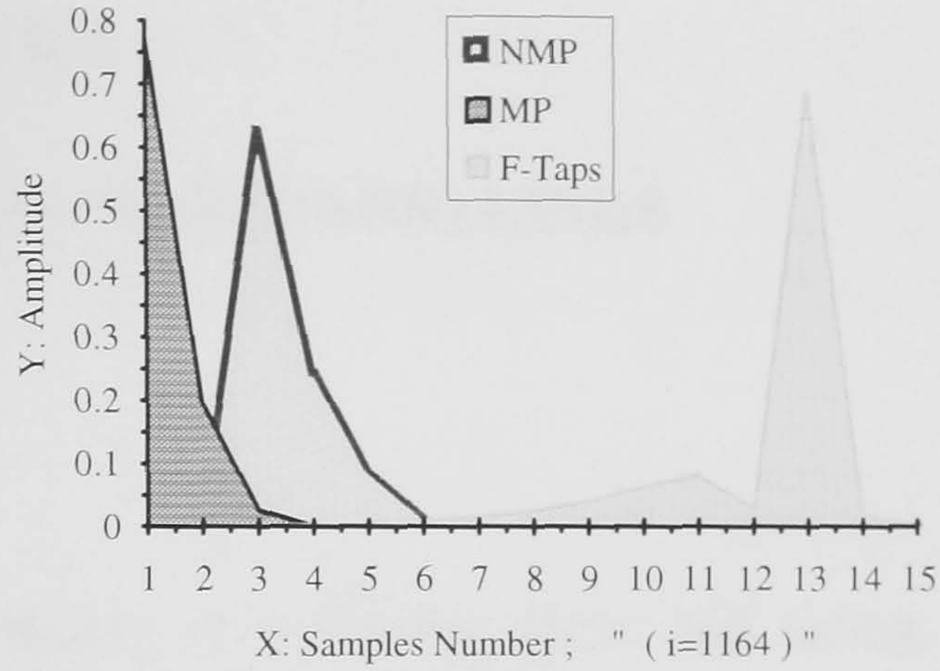
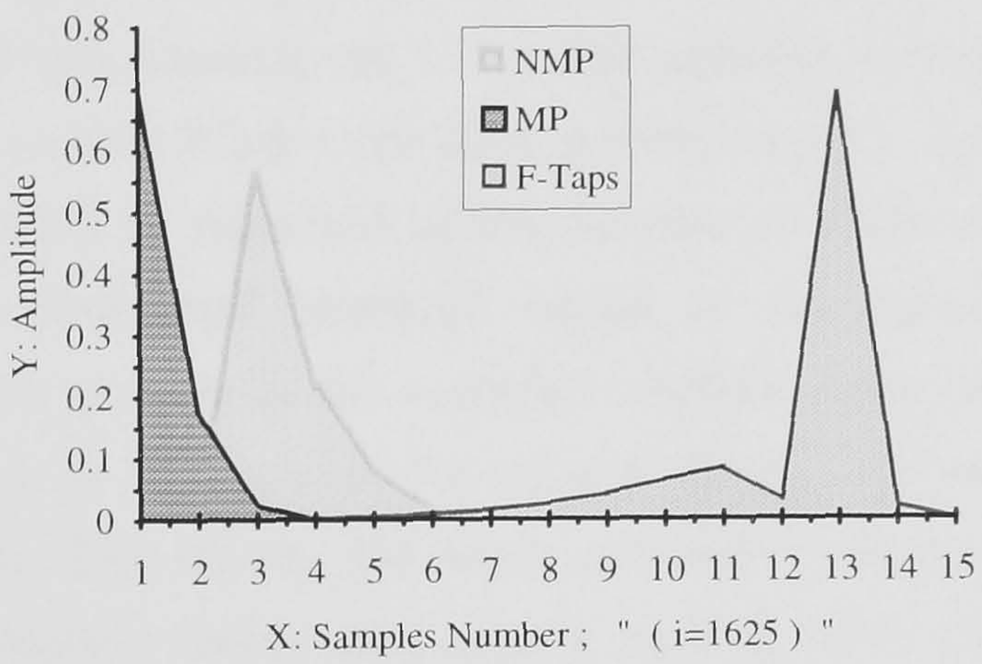
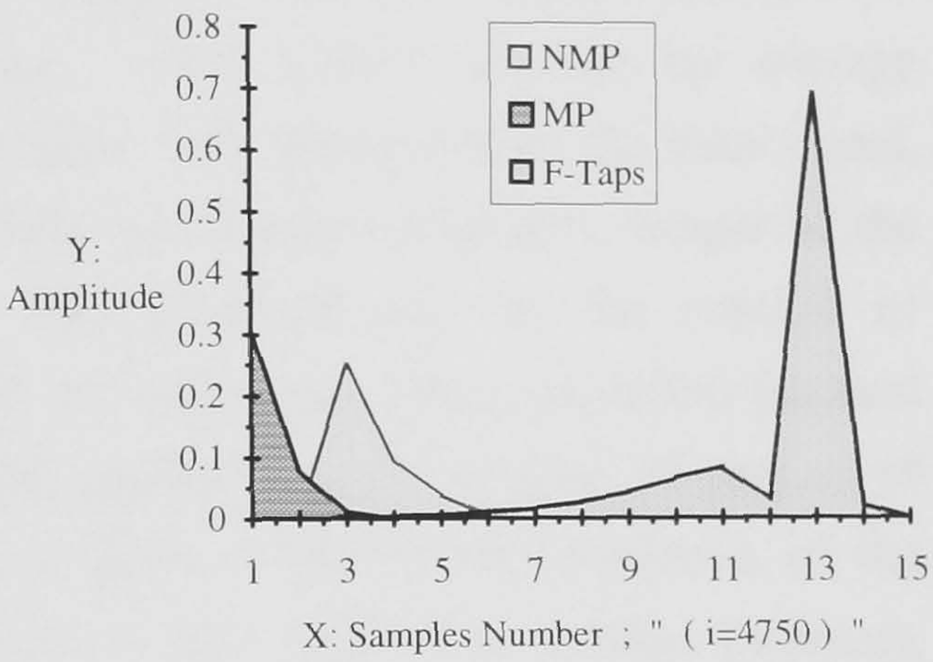


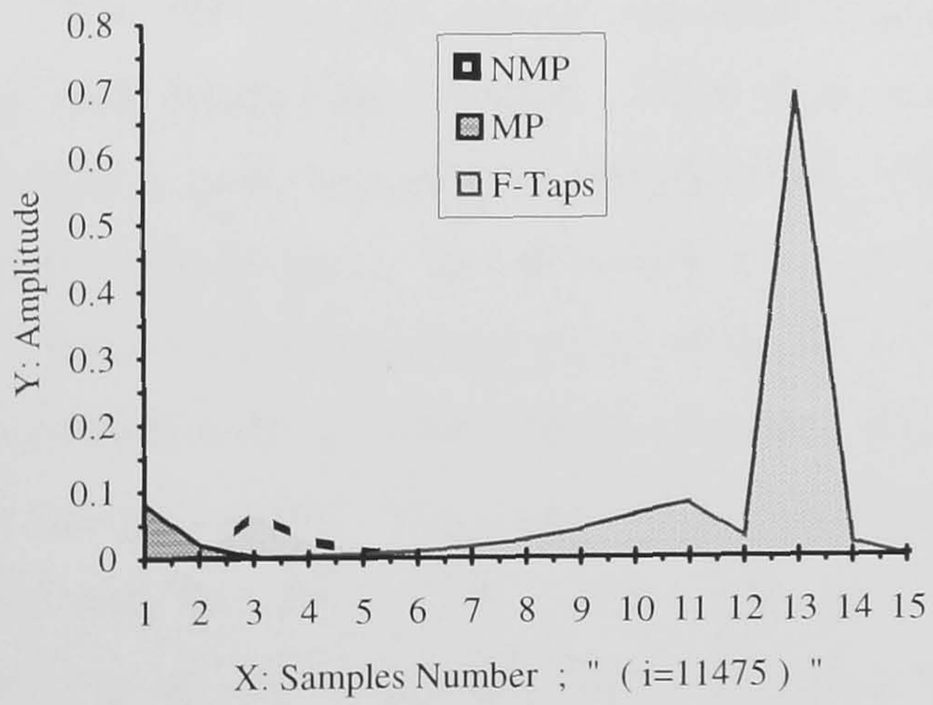
Fig. 4.5.7 Non-MP, MP SIR and Filter Coefficients for Channel3.

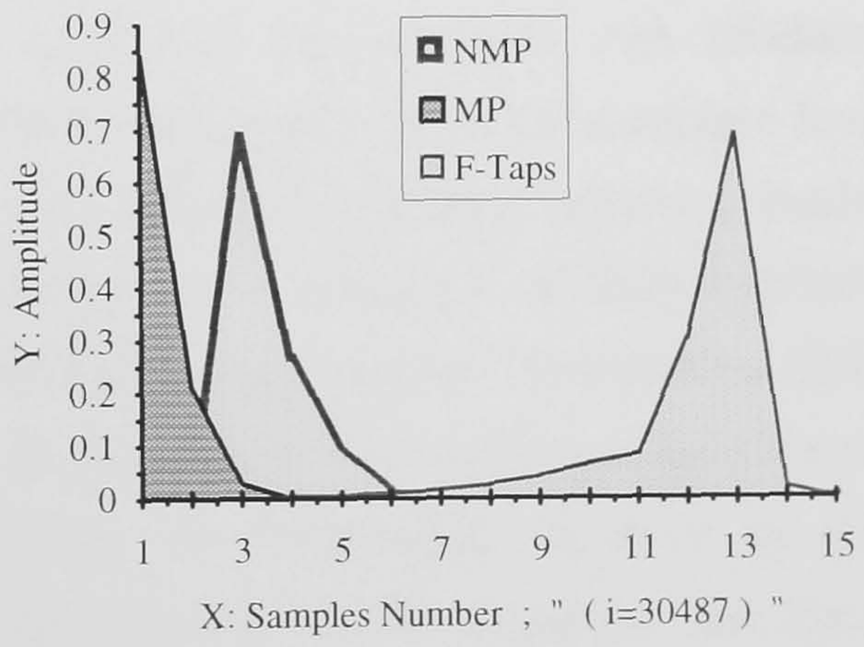


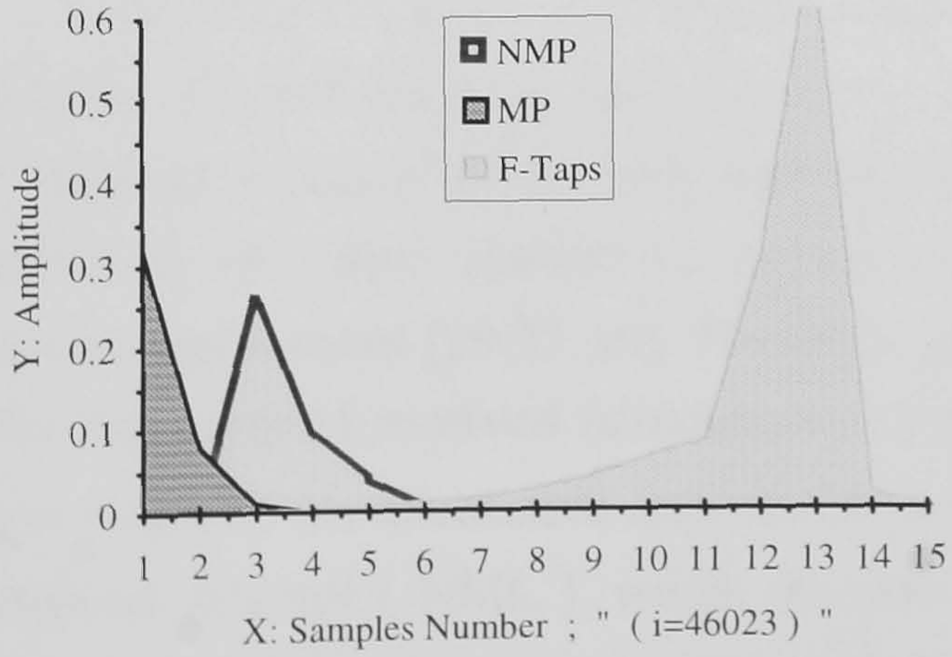












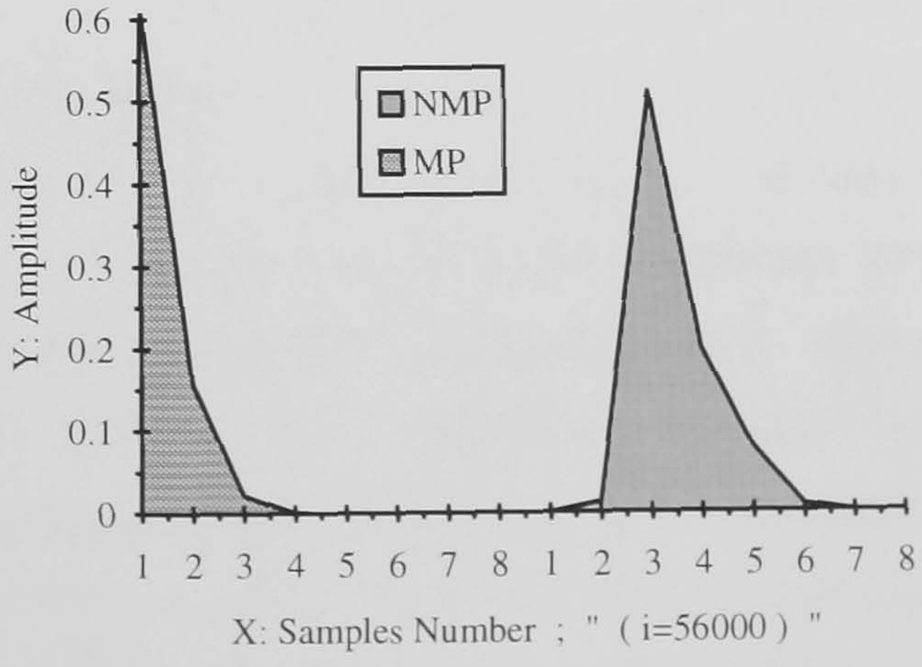


Fig.4.5.8 Non-MP., MP. SIR and Filter Taps-gain for Channel 4.

CHAPTER 5

DETECTION TECHNIQUES OVER HF RADIO LINKS

5.1 INTRODUCTION

The HF channel can introduce a combination of multipath effects and fading, leading to frequency-selective fading of the received signal. The two important parameters of this channel are : a.) The relative fading rate : which is here taken as the average number of fades per data symbol and b.) The relative time dispersion of the data signal, which is measured by the number of different data symbols involved in a sample of the demodulated baseband signal at the receiver, and therefore also by the number of components in the sampled impulse-response of the equivalent linear baseband channel ($L+1$). The greater the relative fading rate and the greater the relative time dispersion of the data signal, the more difficult it becomes to achieve the correct detection of the received fading data signal, such as when the relative time dispersion is large (3 *msec*), with ($L+1$) in the range (15-25) components.

The HF channel can be characterised as a multipath time-varying channel causing time and frequency spreads. This time varying multipath characteristic can produce destructive and constructive interference. The resulting fading can produce complete loss of transmitted signal. The detection of such signal thus affected is difficult and as a result, HF channels exhibit high error rates. It is well known for a sequence of data symbols transmitted over a band-limited channel, which introduces inter-symbol interference (ISI) and additive white Gaussian noise (AWGN), the optimum detection process is maximum likelihood detection [29,37,38]. The latter minimises the probability of error in the detection of the entire transmitted sequence, given that the whole sequence has been received [29,37].

Unfortunately, when m-level data symbols are transmitted and at the same time the SIR of the channel has more than a few non-zero components, as the case considered here, the implementation of a maximum likelihood detector by means of the Viterbi algorithm requires an excessive amount of storage and an excessive number of operations per received data symbol [29,37,38]. Therefore the Viterbi detector becomes complex, where in the detection of received massage, it involves basically (m^L) operations per received single element. An alternative approach that has been widely studied is a near-maximum likelihood detector (NML), which operates without undue complexity [29,42-46]. The latter needs the first few components of the SIR of the channel to be the largest. This can be done by using a suitable linear feed forward filter just ahead of the NML detector, employing a root finding algorithm to adjust its tap-gains, such as the one described in

Chapter 4. The latter can be adjusted directly and in a relatively simple manner, from an estimate of the SIR of the channel which can be determined by a channel estimator. Near maximum likelihood detection is derived from the Viterbi algorithm by reducing drastically the number of stored vectors, but without otherwise changing the basic method of operation [29,43,46].

In this chapter it is necessary first to describe the operation of the Viterbi algorithm. Various NML detectors are next described. The performance of the various NML detectors with and without linear feed forward filter are presented at the end of this chapter by means of computer simulation. The tests are carried out over three HF channel models namely channel 1, 2 and 3.

5.2 MODEL OF THE SYSTEM UNDER TEST

The model of data transmission system is shown in Fig.5.2.1. The system assumed here is the same as that described in Chapter 3. The model itself assumes a synchronous, serial data transmission system, with a four-level QAM signal, with an element rate of *2400 bauds*. The signal at the input of the adaptive filter at time $t=iT$, is the complex-valued quantity and is given by :

$$r'_i = \sum_{h=0}^{L} s_{i-h} \cdot p_{i,h} + v_i = P_i \cdot S_i^T + V_i \qquad 5.2.1$$

where $\{s_i\}$ are assumed to be statistically independent and equally likely to have one of four possible values, where :

$$s_i = s_{1,i} + j\, s_{2,i} \qquad 5.2.2$$

with $s_{1,i} = \pm 1$ & $s_{2,i} = \pm 1$, S_i^T is the transpose of S_i .

$\{v_i\}$ are the additive noise components, statiatically independent Gaussian random variables with zero mean and fixed variance σ_v^2. This sequence is scaled as illustrated in Chapter 3, and P_i is the sampled impulse response of the linear baseband channel $(L+1)$-component row vector given by :

$$P_i = [p_{i,0}, p_{i,1}, \ldots\ldots, p_{i,L}] \qquad 5.2.3$$

For the purposes of simplicity, theoretical analysis of the operation of NML detector is considered first in the absence of the linear feed-forward transversal filter that precedes the detector. Therefore the signal at the input of NML detector is given now by :

$$r_i = \sum_{h=0}^{L} s_{i-h} \cdot y_{i,h} + v_i = Y_i \cdot S_i^T + V_i \qquad 5.2.4$$

where :

$$Y_i = [y_{i,0}, y_{i,1}, \ldots\ldots, y_{i,L}] \qquad 5.2.5$$

is the scaled minimum phase sampled impulse response of the linear base-band channel. The latter is generated by the minimum phase algorithm whose input can be the estimated sampled impulse response of the channel P'_i that is formed by the channel estimator. But for this investigation the input of the latter algorithm is the actual SIR of the channel, i.e. perfect estimation is assumed ($P'_i = P_i$). this assumption is necessary to test the operation of various NML detectors over the HF channel employing the practical minimum phase algorithm to make the channel at least approximately minimum phased. The full detail of the operation of the minimum phase algorithm is given in Chapter 4.

It is assumed for practical purpose that : $y_{i,h} = p_{i,h} = 0$ for $h < 0$ & $h > L$.

The vector Y_i is a minimum phase (or approximately) sequence whose first component is always close to unity, such that :

$$y_{i,0} = 1.0 \qquad 5.2.6$$

An equivalent model of the system for testing various NML detectors by means of computer simulation is used, which is shown in Fig.5.2.2. The sequence of the SIR of the linear baseband HF channel is generated using a software model which is shown in Fig.5.2.3. The model can be programmed to be 3-sky wave or 2-sky wave with different parameters for testing the operation of detectors over the three channel model. The MP algorithm operates on the latter sequence to give the corresponding sequence of the scaled version minimum phased SIR of the linear baseband HF channel, which is fed to NML detector. The differential coded-decoded used in Fig.5.2.2 is explained in Appendix B-2. However, the noise sequence $\{v_i\}$ and the sampled impulse response sequence are scaled in order not to change the signal-to-noise ratio. The latter is given by eqn.3.2.49 as

$$SNR = \psi = 10 \cdot \log_{10} \frac{\bar{s}_i^2}{2\sigma^2} \qquad 5.2.7$$

where σ_v^2 is the variance of the real and imaginary part of discrete time noise sequence $\{v_i\}$. It should be noted here, that the minimum phased channel would have most of its energy concentrated in the first few components. Although not critical in a full Viterbi detector, minimum phasing of the channel is desirable when employing near-maximum likelihood detectors, (essential in the case of a conventional non-linear equaliser) [23,29]. Minimum phasing would be carried out by employing an adaptive linear filter, with root finding algorithm just ahead of the detector, where the function of this process is to convert the linear base-band channel from non-minimum phased to at least approximately

minimum phased, by replacing the zeros of Z-transform of the SIR of the channel which lie outside the unit circle, by the reciprocals of their complex conjugates. It may adjust the tap-gains of the linear feed forward filter, when the latter is used in the tests. For more detail on the model of data transmission system see Chapter 3.

5.3 VITERBI ALGORITHM DETECTOR

It is essential before describing the NML detector to outline the operation of a conventional Viterbi detector. The latter was originally developed in 1967, for the decoding of convolution codes [38]. The objective in this section is to explain the Viterbi algorithm in terms of its operation as maximum likelihood detector and the full detail of the algorithm is given elsewhere in [23,38].

Consider the (n)-componenets row vectors (sequences) :

$$S_n = [s_1, s_2, s_3, \ldots\ldots\ldots, s_n] \qquad 5.3.1$$

$$V_n = [v_1, v_2, v_3, \ldots\ldots\ldots, v_n] \qquad 5.3.2$$

$$R_n = [r_1, r_2, r_3, \ldots\ldots\ldots, r_n] \qquad 5.3.3$$

S_n, V_n, R_n are n-components vectors whose i^{th} elements are respectively, s_i, v_i, r_i for $i=1,2,3,\ldots\ldots,n$. They represent, the symbol values, noise components, and received samples respectively up to time (nT) in the transmission message. s_i may have one of the (m) possible values. $\{v_i\}$ are statistically independent random variable with zero mean and fixed variance. Also let :

$$X_n = [x_1, x_2, x_3, \ldots\ldots\ldots, x_n] \qquad 5.3.4$$

$$Z_n = [z_1, z_2, z_3, \ldots\ldots\ldots, z_n] \qquad 5.3.5$$

$$U_n = [u_1, u_2, u_3, \ldots\ldots\ldots, u_n] \qquad 5.3.6$$

where X_n, Z_n, U_n are n-components vectors whose i^{th} elements are x_i, z_i, u_i for $i=1,2,\ldots\ldots\ldots,n$. x_i can take on any of the (m) possible values of s_i, and z_i is a possible estimate of the signal component in r_i which is given by :

$$z_i = \sum_{h=0}^{L} x_{i-h} \cdot y_h \qquad 5.3.7$$

If u_i is a possible value of v_i then :

$$r_i = z_i + u_i \qquad 5.3.8$$

Then u_i is the estimate of v_i that correspond to z_i. The square of unitary (Euclidean) distance between R_n & Z_n is given by :

$$|U_n|^2 = |u_1|^2 + |u_2|^2 + \cdots\cdots + |u_n|^2 = |R_n - Z_n|^2 \qquad 5.3.9$$

Since $u_i = r_i - z_i$, for $i = 0,1,2,\ldots\ldots,n$, the quantity $|u_i|$ is the absolute value (modulus) of u_i. Assume the $\{s_i\}$ are statistically independent and equally likely to have any of their (m^{n+1}) different possible values, and the statistical independence condition for the noise samples $\{v_i\}$. Under these conditions, the maximum likelihood estimate of the transmitted sequence is given by the vector $(X_n)_{\max}$, where this is the value of the vector X_n such that $|U_n|^2$ is minimised [23,38]. Also $(X_n)_{\max}$ is the possible value of S_n that is most likely to be correct [23,29], given the received vector R_n. The quantity $|U_n|^2$ is said to be the cost of X_n. The Viterbi detector determines the cost of a given vector X_n, by using the components $\{x_i\}$ of this vector to form the estimate z_i of the quantity $(\sum_{h=0}^{L} s_{i-h} \cdot y_h)$, and to do this the receiver employs the linear feedforward transversal filter as shown in Fig.5.3.1. The tap-gains of this filter is given by eqn.5.2.5 which is the output of the minimum phase process. The $(n+1)$ $\{z_i\}$ then form the components of the vector Z_n that corresponds to the vector X_n. The cost of this vector is now calculated using eqn.5.3.9, where R_n & Z_n are known. Now the model of data transmission system can be considered as shown in Fig.5.3.2, which is more suitable for the explanation of the detection process.

The method of operation of the Viterbi algorithm is now explained as follows :

Just prior to the receipt of the sample r_{n+1}, (m^L) different sequences (vectors) $\{X_n\}$ are stored, and associated with each vector X_n is stored its cost $|U_n|^2$. Each of the (m^L) stored sequences $\{X_n\}$ has a different one of the (m^L) possible combinations of the values of its last L components $(x_{n-L+1}, x_{n-L+2}, \ldots\ldots, x_n)$, and each sequence has the smallest cost for its particular combination of the values of $(x_{n-L+1}, x_{n-L+2}, \ldots\ldots, x_n)$. These set of vectors are called "survivors". The cost of a vector is a measure of how likely it is that vector's components have the same values as the corresponding data-symbols [23,38]. The lower the cost, the more likely this is. The sequence X_n that has the smallest cost over all (m^L) stored sequences is the maximum likelihood sequence X_n, which is the possible sequence of the data symbol $\{s_i\}$ most likely to be correct.

On receipt of received signal sample r_{n+1}, each of the (m^L) survivors is expanded (m) ways to give (m^{L+1}) vectors $\{X_{n+1}\}$, where the values of the first n components $(x_{n-L+1}, x_{n-L+2}, \ldots\ldots, x_n)$ are as in the original vector, and the last component x_{n+1} takes on its (m) different possible values. A sequence X_{n+1}, of course, has $(n+1)$-

components. The cost $|U_{n+1}|^2$ of each vector X_{n+1} is next formed according to eqn.5.3.9, where :

$$|U_{n+1}|^2 = |U_n|^2 + |u_{n+1}|^2 \qquad 5.3.10$$

and $|U_n|^2$ is the cost of the vector X_n from which the vector X_{n+1} has been derived. Since $|U_n|^2$ has already been determined, it is only necessary to evaluate $|u_{n+1}|^2$ and add this to $|U_n|^2$. From eqn.5.3.7 and 5.3.8, then :

$$|u_{n+1}|^2 = |r_{n+1} - \sum_{h=0}^{L} x_{n+1-h} \cdot y_h|^2 \Rightarrow$$

$$|U_{n+1}|^2 = |U_n|^2 + |r_{n+1} - \sum_{h=0}^{L} x_{n+1-h} \cdot y_h|^2 \qquad 5.3.11$$

The resulting (m^{L+1}) costs are stored alongside the corresponding (m^{L+1}) vectors.

The detector next selects the vector, which has the smallest cost value of $|U_{n+1}|^2$, for each of the (m^L) possible combinations of the last L-components of X_{n+1}, (that is : $x_{n-L+2}, x_{n-L+3}, \ldots\ldots, x_{n+1}$). The resulting (m^L) selected sequence $\{X_{n+1}\}$ are next stored, together with their cost, ready for next detection. The maximum likelihood sequence $(X_{n+1})_{\max}$, is now chosen as the sequence with the lowest cost amongest the survivors .

The detection is carried out, on receipt of the whole message, such that the components $\{x_i\}$ in the sequence $(X_{n+1})_{\max}$ are taken to be detected values of the data symbols $\{s_i\}$, assuming of course, the duration of the entire message to be (nT) seconds. In practice, the number of data symbols transmitted in a message is likely to be large, this would involve an excessive amount of storage in operating with the vectors $\{X_n\}$. Thus, each X_n is replaced by the corresponding vector Q_n where :

$$Q_n = [x_{n-\mu}, x_{n-\mu+1}, x_{n-\mu+2}, \ldots\ldots, x_n] \qquad 5.3.12$$

Thus an appropriate delay of μ time intervals is introduced before the detection of the s_i, where $\mu > L$. Q_n is given by the last $(\mu+1)$-components of the particular X_n, and the cost $|U_n|^2$ of X_n is now taken as the cost of the corresponding Q_n. The detector holds now in store (m^L) $(\mu+1)$-components vectors $\{Q_n\}$ instead of (m^L) n-components sequence $\{X_n\}$. $\{Q_n\}$ having the (m^L) different possible combinations of the values of the last L-components $(x_{n-L+1}, x_{n-L+2}, \ldots\ldots, x_n)$. Each sequence is that with the smallest cost for a particular combination of values of its last L-components. Associated with each sequence (Q_n) is stored the corresponding cost $|U_n|^2$. The sequence with the smallest cost forms the last $(\mu+1)$-components of the maximum likelihood sequence X_n, and the value of the first component $x_{n-\mu}$ of this sequence $(Q_n)_{\max}$ is taken as the detected value $s'_{n-\mu}$ of the data symbol $s_{n-\mu}$. The delay in detection is now μ sampling intervals. The algorithm now

operates with the sequences $\{Q_n\}$ in the same manner as before. To avoid a noticeable reduction in tolerance to noise relative to the ideal Viterbi algorithm detector the delay (μ) should be large enough, since, in almost every case the segment $(x_1, x_2, \ldots\ldots, x_{n-\mu})$ becomes common to all survivors. Thus, the true maximum likelihood vector $(X_n)_{\max}$ is always held in the detector store. However μ is always chosen to be greater than $3(L+1)$ [23,29].

The algorithm may be required to start, in practice, by making one of the (m^L) stored vectors exactly equal to the first (L) data symbols and assigning to it a zero cost. The remaining (m^{L-1}) vectors may be chosen arbitrarily with a very high value of cost assigned to each of them. All of the arbitrarily chosen vectors would have been discarded after (L) recursion cycles, and the detector now holds (m^L) survivors. Every one of the new group of (m^L) vectors, would have originated from the vector to which was initially assigned a zero cost.

The process is simple to implement and can be applied, when suitably modified to any of the near-maximum likelihood detectors described in the following sections. It has been shown [23,38] the computational complexity of the Viterbi detector involves (m^{L+1}) operations $[(m^{L+1})$squaring $+(m^L)$comparison and (m^{L+1})addition] together with the storage of m^L (μ+1)-components vectors (storage location) and m^L costs $|U_n|^2$ in the detection of each received signal element. Since L+1 is the length of the channel response which is likely to be greater than *10* for an HF channel and m is the size of the signal set which is equal to *4* for QPSK system, therefore an excessive number of operation per received data symbols, together with an inordinate amount of storage, is usually required. Thus, the Viterbi algorithm is unlikely to be cost effective. An alternative method of reducing the complexity of the Viterbi detector, is the modification of the Viterbi algorithm itself by reducing the number of vectors held in the detector store at any particular time instant. Thus, instead of holding m^L survivors, the detector may hold k survivors ($k << m^L$), where these are chosen according to some criterion [42].

These detectors are known as reduced state Viterbi detectors, and their performance approaches suboptimum at low and moderate signal-to-noise ratios. Near-maximum likelihood detectors are a kind of reduced state Viterbi detectors, where k is much smaller than m^L and where all (k) stored vectors need not be survivors. Several such criteria have been developed [25,43-46,58].

Finally it should be noted that for the Viterbi detector to perform true maximum likelihood detection, the noise components in the recived signal samples at the input to the detector should be uncorrelated. This is not the case in practice where the receiver filter is not ideal (flat amplitude characteristic with sharp cut-off) but, it was shown by [59] that the previous condition need not necessarily be the case and that if the noise entering the

detector were correlated, maximum likelihood detection could still be performed. However, the aim of this investegation is not to study this problem and for full details of this matter the intersted reader should refer to [17,29,38].

Various NML detectors will be explained in the next sections, where an adaptive linear transversal filter just ahead of the NML detector including the minimum phase algorithm has been employed. The minimum phase algorithm is needed to make the sampled impulse response of the channel and filter at least approximately minimum phased or in the absence of the filter to convert the linear baseband channel from non-minimum phased to near minimum phased.

In practical applications of data transmission, where there is severe signal distortion such as in HF channel (amplitude and phase distortion), the sampled impulse response of the channel has a large number of components of relatively small magnitude. Under these conditions and with binary or quaternary signals, a Viterbi algorithm detector can be modified into one of the NML detectors to give a very considerable reduction both in amount of storage required and in the number of operations per received signal-element, but with no serious reduction in tolerance to noise. An important condition that must be satisfied here is that the first component of the SIR of the channel is not too small. To achieve this the NML detector should be preceded by an adaptive linear filter that employs minimum phase algorithm to make the resultant channel near minimum phased, or in other words it concentrates the energy in the earliest components of the SIR .

5.4 NEAR MAXIMUM LIKELIHOOD DETECTORS (NML)

Various NML detectors have been developed which are derived from the Viterbi detector described in section 5.3 [23,29,38]. These detectors hold only k vectors, where $k << m^L$ and they differ from the original Viterbi detector in the way of selecting the stored vectors $\{Q_n\}$. Since it is the selection process that characterises the new NML detector and distinguishes it from another, therefore it is convenient to describe the detector by considering the period immediately before and after the receipt of the sample r_n [23]. This reduces the complexity of some of the terminology used. These detectors were developed for operating over telephone channels, and it is well worth testing the NML detector when operating over HF radio links using the minimum phase algorithm. Therefore three different detectors, namely detector 1, detector 2, and detector 3 have been tested here. They are based on the NML detector, but with some modification in order to reduce the operations per received sample, which is useful when the minimum phase algorithm is applied. It is necessary to described first the original NML detector which was developed in [36,43,45].

5.4.1 DETECTOR 1 (NML DETECTOR)

Detector 1 holds in store only a limited and predetermined number of vectors (as survivors). The length of each of these vectors is fixed, and shorter than in the case of the Viterbi detector [36]. The model of the system is shown in Fig.5.2.1 and 5.3.2. The NML detector operates on the received sample $\{r_i\}$ eqn.5.2.4 to give at time $t=iT$ an estimate $\{s'_{i-\mu}\}$ of the transmitted data symbol $\{s_{i-\mu}\}$, where μ is an integer and represents the delay in detection. The operation of detector 1 can be described as follows :

Just prior to the receipt of r_i at time $t=(i-1)T$, the detector holds in store (k) different μ-components vectors (sequences) $\{Q_{i-1}\}$, where :

$$Q_{i-1} = [x_{i-\mu}, x_{i-\mu+1}, x_{i-\mu+2}, \ldots\ldots\ldots, x_{i-1}] \qquad 5.4.1$$

and $(x_{i-h}$, for $h = 0,1,\ldots\ldots,\mu)$, can take on any one of their 4 possible values of s_i, as given in eqn.5.2.2. Each vector Q_{i-1}, is formed by the last μ-components of the $(i-1)$-components vector X_{i-1} where :

$$X_{i-1} = [x_1, x_2, x_3, \ldots\ldots, x_{i-\mu}, \ldots\ldots, x_{i-1}] \qquad 5.4.2$$

and each vector X_{i-1} represents a possible transmitted sequence of data symbols $\{s_j\}$, for $(j=1,2,\ldots\ldots\ldots\ldots i-1)$.

It is assumed that $\mu > L$, where $L+1$ is the number of components in the SIR vector given by eqn.5.2.3. Each stored vector is associated with a cost B_{i-1} , where :

$$B_{i-1} = |u_{i-1}|^2 = \sum_{j=1}^{i-1} |r_j - \sum_{h=0}^{L} x_{j-h} \cdot y_{j,h}|^2 \qquad 5.4.3$$

which is also stored. The detector here considers that all $\{x_i\} = 0$, for $i \leq 0$.

On the receipt of the sample r_i, each stored vector Q_{i-1} is expanded into four $(\mu+1)$-components vectors $\{Q'_i\}$, where :

$$Q'_i = [x_{i-\mu}, x_{i-\mu+1}, x_{i-\mu+2}, \ldots\ldots\ldots x_{i-1}, x_i] \qquad 5.4.4$$

In each group of four vectors $\{Q'_i\}$ derived from any one vector Q_{i-1}, the first μ-components $\{x_i\}$are as in the original vector Q_{i-1}, and the last component x_i takes on its 4 different possible values s_i. Each of the resulting $4k$ expanded vectors $\{Q'_i\}$ has the cost given by :

$$B_i = B_{i-1} + |r_i - \sum_{h=0}^{L} x_{i-h} \cdot y_{i,h}|^2 \qquad 5.4.5$$

There are now a total of $4k$ different vectors $\{Q'_i\}$ and $4k$ associated costs $\{B_i\}$. Assuming the minimum phase algorithm work perfectly, which can make the first component of the SIR of the channel nearly the largest component and close to unity, such as :

$$y_{i,0} = 1.0 \qquad 5.4.6$$

then :

$$B_i = B_{i-1} + |r_i - \sum_{h=1}^{L} x_{i-h} \cdot y_{i,h} - x_i \cdot y_{i,0}|^2 = \\ = B_{i-1} + |r_i - \sum_{h=1}^{L} x_{i-h} \cdot y_{i,h} - x_i|^2 \qquad 5.4.7$$

where B_{i-1} is the cost of the vector Q_{i-1} from which Q'_i is derived. The detector now selects from the resulting set of $4k$ expanded vectors $\{Q'_i\}$, the vector with the smallest cost denoted as Q'_s. The latter is the most likely vector to be correct, over all components of possible values of x_i [29,43,45]. The detected data symbol $s'_{i-\mu}$ of the data symbol $s_{i-\mu}$ is taken as the value of the first component $x_{i-\mu}$ of vector Q'_s. Thus there is a delay in detection of μ sampling intervals, and μ is typically in the range 16 to 32 [23]. Any vector Q'_i, whose $x_{i-\mu}$ differs from $s'_{i-\mu}$ is then discarded by assigning to it an arbitrarily high value of cost. The detector, next selects from the remaining vectors (including Q'_s) k vectors with the smallest cost. The first component $x_{i-\mu}$ of all selected vectors is now omitted to avoid the similarity in selected vectors, to give the corresponding μ-components vectors $\{Q_i\}$, where :

$$Q_i = [x_{i-\mu+1}, x_{i-\mu+2}, \ldots\ldots x_{i-1}, x_i] \qquad 5.4.8$$

To prevent the overflow, i.e. to avoid an unacceptable increase in the value of the costs $\{B_i\}$over a long message, the smallest cost is subtracted from all costs of k selected vectors $\{Q_i\}$, so that the smallest cost becomes zero. This does not change the differences between the various costs.

Finally the detector stores now the new set of k vectors $\{Q_i\}$ with their costs and ready for the next detection process, that is the detection of $s'_{i-\mu+1}$, on receipt of r_{i+1}. It can easily be shown that the discarding of the vector $\{Q'_i\}$, just mentioned, ensures that the k stored vectors $\{Q_i\}$ are all different, provided only that they were different at the first detection process [23]. The latter can be easily done using a very simple starting-up procedure, such as to make the initial set of k stored vectors all correct and therefore the same as each other. One of the vectors has a cost of zero and the remaining vectors have a

very high cost [23]. The performance degradation of the detector can be due to the reason that all vectors $\{Q_i\}$ become the same after the receipt of a few million $\{s_i\}$ and also have very similar costs [23]. Another effect that occurs from time to time is that all stored vectors $\{Q_i\}$ have the same value of $x_{i-\mu}$, for some $i=0,1,2,\ldots\ldots$, so that $s_{i-\mu}$ is now effectively detected as $x_{i-\mu}$ after only the corresponding small delay. The degradation becomes very severe, when x_i has the same value in all k vectors $\{Q_i\}$. These may noticeably reduce the tolerance of the system to noise. The number of vectors k, retained by NML detector, is usually very small compared with the m^L survivors required to be retained by the Viterbi detector, which leads to a possible reduction in the detector tolerance to additive noise [17] compared with the Viterbi detector. Therefore the criterion employed in the choice of the k vectors $\{Q_i\}$ must be such that the degradation in tolerance to noise is kept to the minimum under the given conditions. Several techniques have been suggested for choosing these vectors [17,23,26,29], which attempt in various ways to prevent the stored vectors $\{Q_i\}$ from becoming the same. In this study two different detectors have been developed for operating over the HF channel model using a minimum phase process to adjust the linear filter that precedes the detector.

Bear in mind that the adaptive linear feed forward filter (including minimum phase process) that precedes the NML detector is an all-pass network that maximises the ratio of $|y_{i,0}|$ to the noise variance at the output of the filter (the filter now being considered as part of the channel) [29]. Under these conditions the NML detectors operate well, with a relatively small number of stored vectors. Where in the case that y_0, y_1, y_2 are all very small, the NML detectors become quite unsuitable and satisfactory operation is no longer achieved over a channel with severe distortion.

Finally, detector 1 requires the computation per received data signal, of $4k$ costs and k searches, i.e. k comparisons through these $4k$ costs. It requires $2k$ permanent storage locations. The operation of detector 1 can be illustrated by means of a simple example, taking the number of vectors k equal to 8, as shown in Fig.5.4.1.

5.4.2 DETECTOR 2

Detector 2 is a simple modification of detector 1, in order to reduce the storage and operations required by detector 1. This detector operates as follows :

Just prior to the receipt of the received sampled r_i, the detector holds in store k different vectors $\{Q_{i-1}\}$, together with the associated cost $\{B_{i-1}\}$ as in detector 1. eqns.5.4.1 and 5.4.3.

On receipt of r_i the detector expands each vector $\{Q_{i-1}\}$ into the corresponding two vectors $\{Q'_i\}$, where :

$$Q'_i = [x_{i-\mu}, x_{i-\mu+1}, x_{i-\mu+2}, \ldots\ldots\ldots x_{i-1}, x_i] \qquad 5.4.9$$

with the smallest cost instead of *4* vectors. The first μ-components of each of the two vectors $\{Q'_i\}$ derived from any one Q_{i-1} are as in the original vector Q_{i-1}, and the last component x_i takes on the two different values for which the cost of Q'_i is smallest. The two values of x_i are determined as in the pseudo binary Viterbi detector [23,45,44], in the following way : The cost of vector Q'_i is given by,

$$B_i = B_{i-1} + |r_i - \sum_{h=1}^{L} x_{i-h} \cdot y_{i,h} - x_i \cdot y_{i,0}|^2 =$$
$$= B_{i-1} + |r_i - \sum_{h=1}^{L} x_{i-h} \cdot y_{i,h} - x_i|^2 = B_{i-1} + |C_i|^2 \qquad 5.4.10$$

where :

$$C_i = (r_i - \sum_{h=1}^{L} x_{i-h} \cdot y_{i,h}) - x_i = d_i - x_i \qquad 5.4.11$$

and $d_i = (r_i - \sum_{h=1}^{L} x_{i-h} \cdot y_{i,h})$

Assume that eqn.5.4.6 holds. Bear in mind that the $(x_i, x_{i-1}, ..., x_{i-L})$ are as in the vector Q'_i eqn.5.4.9, B_{i-1} is the cost of the vector Q_{i-1} from which Q'_i is derived. Since the two vectors $\{Q'_i\}$ derived from any particular vector Q_{i-1} differ only in the values of x_i, therefore they have the same value of B_{i-1} eqn.5.4.3 but different values of $|C_i|^2$, so that the two vectors $\{Q'_i\}$ with the smallest cost $\{B_i\}$ also have the smallest $|C_i|^2$.
The value of x_i for which $|C_i|^2$ and therefore B_i are smallest, is determined first by selecting the possible values of the real and imaginary parts of x_i that are closest, respectively, to the real and imaginary parts of (d_i). The selection process is done by applying a simple threshold level comparison [26,44,45], without evaluating $|C_i|^2$ itself .

The value of x_i just determined is now used to form the quantity C_i eqn.5.4.11 and depending upon whether the real or imaginary part of C_i has the greater magnitude, the second selected value of x_i is determined by changing the real or imaginary part, respectively, of x_i eqn.5.4.11 to the next adjacent possible value in the direction given by the sign of the corresponding real or imaginary part of C_i . If there is no possible value of x_i in the location given by the selection process, the detector repeats the process for the real or imaginary part of C_i having the smaller magnitude. The real or imaginary part of original value of x_i eqn.5.4.11 now has one of its extreme (most positive or most negative) values. If again the selection process does not lead to a possible value x_i, the process is repeated for the real or imaginary part of C_i having the smaller magnitude but now with the sign of this reversed. The real and imaginary parts of the original value of x_i eqn.5.4.11 now have one of their extreme values. The value of x_i given by the above procedure is always that for which $|C_i|^2$ in eqn.5.4.10 has the second smallest value, giving therefore the second smallest cost B_i.

Clearly, the second selected value of x_i differs from the first in only the real or imaginary part and by the difference between two adjacent possible values of this part. The initial selection process just described is relatively simple to implement and does not involve the computation of any cost. The detector now holds $2k$ vectors $\{Q'_i\}$ with their costs, the remaining values of x_i being ignored here. The detection process continues exactly as in detector 1. This detector requires the computation of $2k$ costs and k comparisons through $2k$ costs per received data symbol. The memory requirement, in terms of permanent storage locations, is $2k$ and the delay in detection is μ sampling intervals. Fig.5.4.2 shows the operation of detector 2 in vector representation.

5.4.3 DETECTOR 3

Detector 3 has been developed to work with minimum phased HF channel, where now the detector is a simple modification of detector 1 and detector 2 and operates as follows :

Initially the detector holds in store k μ-component vectors $\{Q_{i-1}\}$ with their costs $\{B_{i-1}\}$. The k vectors now divided into three groups, where the first, the second and the third group has k_1, k_2 and k_3, $\{Q_{i-1}\}$ vectors respectively.

On the receipt of r_i, each vector Q_{i-1} in group one is expanded into four (*4*) vectors $\{Q'_{i,1}\}$ as in detector 1 to give $4k_1$ vectors in group1 with their costs $\{B_{i,1}\}$ eqn.5.4.3. Each vector $\{Q_{i-1}\}$ in the second group is expanded into two vectors $\{Q'_{i,2}\}$ with smallest cost as in detector 2 to give $2k_2$ vectors with their costs $\{B_{i,2}\}$. Finally the detector expands each vector $\{Q_{i-1}\}$ in group3 into the corresponding one vector with smallest cost to give k_3 vectors $\{Q'_{i,3}\}$ with their costs $\{B_{i,3}\}$, where $(k_3 = k - k_1 - k_2)$.

As before, the first μ-components of vector Q'_i are again as in the original vector Q_{i-1}, and the last component x_i, now takes one of its *4* possible values given by the *4* possible values of s_i in group1, where in group2 and group3 it is determined by threshold level comparison as in detector 2.

The detector now holds $k_{tot} = 4k_1 + 2k_2 + k_3$ expanded vectors $\{Q'_i\}$, together with the associated costs $\{B_i\}$. Next, the detector selects from the set of k_{tot} expanded vectors $\{Q'_i\}$, the one with the smallest cost denoted Q'_s, and the detection process continues exactly as in detector 1, with the detection of $s'_{i-\mu}$ and the selection of k vectors $\{Q_i\}$. This arrangement involves the computation, per received data symbol of $k_{tot} = 4k_1 + 2k_2 + k_3$ costs and k searches, i.e. k comparison through these costs.

This detector is very useful when $k \geq 8$ & $m \geq 4$, where it gives the same performance as detector 1 [114] with less computational requirement per received data element. Bear in mind that the successful operation of such a NML detector over an HF channel requires the channel to be minimum phased.

Note: k_1, k_2 and k_3 can take any integer in the range $1 \rightarrow k/2$ and for example $k=16,\ k_1=2,\ k_2=4\ ,k_3=10 \Rightarrow k_{tot}=26$ expanded vectors.

The operation of detector 3 is illustrated via an example shown in Fig.5.4.3.

5.5 CONVENTIONAL NON-LINEAR (Decision Feedback) EQUALISER

The performance of a conventional non-linear equaliser operating over an HF channel under the same conditions considered previously is presented here, for comparison purposes with various NML detectors. The arrangement of a conventional non-linear equaliser (NLQ) is shown in Fig.5.5.1. The tap-gains $y_{i,h}$, for $h=1,2,....L$ of the feedback section of the NLQ are adjusted every sample interval by the minimum phase algorithm. The detailed structure of the equaliser was given in Chapter 4. The delay in detection is zero ($\mu=0$), so that the detected data symbol s'_i is determined from r_i. The output signal of the feed-back section is given by :

$$g_i = \sum_{h=1}^{L} s_{i-h} \cdot y_{i,h} \qquad 5.5.1$$

Where g_i is an estimate of intersymbol interference (ISI) in the received signal r_i. The signal e_i at the input of the detector is given by :

$$e_i = r_i - g_i \qquad 5.5.2$$

With the correct detection of the data-symbols $(s_{i-1}, s_{i-2}, ..., s_{i-L})$ and using eqns.5.2.4 and 5.5.1, then :

$$e_i = s_i + u_i \qquad \text{(assuming } y_{i,0}=1.0\ \text{)}. \qquad 5.5.3$$

The detected data symbol s'_i is now taken as its possible value closest to e_i. Errors in detection can occur in bursts due to the incorrect detection of one or more of the data symbols leading to ISI in e_i and so greatly increasing the probability of error in s'_i. The performance of the NLQ and NML detectors are given at the end of this chapter .

5.6 SIMULATION RESULTS AND DISCUSSION

Tests by computer simulation have been carried out to assess the performance of the various NML detectors over three HF channel models (channel 1, 2 and 3). Also the performance of a conventional non-linear equaliser has been included, under the same conditions to determine in each case the tolerance of the detector to additive noise. The model of the data transmission system used is that described in section 3.6.2 and section 5.2.2. The information to be transmitted by the modulator is converted to a serial stream

of four-level QAM signal elements (QPSK) with an *1800Hz* carrier and an element rate of *2400 baud*. The voice-band QPSK signal, whose band-width extends from about *600* to *3000 Hz* is fed to the HF channel model.

Tests have been carried out over channel 1, 2 and 3 whose details are given in Chapter 2. It is first necessary to consider all conditions assumed in these tests, in order the obtained results are viewed in their proper perspective. As mentioned previously, it is assumed that the SIR of the linear base-band channel is known, i.e. perfect estimation of the SIR is assumed and hence the minimum phase algorithm operates directly on the actual SIR of the channel. This is the most important assumption in testing the performance of the NML detectors operating over fading channels, and using practical minimum phase algorithm to make the channel and filter at least approximately minimum phase, rather than using the NAG routine from the software library to do this. Bear in mind that the subsequent results do not represent the practical system performance, in the absolute sense, due to the above assumption. The objective here is to chose a NML detector that is effective in combating the signal fading experienced over such channels using a minimum phase process. The signal-to-noise ratio ($\psi \equiv SNR$) in all tests is given in dB, where :

$$SNR = \psi = 10 \cdot \log_{10} \cdot \frac{E_b}{0.5 N_0} \qquad 5.6.1$$

and E_b is the average transmitted energy per bit at the input to the HF channel, and $(N_0/2)$ is the two-sided power spectral density of the noise. An average of around $(2 \cdot 10^6)$ data symbols $\{s_i\}$ have been involved in the drawing of each curve. The *95%* confidence limits of the results are generally of the order of $\pm 0.5 dB$.

The delay in detection introduced by NML detectors is μ=*32* symbol intervals, whereas in the case of conventional non-linear equaliser no delay is introduced in detection. The length of the SIR of channels 1, 2 and 3 are respectively *20*, *20* and *18*. The number of stored vectors (*k*) used in each detector can be in the range *4* to *16*.

In all results presented here, the same fading sequence has been employed, for a given channel under test, i.e. the same sequence of values of the sampled impulse response of the overall linear base-band channel (Fig.5.2.3), was used. This allows the most accurate comparison possible between the various detectors, when operating over a given channel. The same noise sequence is not used, where the noise and data sequences have been varied at each (SNR). Differential encoding-decoding is employed to prevent the prolonged error bursts caused by the rapid phase changes, when the transmitted signal is in deep fade. This is done by coding the difference in phase between two consecutive symbol values. The performance of a conventional non-linear equaliser is shown in each figure for the sake of comparison and is denoted (NLEQ). A short symbol related with each detector is written beside each plot in all the results. For example *d1k16Ch1* is detector 1 with, the number of stored vectors *k=16*, over channel 1 "*Ch1*". The number of

stored vectors (*k*) used in each detector can be in the range between *4-16*. For detector 1 and 2 the *k* vectors are expanded four ways and two ways respectively, but for detector 3 the expansion is not fixed, where k_1 vectors in group1, k_2 vectors in group2 and k_3 vectors in group3 are expanded 4-ways, 2-ways and 1-way respectively, to give k_{tot} vectors. k_1, k_2 lie in the range *1--k/2*, $k_3=k-k_1-k_2$ and $k_{tot}=4k_1+2k_2+k_3$. For example when $k=16$, $k_1=2$, $k_2=4$, then $k_3=10$, and $k_{tot}=26$ expanded vectors.

The results of the computer simulation tests are summarised in Figs.5.6.4-5.6.19, where Figs.5.6.1-5.6.3 show the amplitude variation of the overall sampled impulse response (SIR) of the three channels. In terms of the depth of the fades, channel 1 (Fig.5.6.1) is the mildest, with the deepest fade around *-12dB*, and channel 2 is the severest, with the deepest fade around *-18dB* (Fig.5.6.2). Channel 3 with the lower frequency spread (*fsp=1Hz*) has the slower variations of the signal level as shown in Fig.5.6.3. Therefore channel 2 is considered as the worst of the three channels, exhibiting the deepest fades and the faster variations.

The minimum phase algorithm (MPA) described in Chapter 4 is employed here to generate the SIR of the overall linear base-band channel and the model of the system is shown in Fig.5.2.2. The resultant SIR is now at least approximately minimum phased, thus the signal at the input to the detector is only severely distorted in amplitude, where the phase distortion that was introduced by the channel is now removed by the algorithm. In Fig.5.6.4-5.6.17, the number of iterations used by the minimum phase algorithm is set to *40*, but this is not critical as we will see later. The algorithm operates on the roots (zeros) of the Z-transform of the SIR that have absolute values greater than *d*, where *d* is chosen to be *1.05* in obtaining the results in Figs.5.6.4-5.6.17. The performance of detector 3 tested with different values of *d* will be presented at the end of this section.

Figs.5.6.4-5.6.7 show the performance of the various NML detectors when operating over channel 1. The figures indicate that an advantage of about *1* to *4dB* can be gained over the conventional non-linear equaliser (NLEQ). Detector 1 (*d1k16ch1*) and detector 3 (*d3k16:24ch1*) with ($k=16$, $k_1=2$ and $k_2=4$) are the best. As expected, for a given detector, the performance improves as the number of stored vectors is increased. Mainly when *k* is increased from *4* to *8*, than from *8* to *16*, with the latter offering only a little improvement. Therefore increasing the number of stored vectors (*k*) beyond *16* would give negligible improvement in performance in a given system. Also it can be seen from Fig.5.6.7 that detector 3 with $k=16$ gives very close performance to detector 1, with less computational complexity.

Figs.5.6.8-5.6.11 show the performance of the NML detectors and non-linear equaliser when operating over channel 2. Again the non-linear equaliser is the worst of the detectors and the same observation can be seen as that for channel 1. It can be seen from Fig.5.6.11 that detector 3 with $k=16$ again gives the same performance as detector 1, and they are the best.

Figs.5.6.12-5.6.14 present the performance of the detectors and NLEQ over channel 3. These figures indicate the same results as in the pervious figures for channel 1 and 2.

As the result of the above observation, all detectors perform best over channel 1 and worst over channel 2. The inferior performance of the detectors over channel 2 is essentially due to the increase in signal distortion, and more rapid fading rate. Since it has two paths with a large frequency spread and time delay, where channel 1 has three paths with the same frequency spread and time delay, the latter would exhibit the mildest fades due to the probability of the signal being in a deep fade at any particular time instant, decreases as the number of paths increase.

All detectors give better performance than the conventional non-linear equaliser, particularly at low error rates. The latter is between *1* and *4dB's* worse than the worst of NML detectors. Since the lower the error rate the more of the errors occur during the deepest fades, therefore the non-linear equaliser is more affected than the NML detectors by deep fades, due to its inherent error propagation tendencies. Since channel 2 is the worst, the initial choice of a detector would be based on its performance, since a detector that operates satisfactorily over poor channels, should perform well over less severe fading channels. Therefore the initial choice of a detector would be based on its performance over channel 2 and its computational complexity. The computational complexity of the three detectors with the number of stored vectors $k=16$ is summarised in Table 5.6.1, in terms of cost calculation and cost comparison. Table 5.6.1 indicates that detector 3 requires less computational complexity than detector 1 and 2.

As the result of this discussion, it appears the detector 3 involves a quite reasonable number of operations per received signal element even with *16* stored vectors, and gives very close performance to detector 1 with less computation complexity. This makes detector 3 the best candidate for a combined detection estimation system. Fig.5.6.15 shows the performance of detector 3 with $k=16$ (*d3k16:24*) when operating over channel 1, 2 and 3. It can be seen from the figure that the worst performance is over channel 2.

Other tests are carried out to test the effectiveness of the feed-forward transversal filter that precedes the detector, where the tap-gains of this filter are adjusted adaptively at every instant of time using the method described in Chapter 4. Fig.5.6.16-5.6.17 show the performance of detctor 3 and a conventional non-linear equaliser when operating over channel 1 and 2 under the same conditions. The symbol (*NF*) in the labelled denotes no filter, whereas the symbol (*WF*) denotes with filter. It can be seen from the figures that the linear feed-forward filter gives little improvement when the NML detector is employed, where in the case of the non-linear equaliser the filter does not give any improvement. This is due to the reason that, some times the minimum phase algorithm misses some roots to allocate (process) them leading to miss-adjustment of the filter. This is the reason why we need a good NML detector to operate over fading channels where, a realisable

MP algorithm is employed. The close performance of all NML detectors is due to the successful operation of the MP algorithm over the three channels.

Finally detector 3 also has been tested with different values of d to that used by the MP algorithm over channel 1 and 2, and the results are shown in Figs.5.6.18-5.6.19. The value of d has been set to *1.002*, *1.05*, and *1.1*, and for any value of d the MP algorithm processes those roots that are greater than d, the remaining roots being left unchanged. The number of iteration processes (j) used by the algorithm is also reduced from *40* to *30*. Figs.5.6.18-5.6.19 indicate that as d get smaller the performance of the detector and non-linear equaliser is improved. The reason for this improvement is due to the obvious reason that as d get smaller more roots of the SIR of the channel that lie outside the unit circle will be processed and the resultant SIR becomes closer to minimum phase. Also the figures indicate that when j is reduced to *30* there is no change in the performance of the detector over both channels. The aim for testing the detector with different value of d and j is not only to test its performance improvement, but also to obtain a feel for the best compromise between the complexity of the adaptive filter and that of the detectors. The greater value of d, the fewer operations per received sample r_i are needed to hold the filter correctly adjusted [39]. Thus reducing d increases the complexity of the adaptive filter. However, it is clear from Figs.5.6.18-5.6.19 that the detector and NLEQ are seriously affected by changing the value of (d), and the smaller value of d (*1.002*) gives the best result. Therefore as the number of iterations can be reduced without affecting the result, the smaller value of d (*1.002*) can be used without making the system unduly more complex.

Whenever there is linear-phase amplitude distortion, and the MP algorithm converts this distortion to at least approximately minimum phase amplitude distortion, a useful improvement in performance over that of a conventional non-linear equaliser can be achieved by a suitable NML detector. The results also suggest that detector 3 with MP algorithm that uses a number of iteration processes equal to *30 or 40*, with the value of d equal to *1.05 or 1.002*, should be chosen as the preferred system for combined detection estimation. In the latter, perfect estimation is not assumed any more and the MP algorithm operates directly on the estimated SIR of the channel. Chapter 6 and 7 will deal with the channel estimators and the combined detection estimation over fading channels respectively.

	number of costs per data symbol	number of searches through the costs.
Detector 1	64	16
Detector 2	32	16
Detector 3	26	16

Table 5.6.1 Relative computational complexity of detectors (k=16).

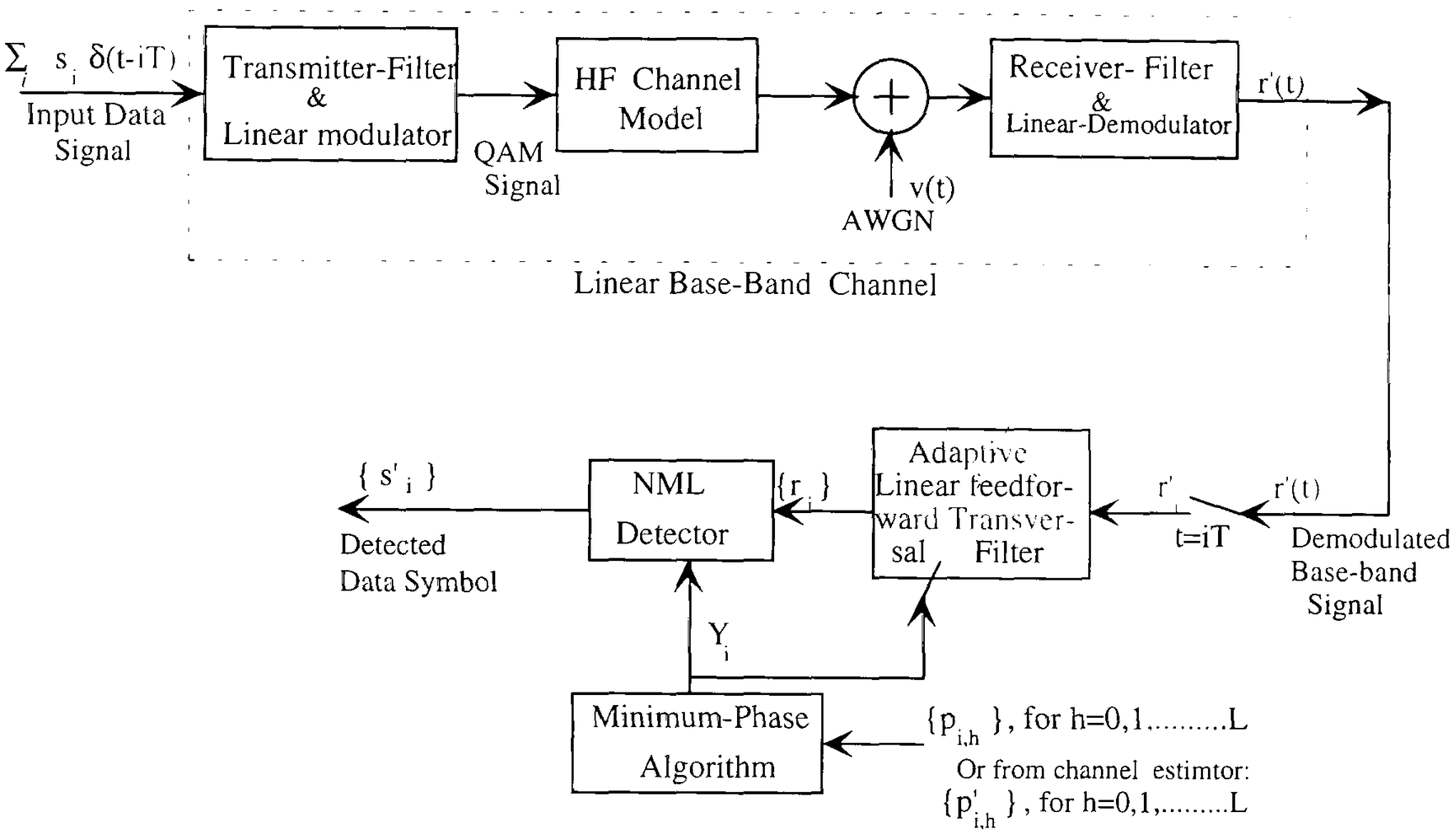


Fig.5.2.1 : Model of The Data Transmission System .

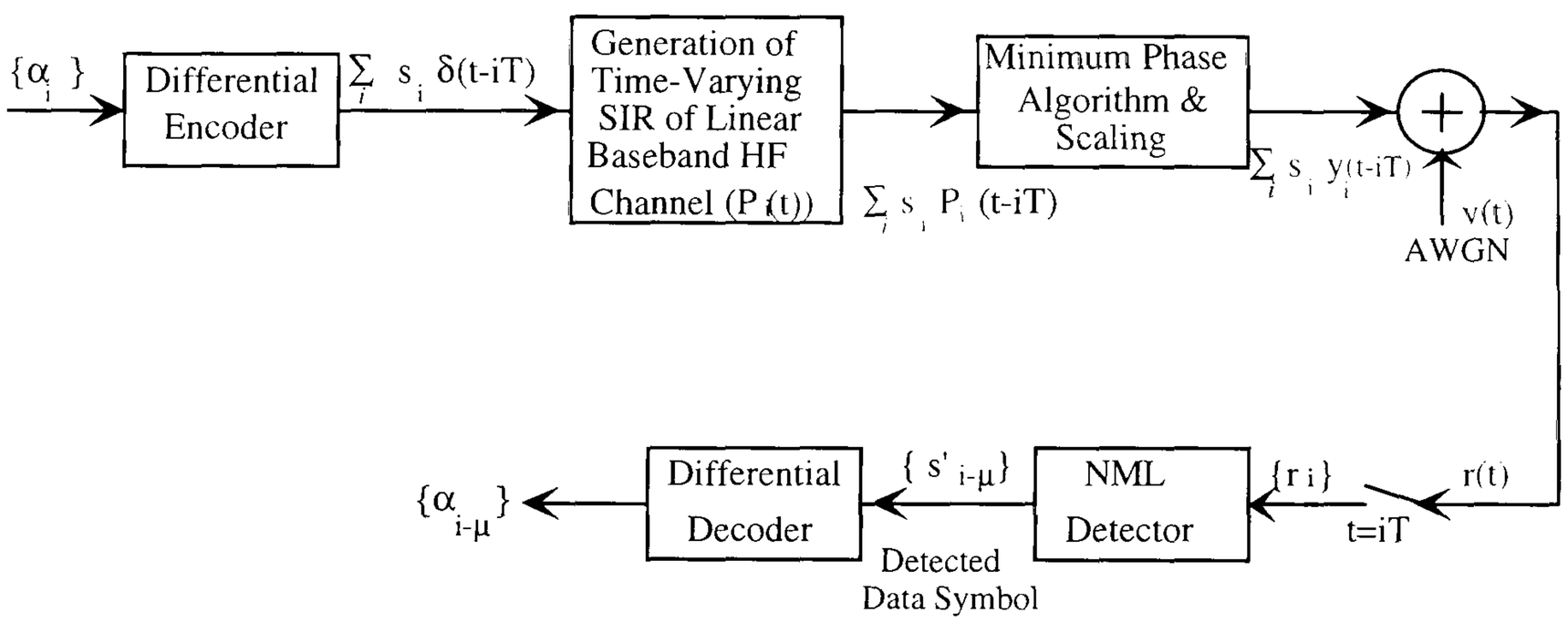


Fig.5.2.2 : Model of The System used in The Computer Simulation.

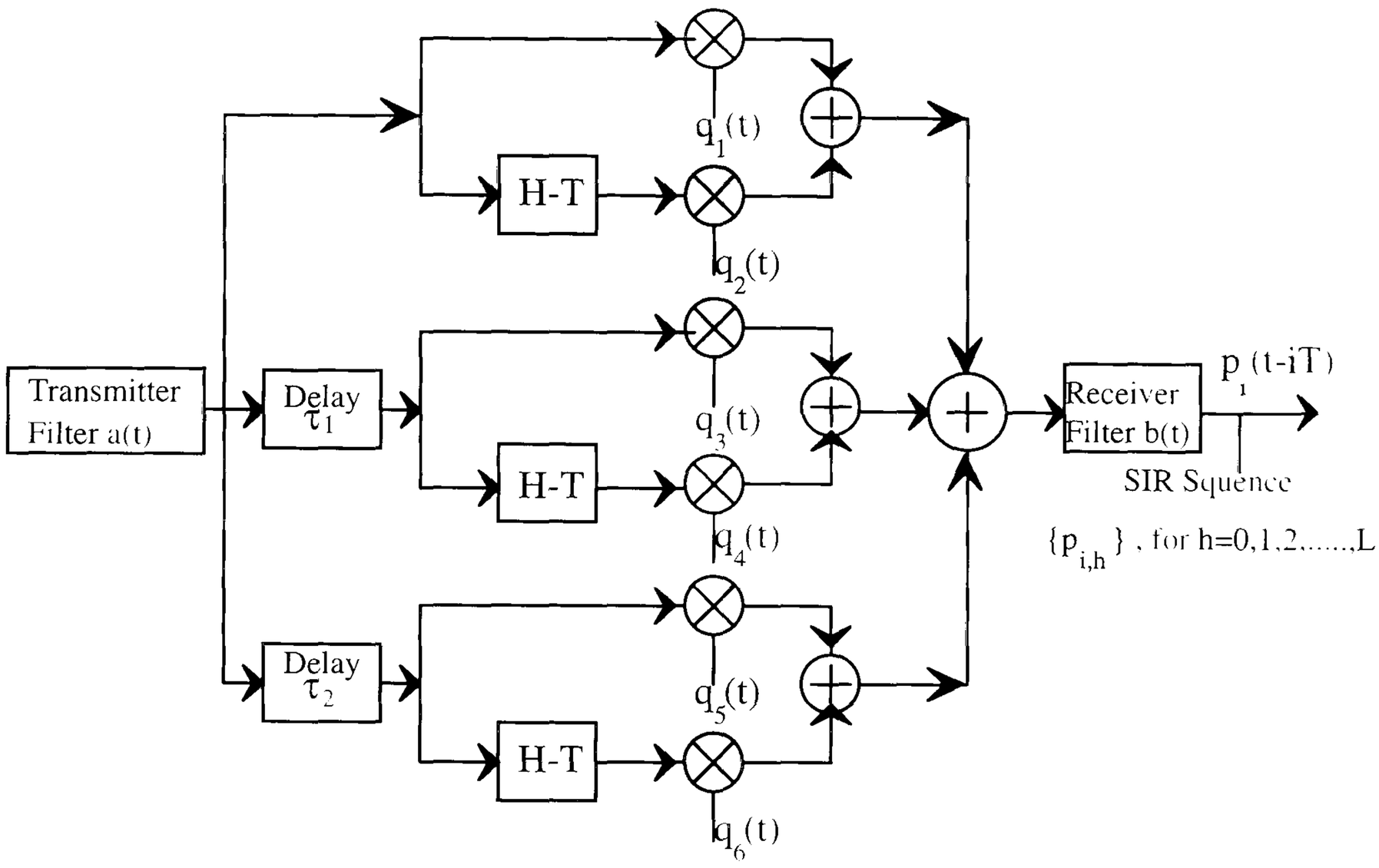


Fig.5.2.3 : Model of Linear Baseband HF Channel for Generation The SIR By Computer Simulation.

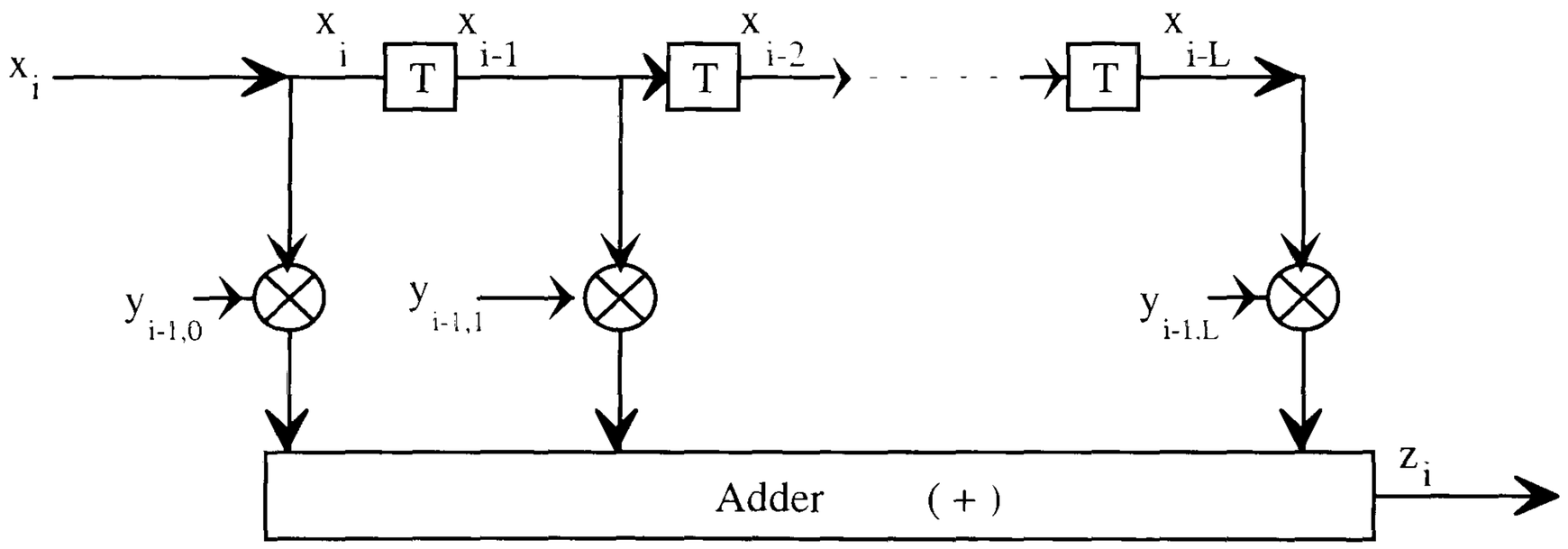


Fig.5.3.1 : Model of Generation the Signal z_i from $(x_i, x_{i-1}, \ldots\ldots x_{i-L})$.

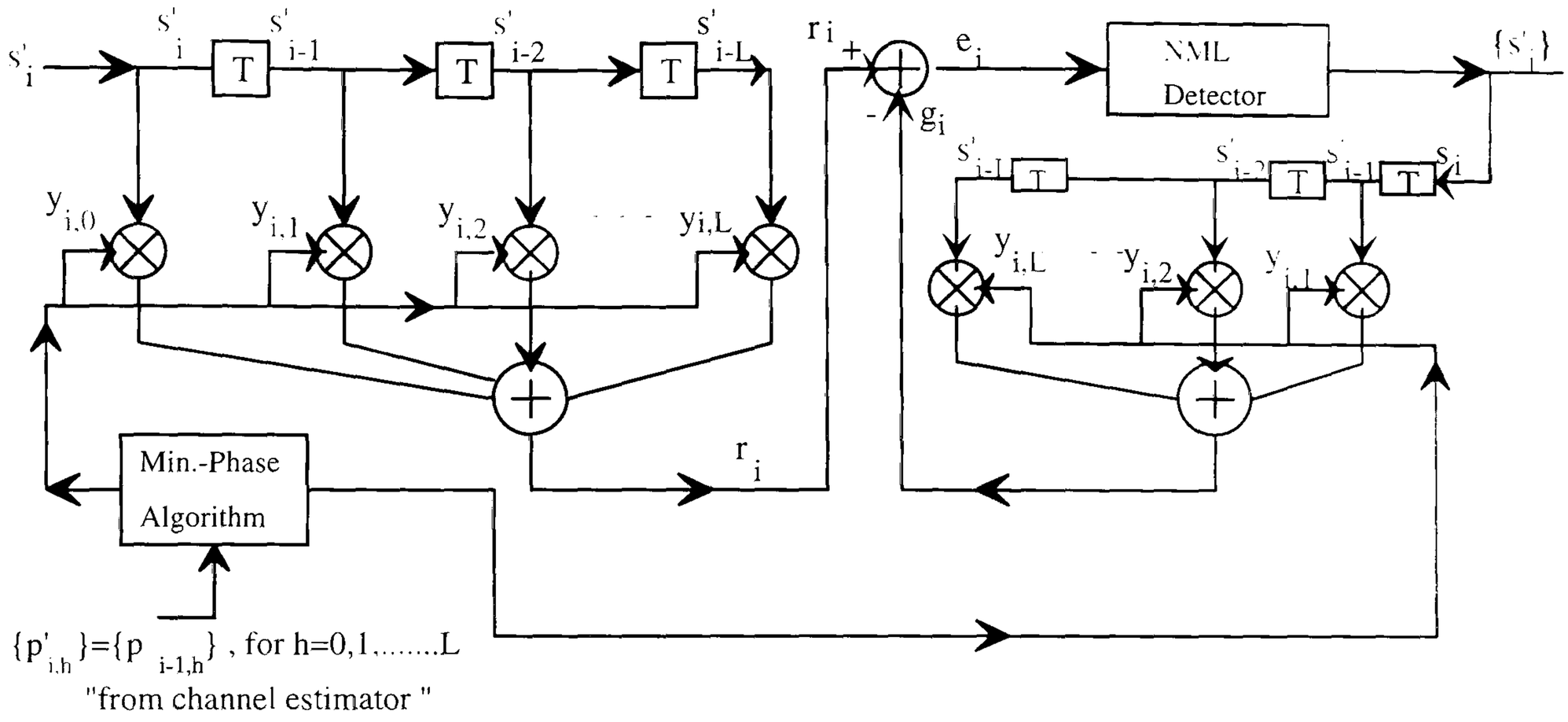
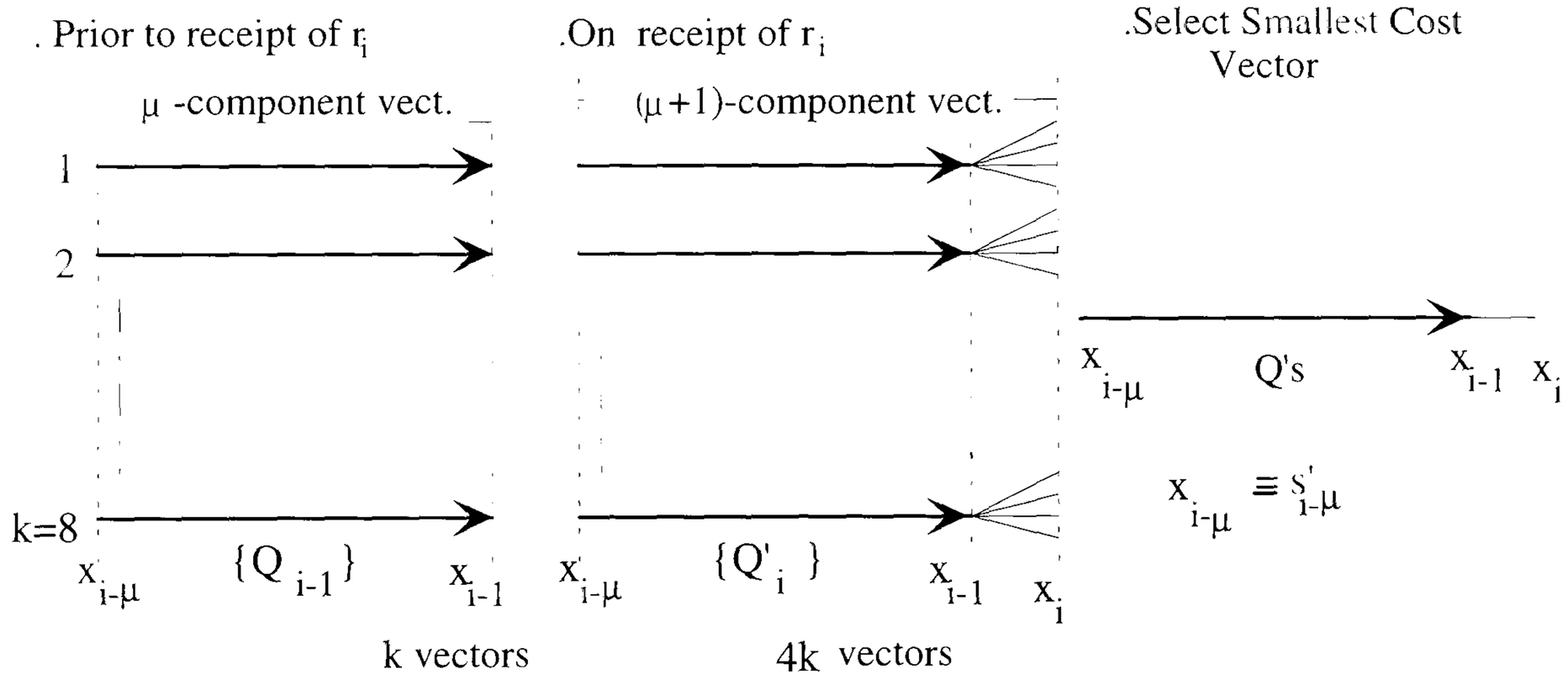
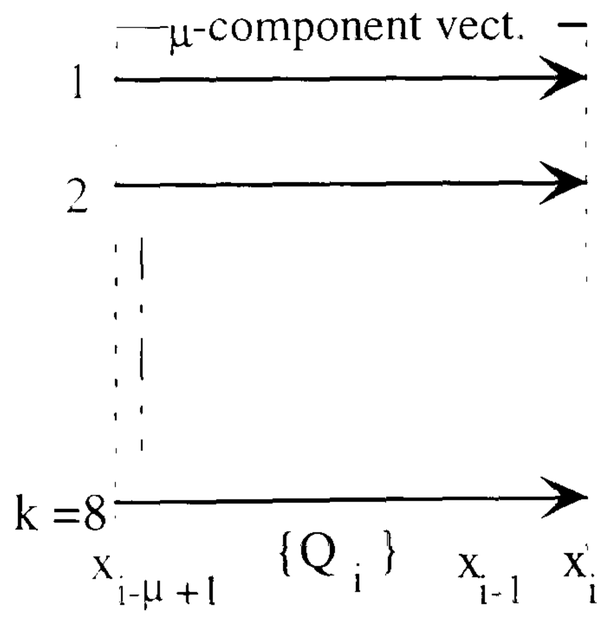


Fig.5.3.2 : An Equivalent Model of The System.

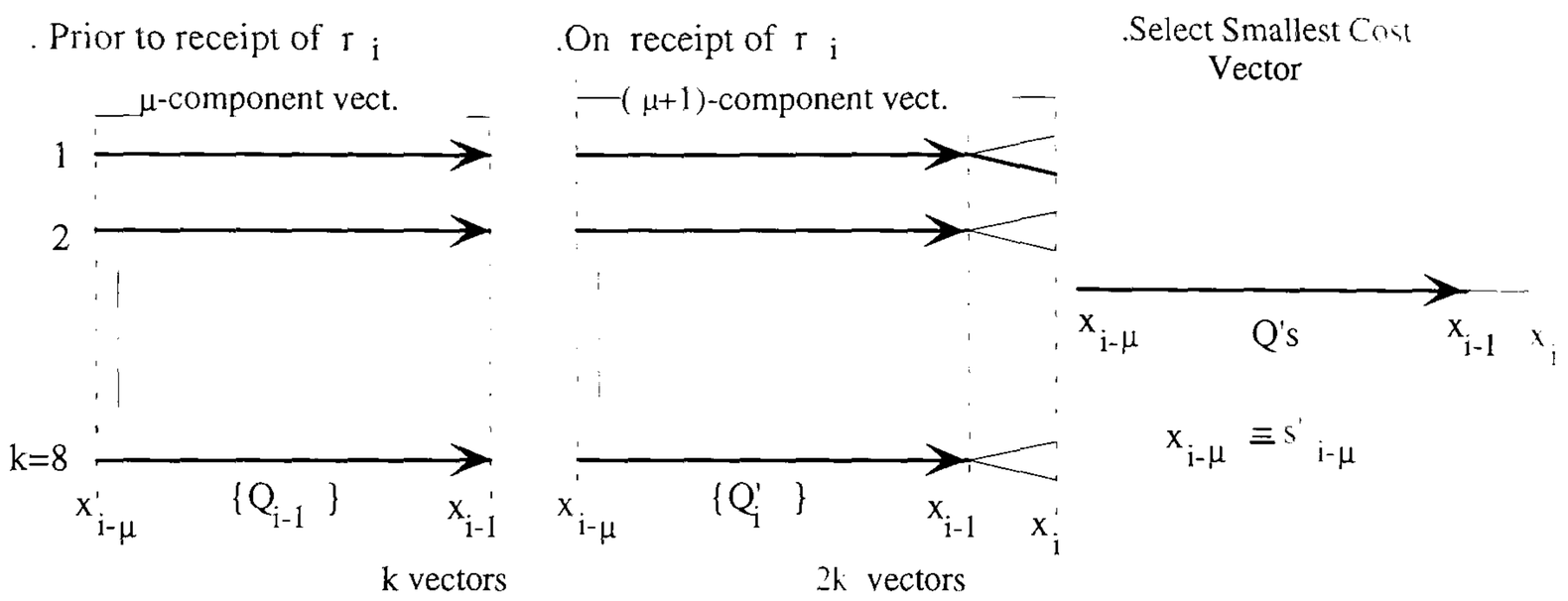


. Select k=8 vectors with smallest costs.

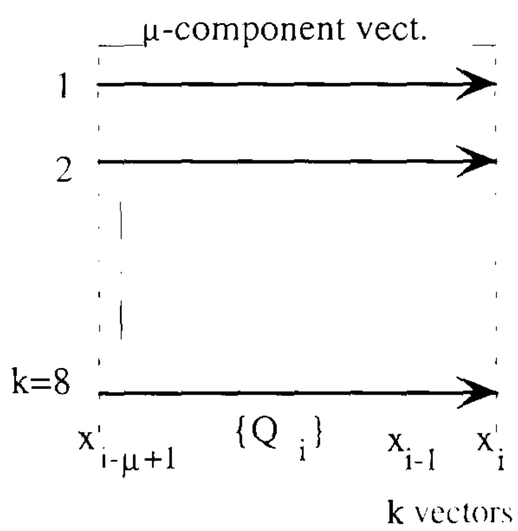


. Detector Ready for Next Detection.

Fig.5.4.1 : Operation of Detector 1 in vector representation.

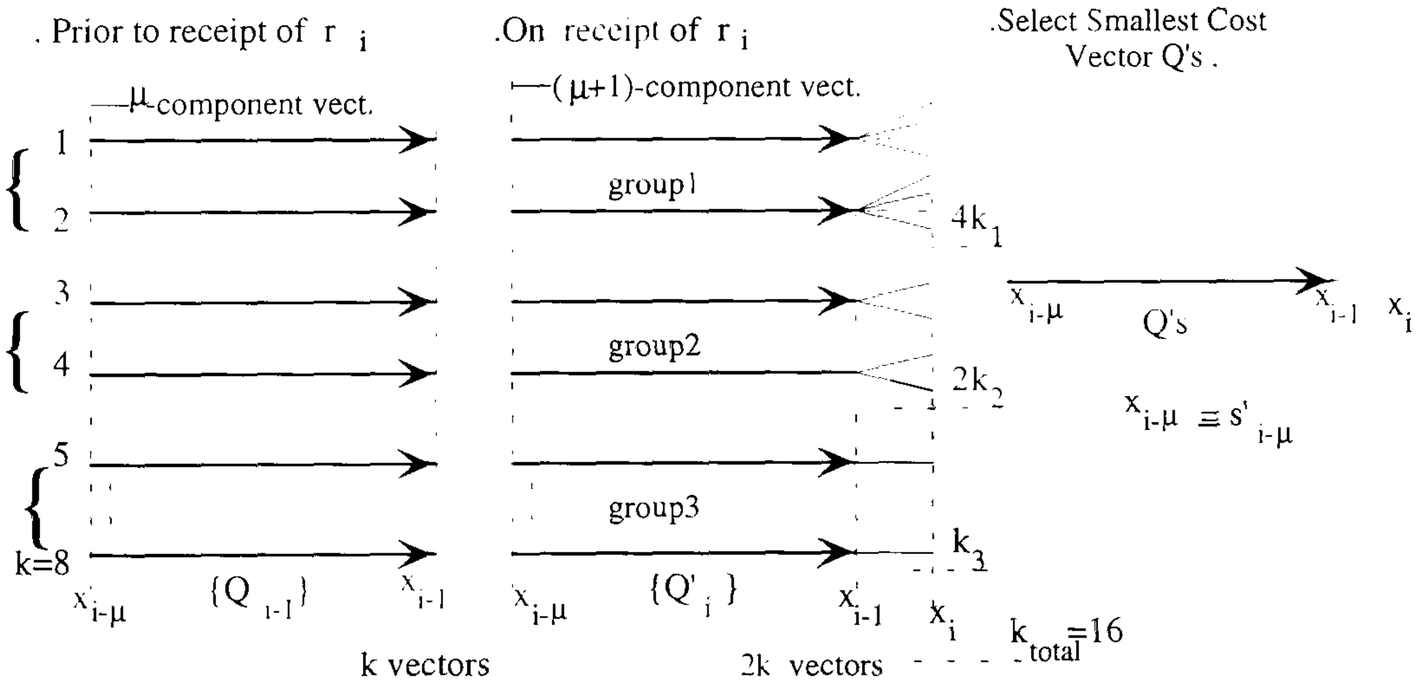


. Select k=8 vectors with smallest costs.

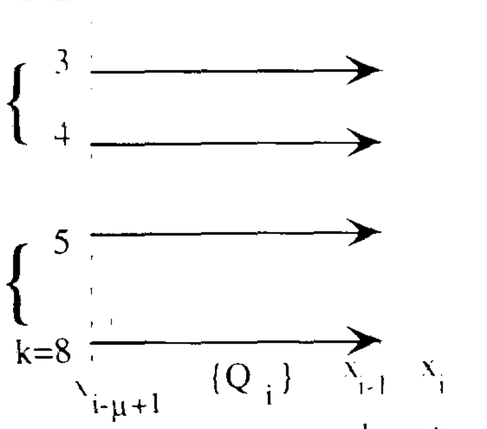


. Detector Ready for Next Detection.

Fig.5.4.2 : Operation of Detector 2 in vector representation.



. Select k=8 vectors with smallest costs.



. Detector Ready for Next Detection.

Fig.5.4.3 : Operation of Detector 3 in vector representation.

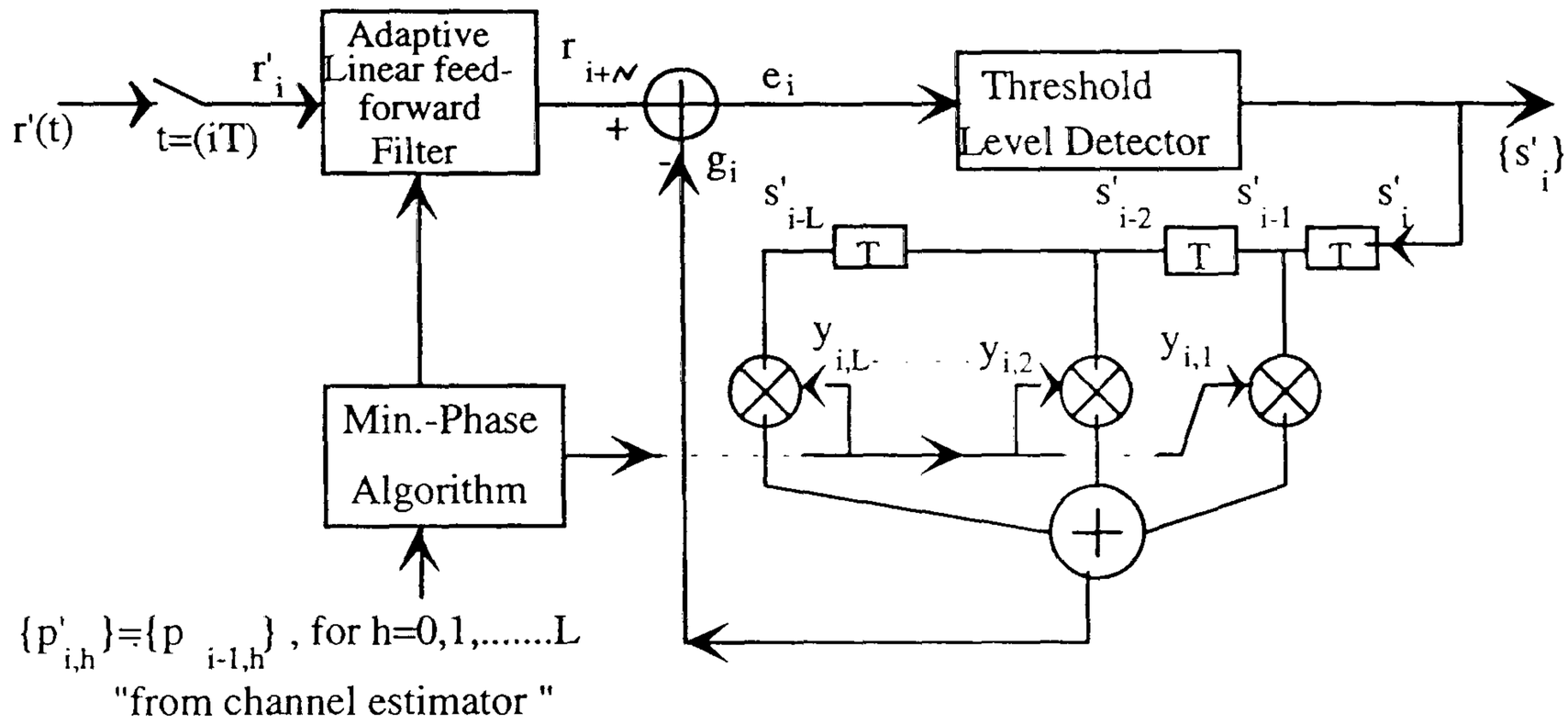


Fig.5.5.1 : Conventional Non-Linear Equaliser .

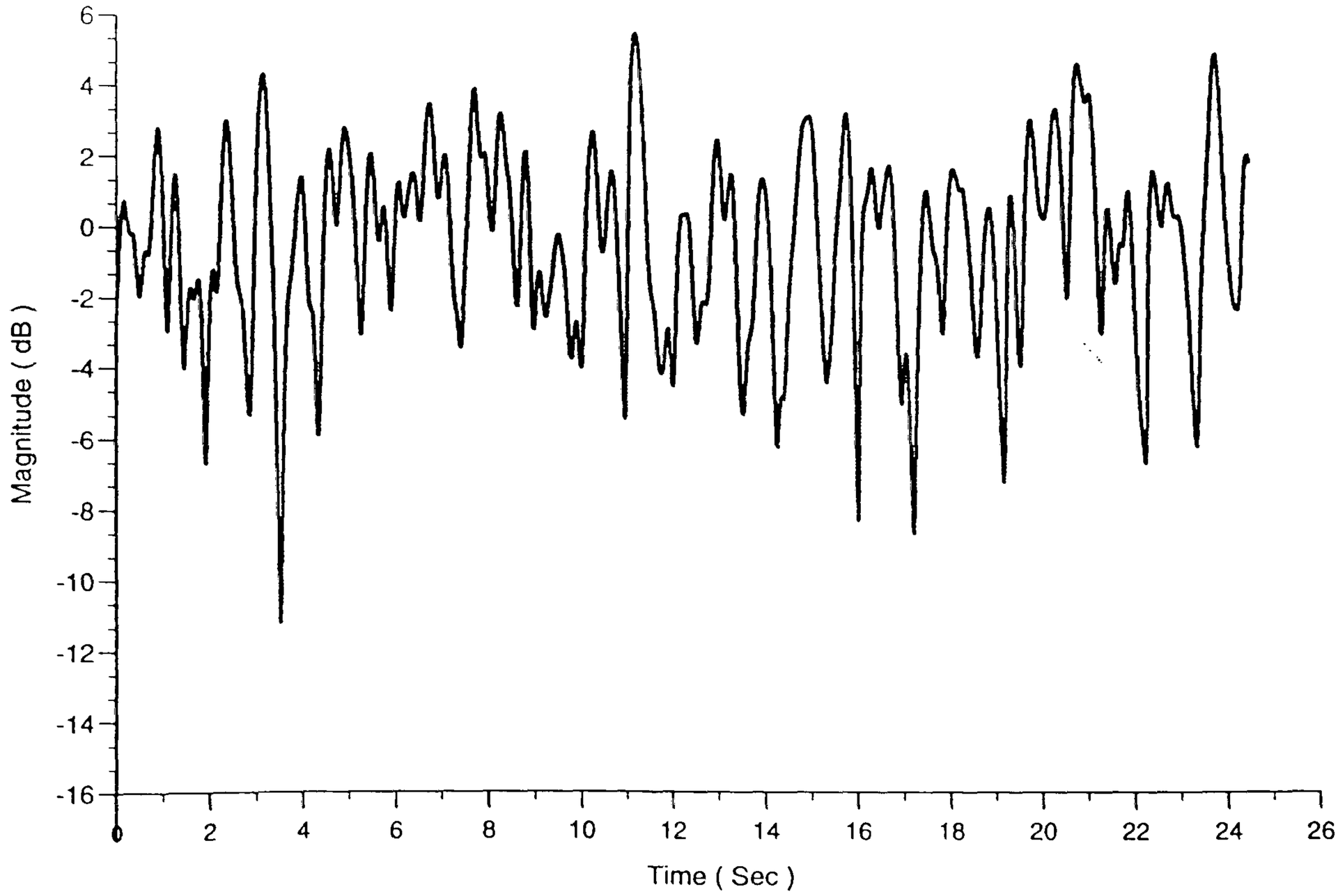


Fig. 5.6.1 Magnitude Variation of the SIR of Channel1 undet Test

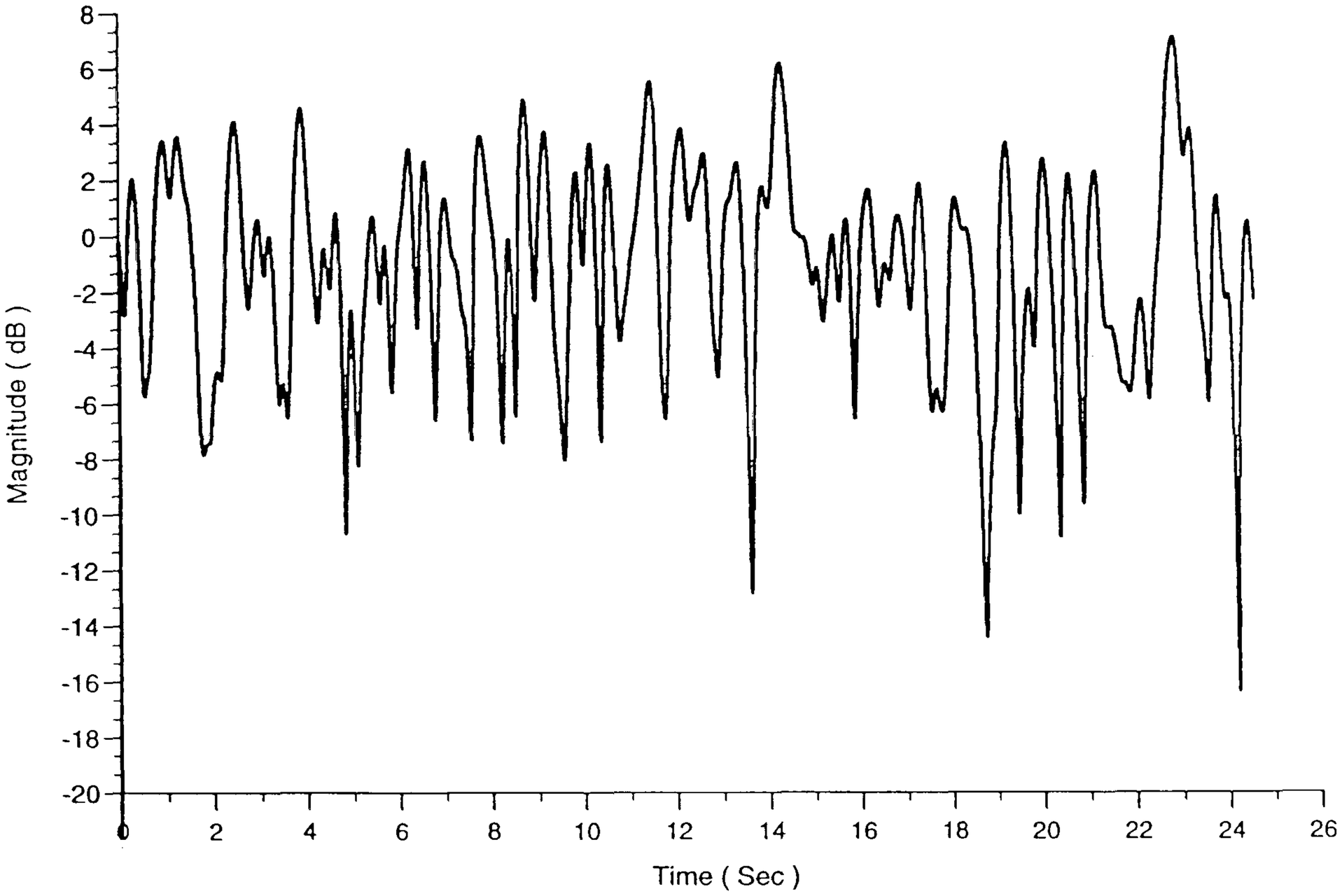


Fig. 5.6.2 Magnitude Variation of the SIR of Channel2 undet Test.

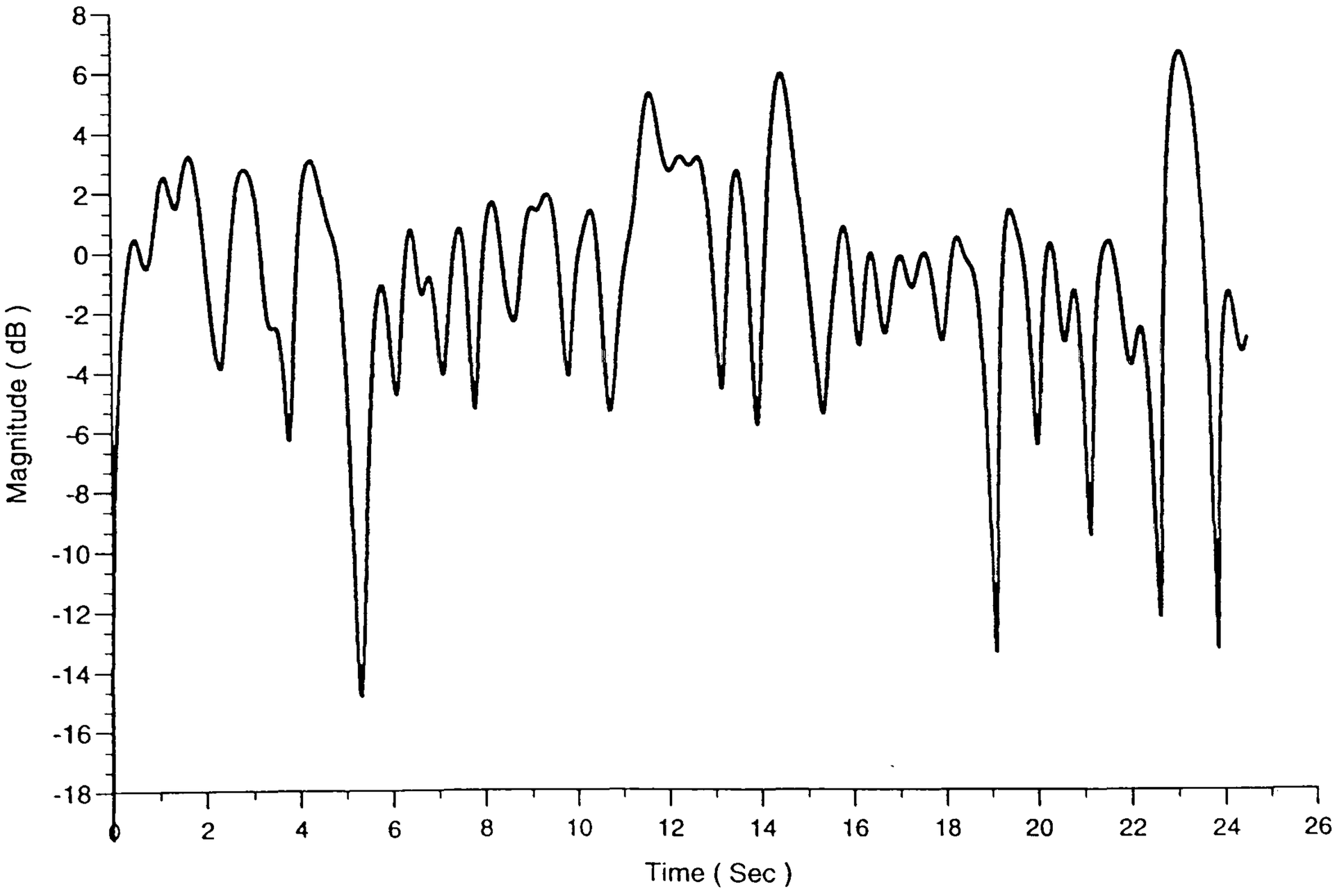


Fig. 5.6.3 Magnitude Variation of the SIR of Channel3 under Test.

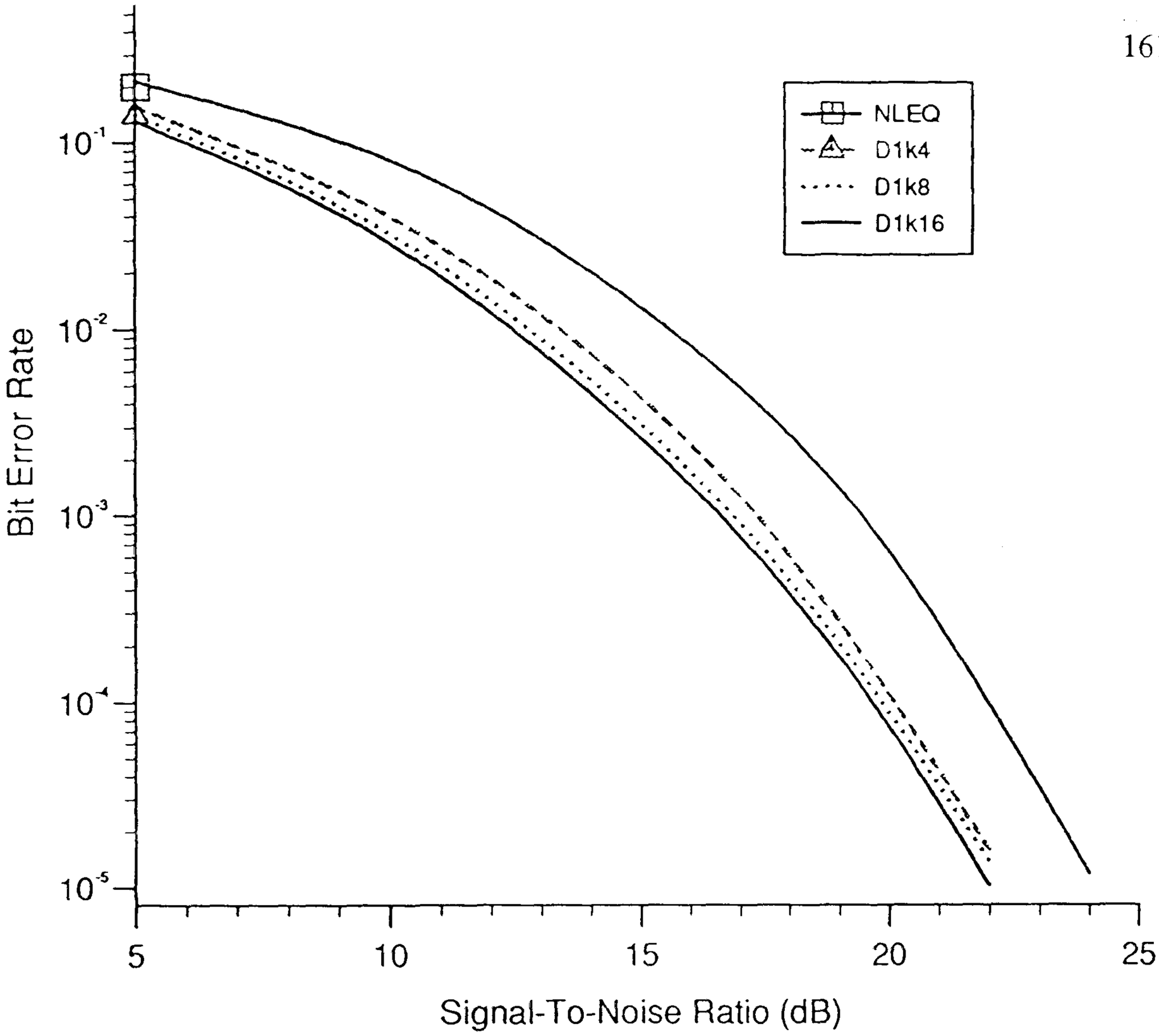


Fig. 5.6.4 Bit Error Rate Measurement of Detector1 over Channel1.

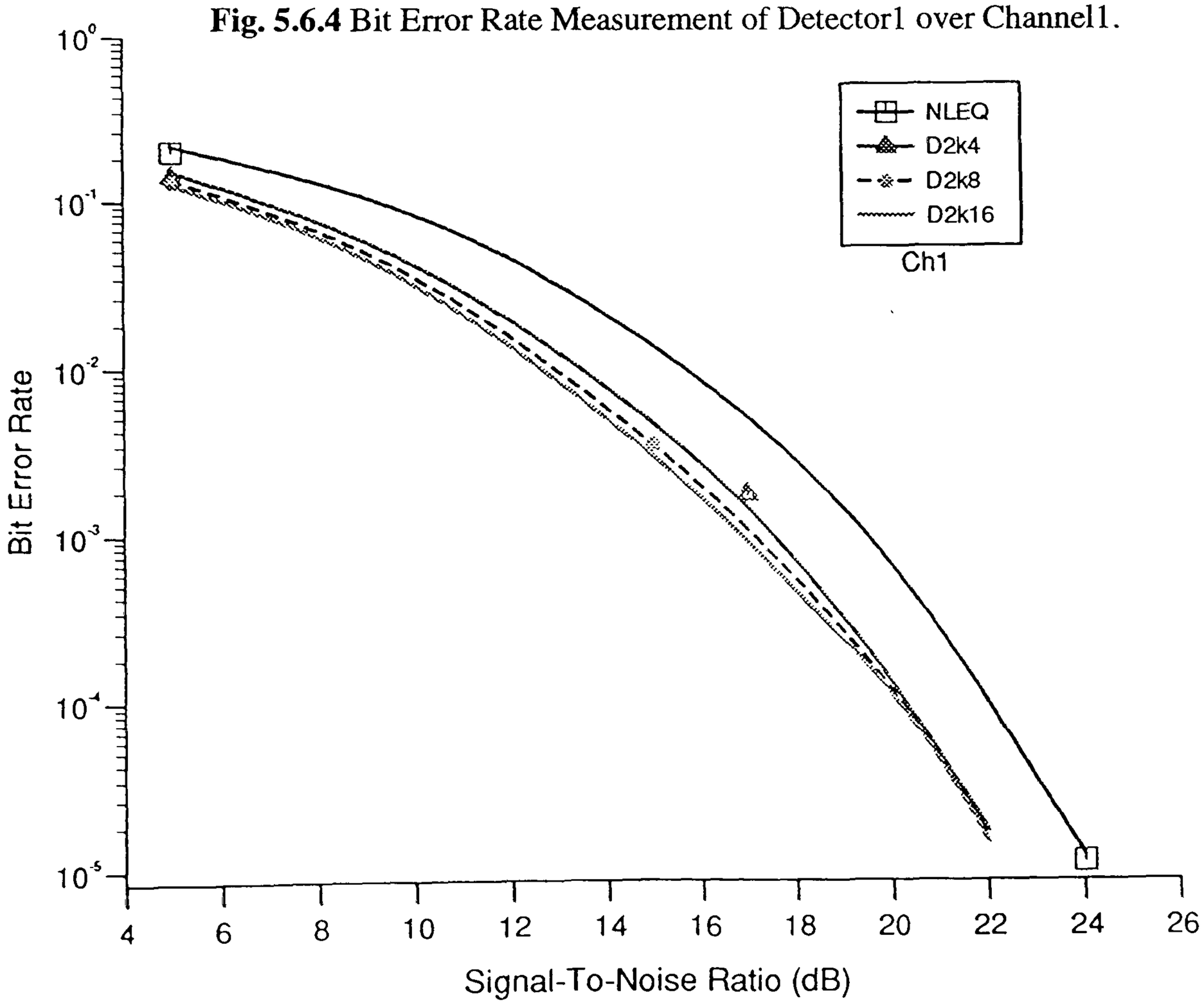


Fig. 5.6.5 Bit Error Rate Measurement of Detector2 over Channel1.

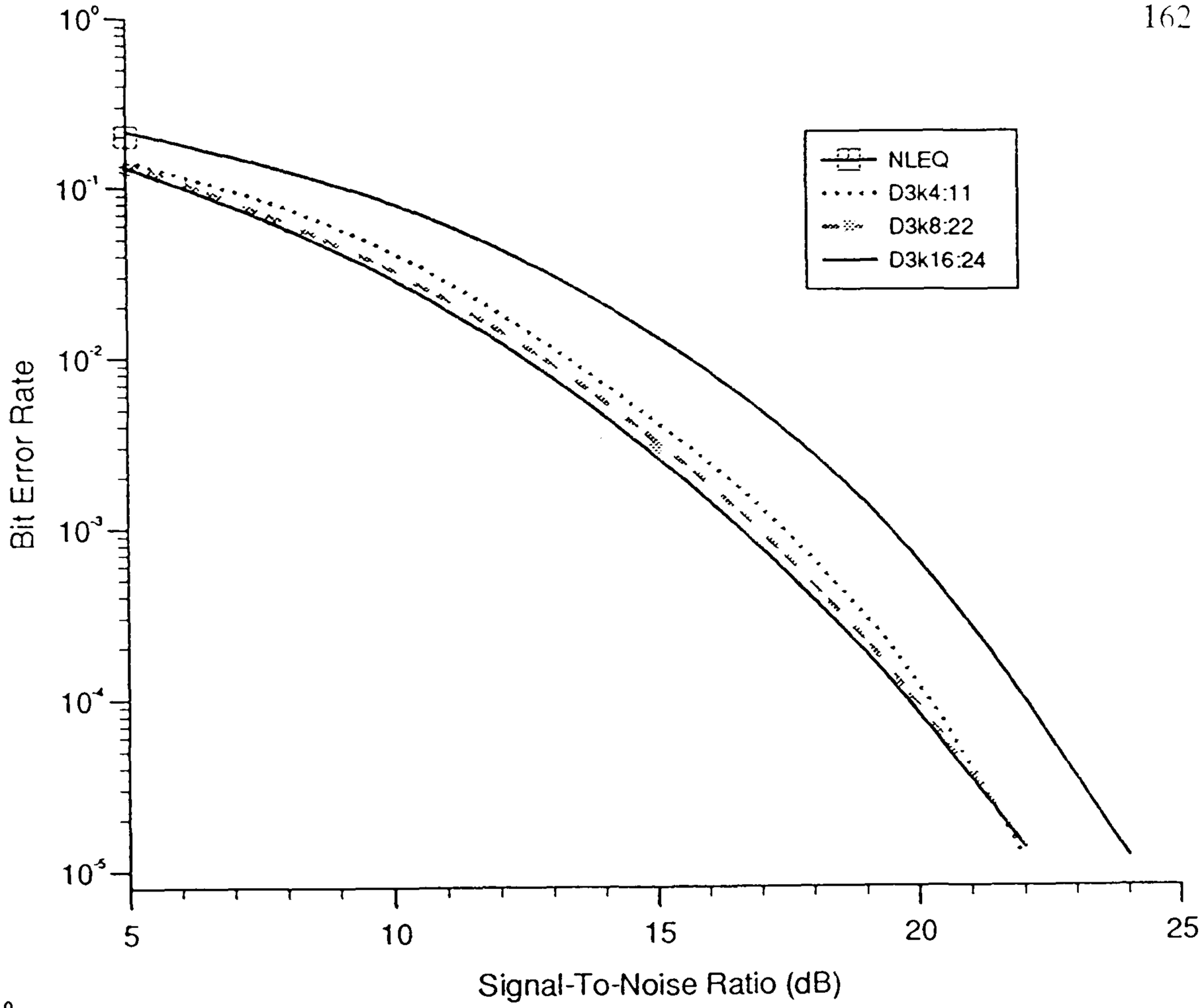


Fig. 5.6.6 Bit Error Rate Measurement of Detector3 over Channel1.

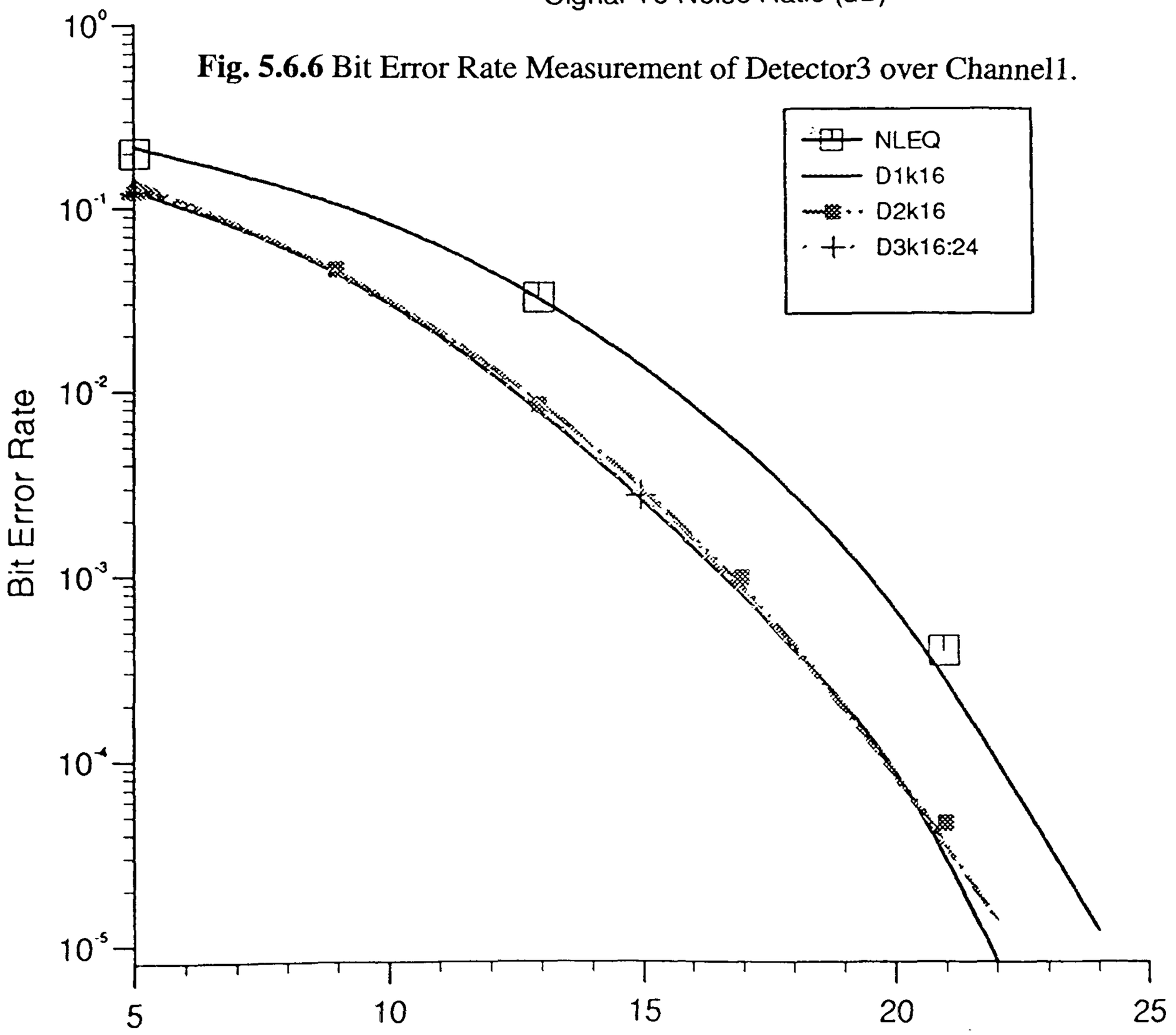


Fig. 5.6.7 Bit Error Rate Measurement of Detectors with k=16 over Channel1.

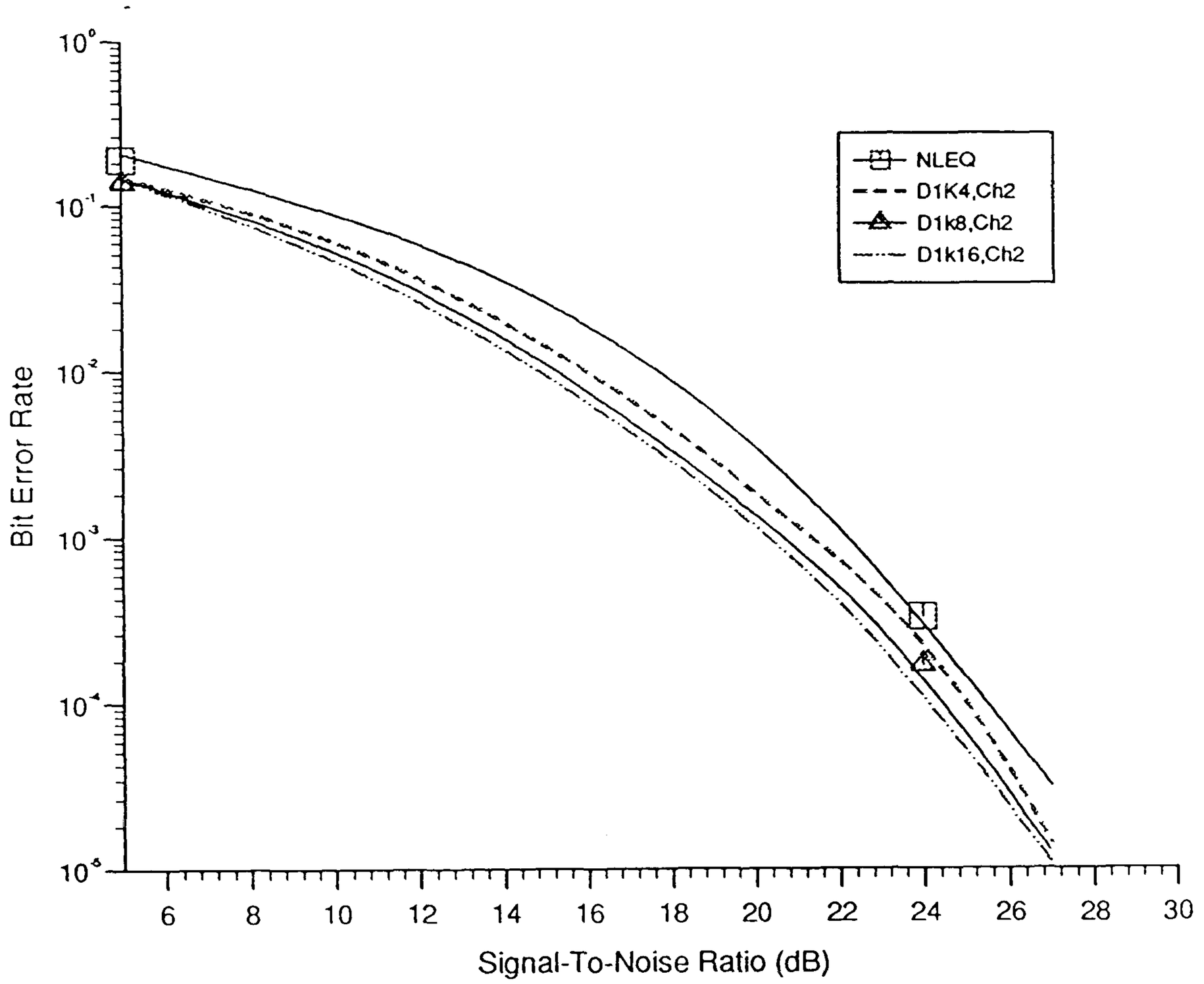


Fig. 5.6.8 Bit Error Rate Measurement of Detector1 over Channel2.

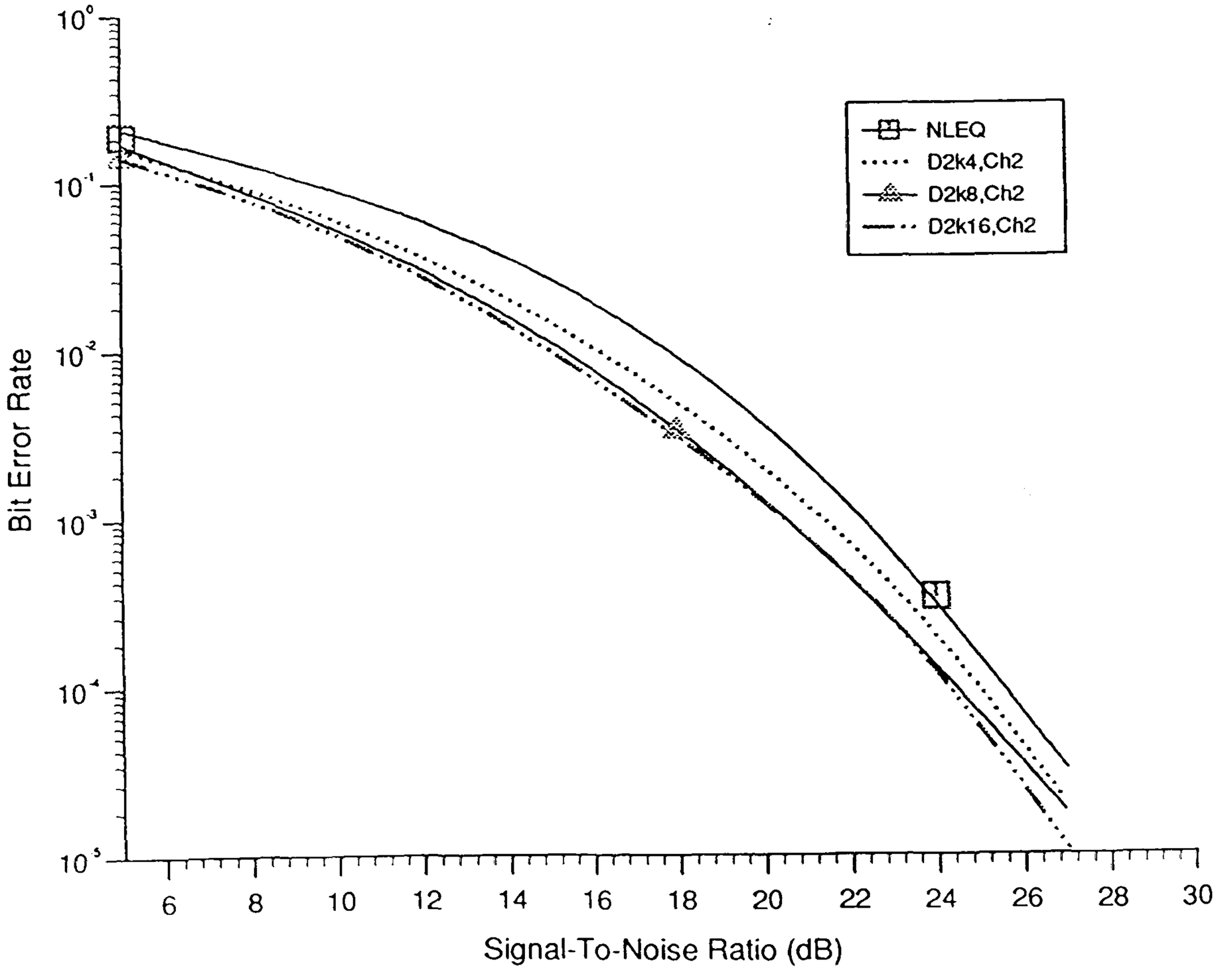


Fig. 5.6.9 Bit Error Rate Measurement of Detector2 over Channel2.

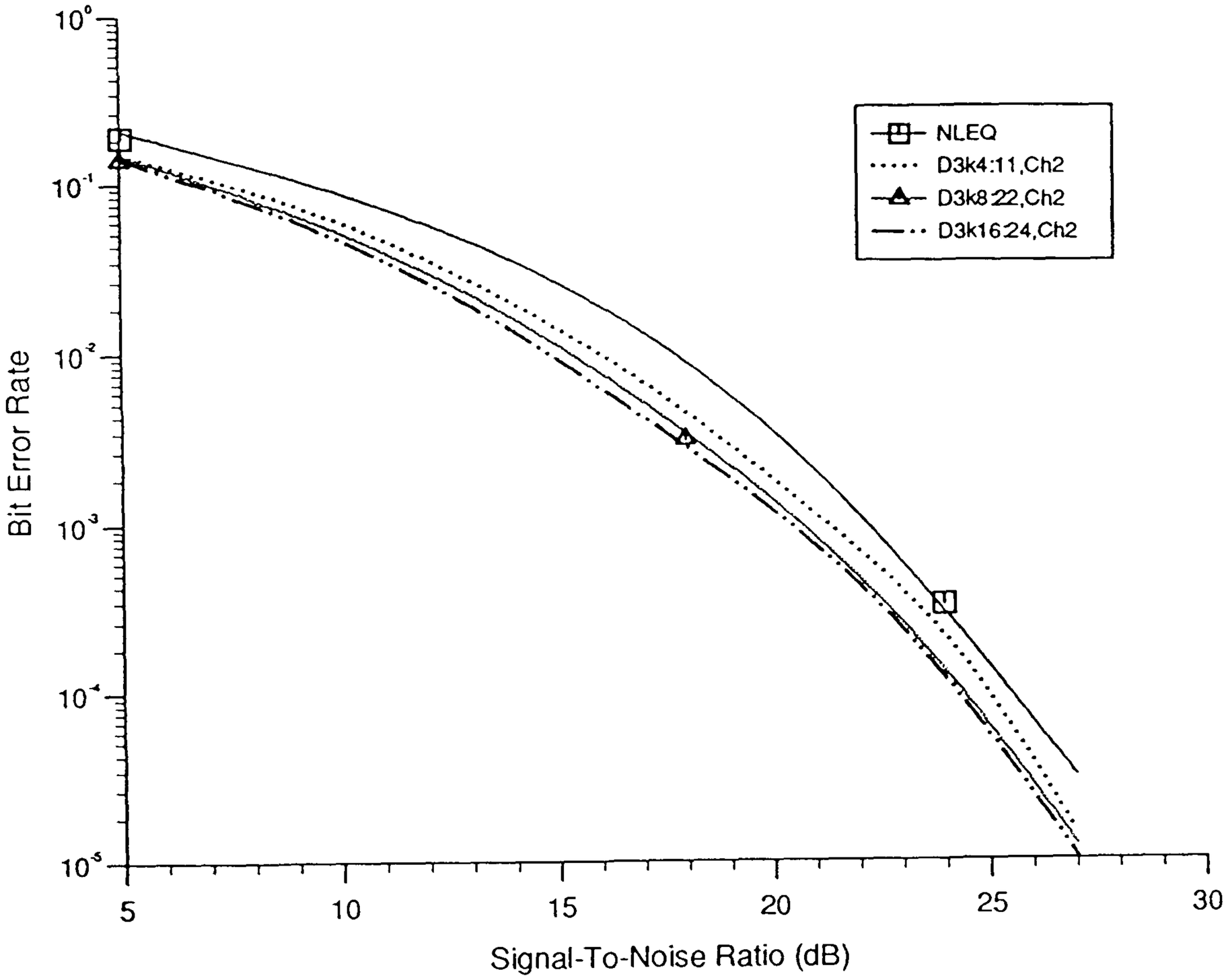


Fig. 5.6.10 Bit Error Rate Measurement of Detector3 over Channel2.

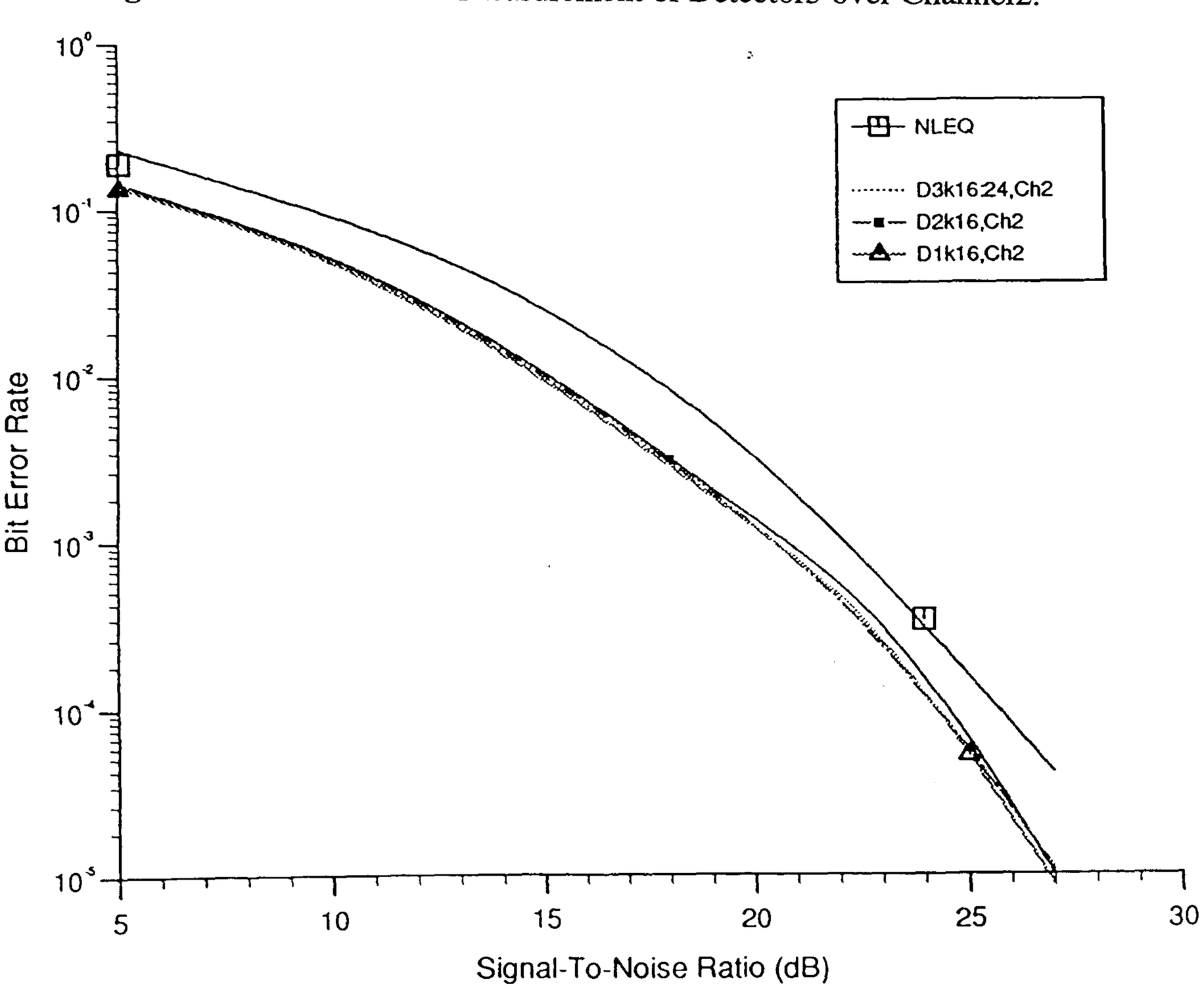


Fig. 5.6.11 Bit Error Rate Measurement of Detectors with k=16 over Channel2.

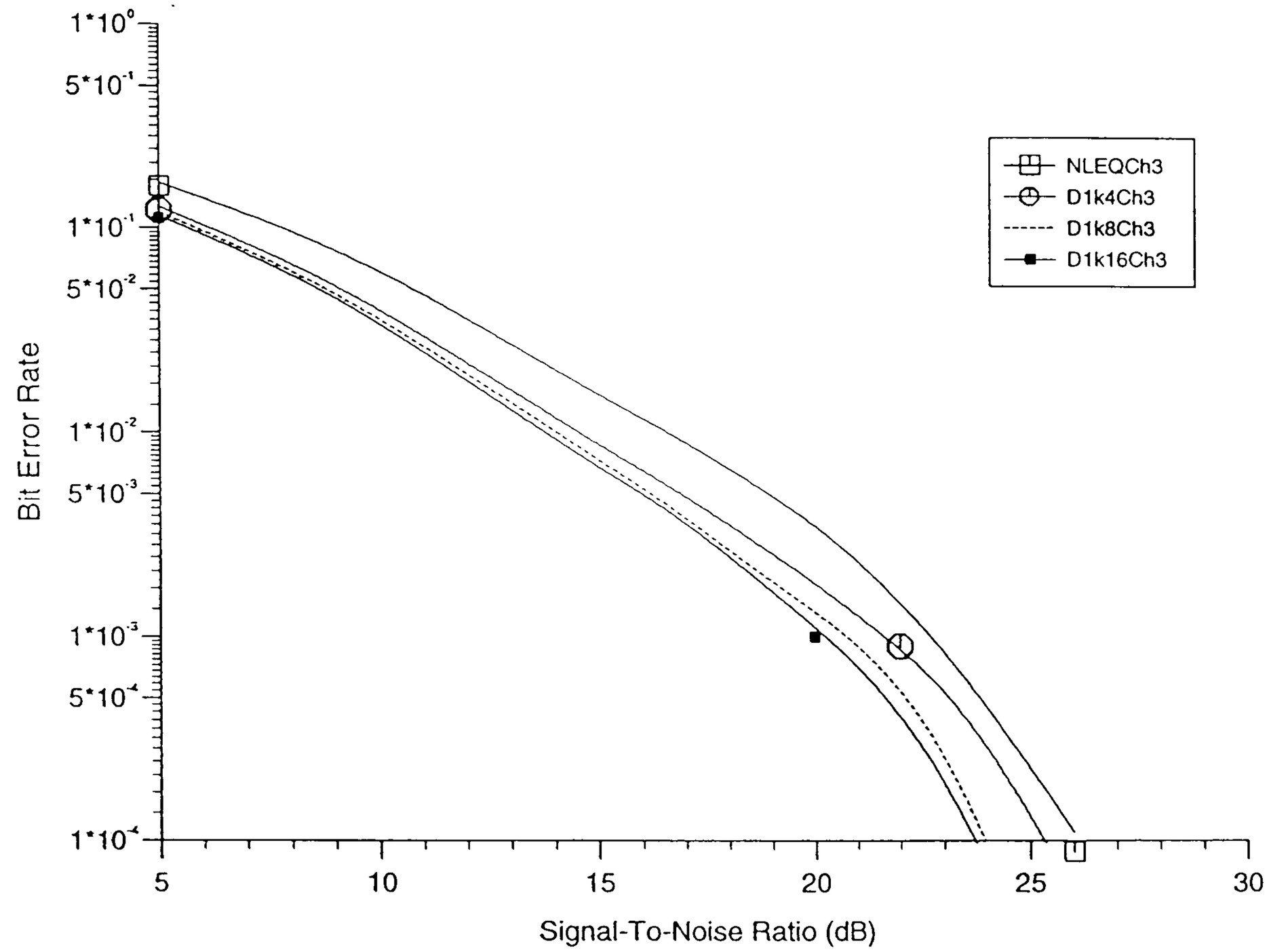


Fig. 5.6.12 Bit Error Rate Measurement of Detector1 over Channel3.

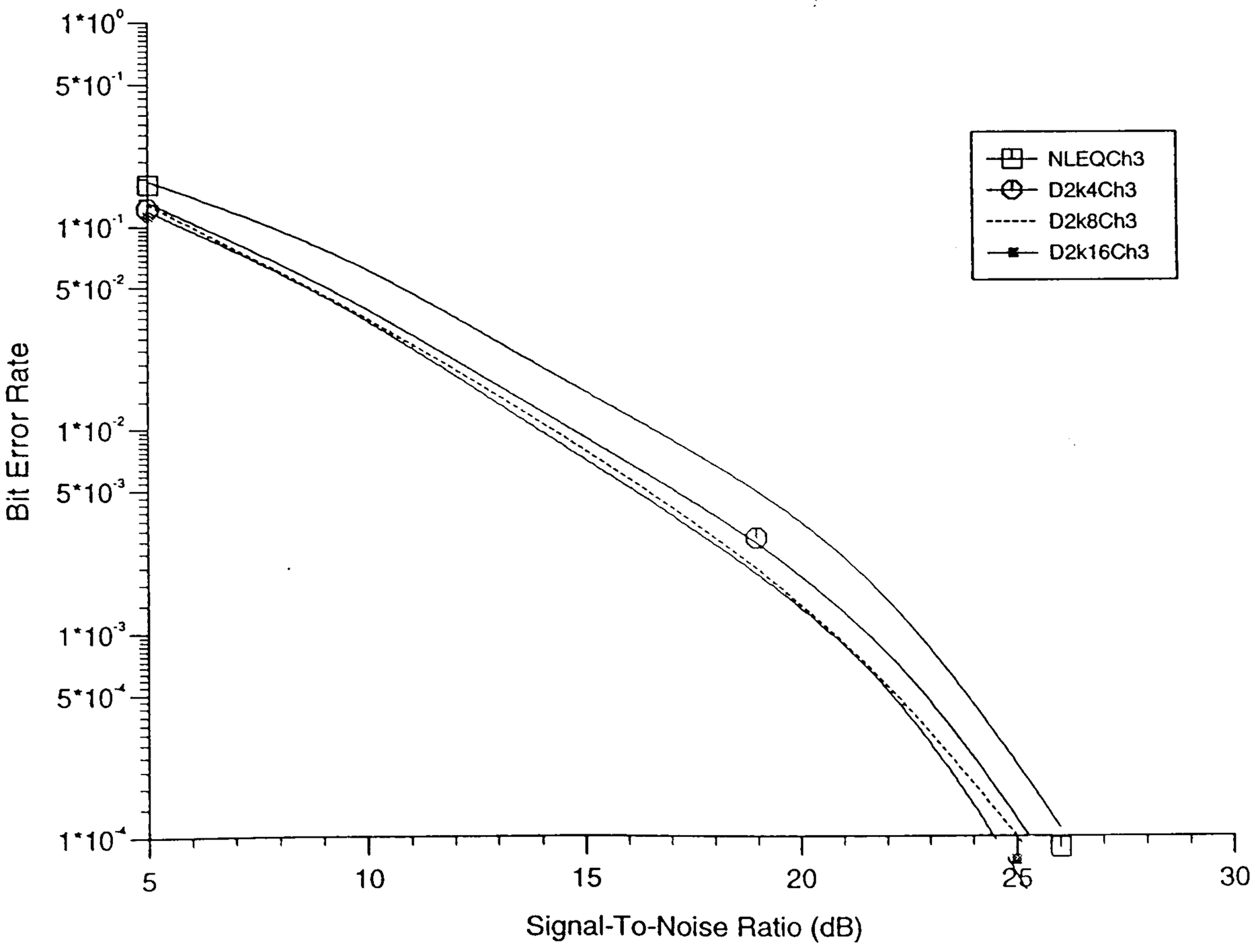


Fig. 5.6.13 Bit Error Rate Measurement of Detector2 over Channel3.

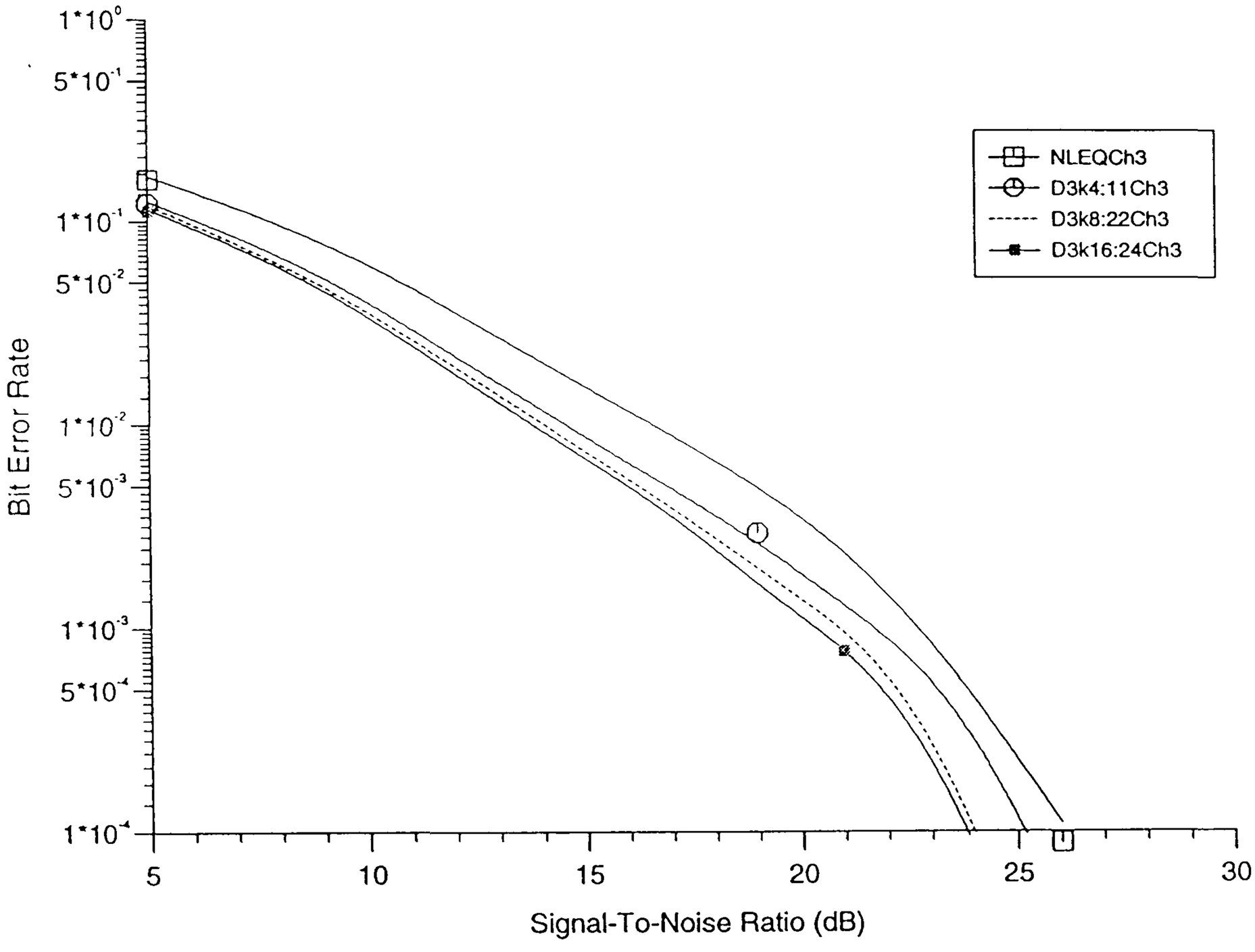


Fig. 5.6.14 Bit Error Rate Measurement of Detector3 over Channel3.

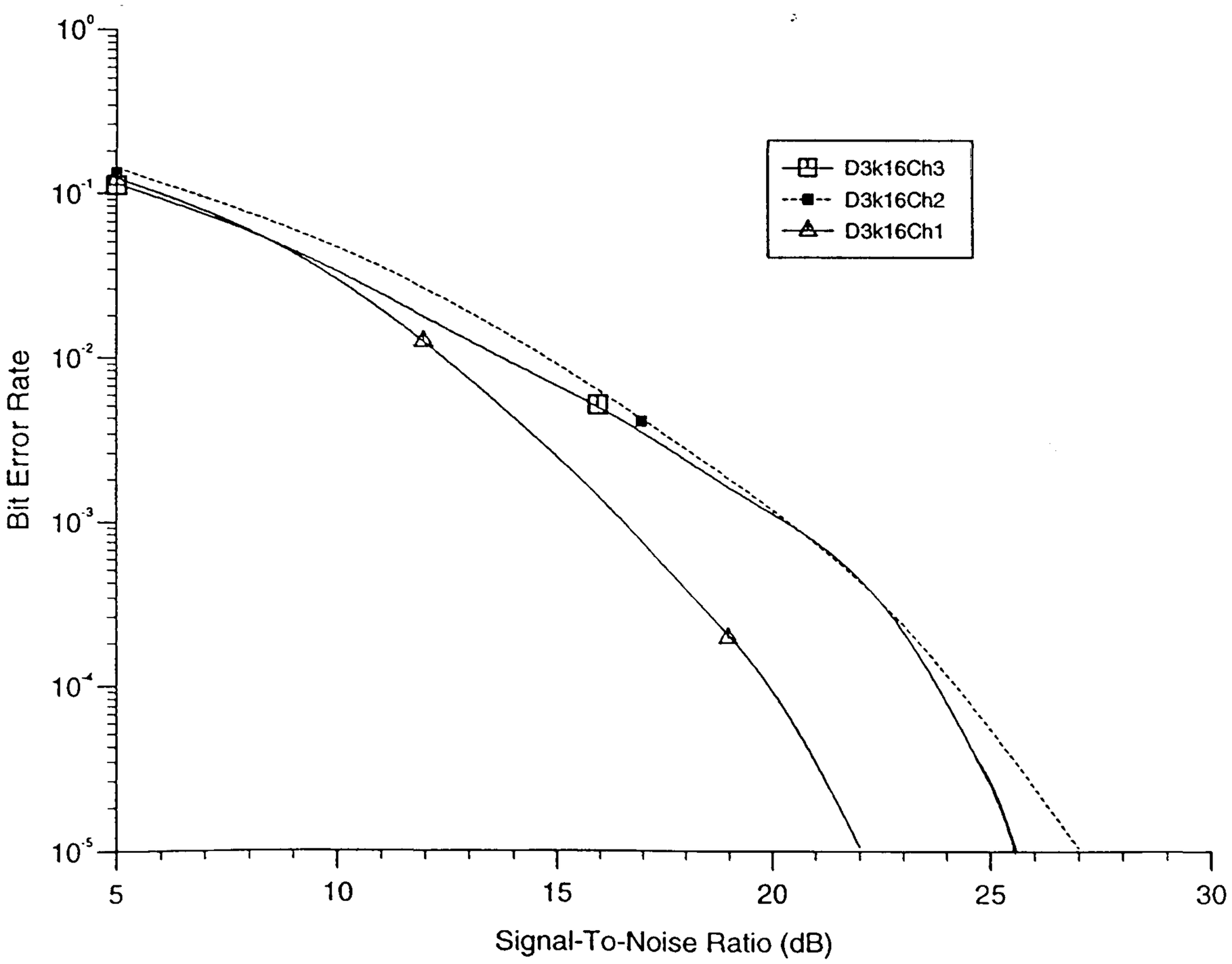


Fig. 5.6.15 Bit Error Rate Measurement of Detector3 over three Channels.

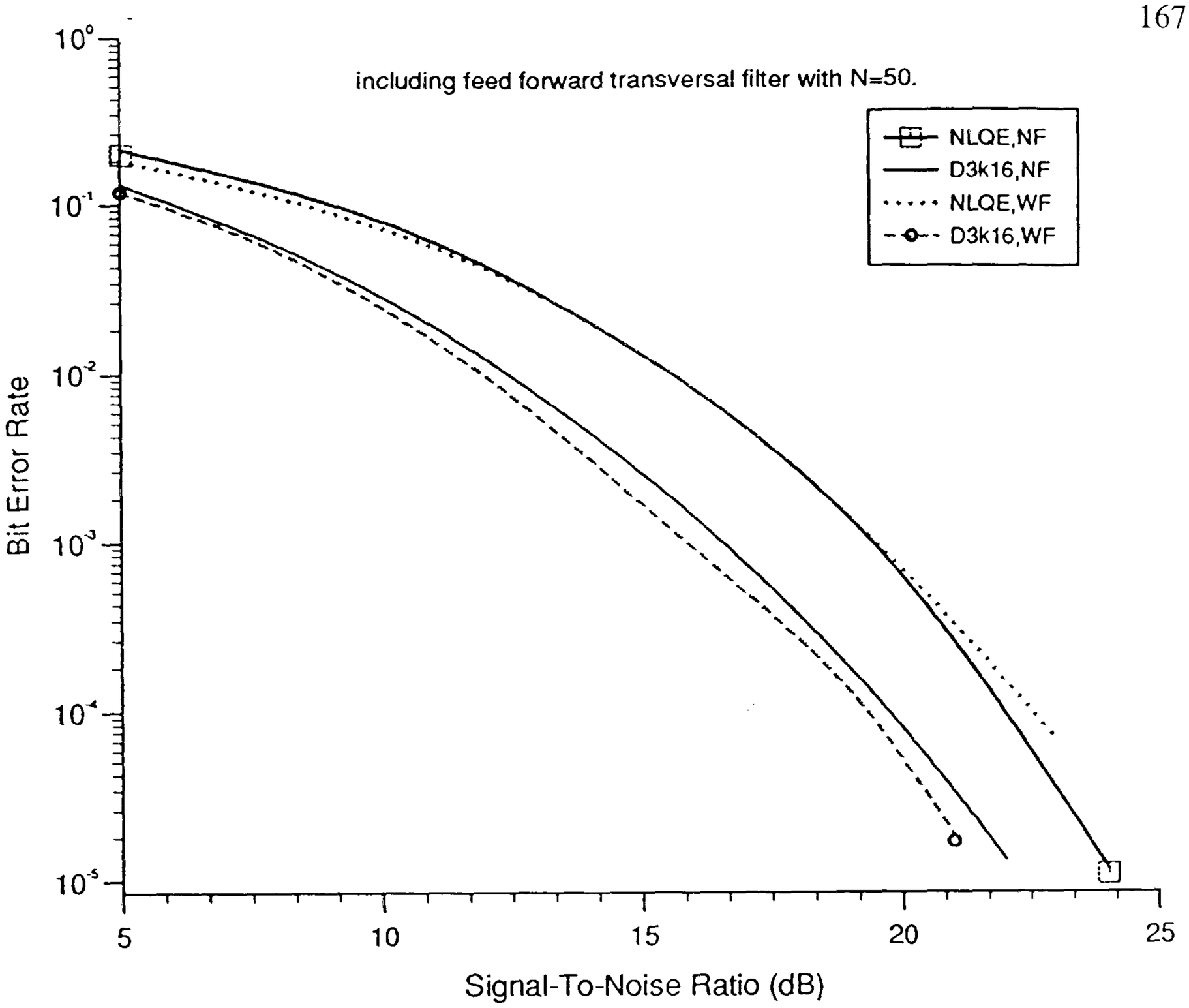


Fig. 5.6.16 Performance of Detector3 and Non-linear Equaliser over Channel1.

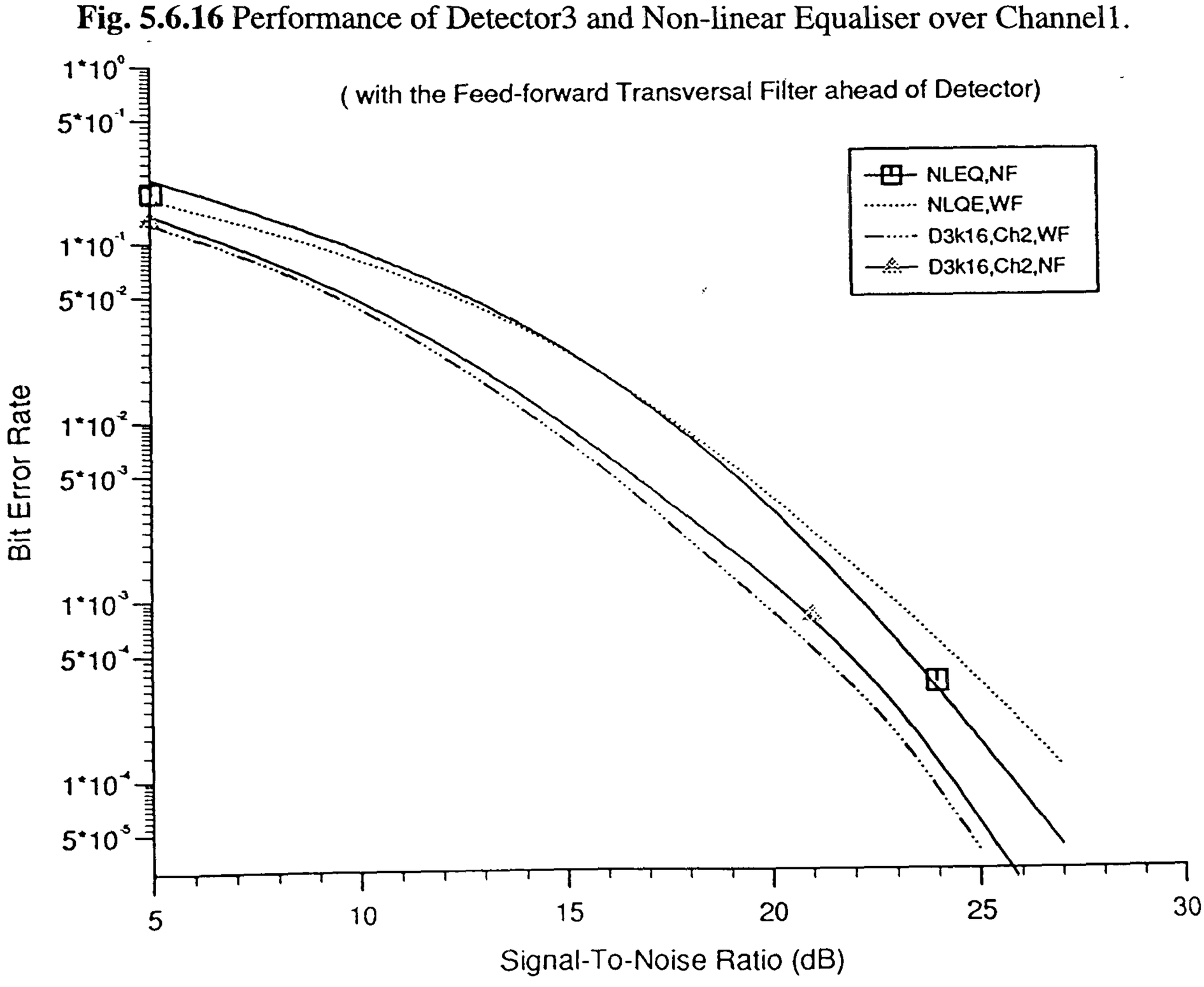


Fig. 5.6.17 Performance of Detector3 and Non-linear Equaliser over Channel2.

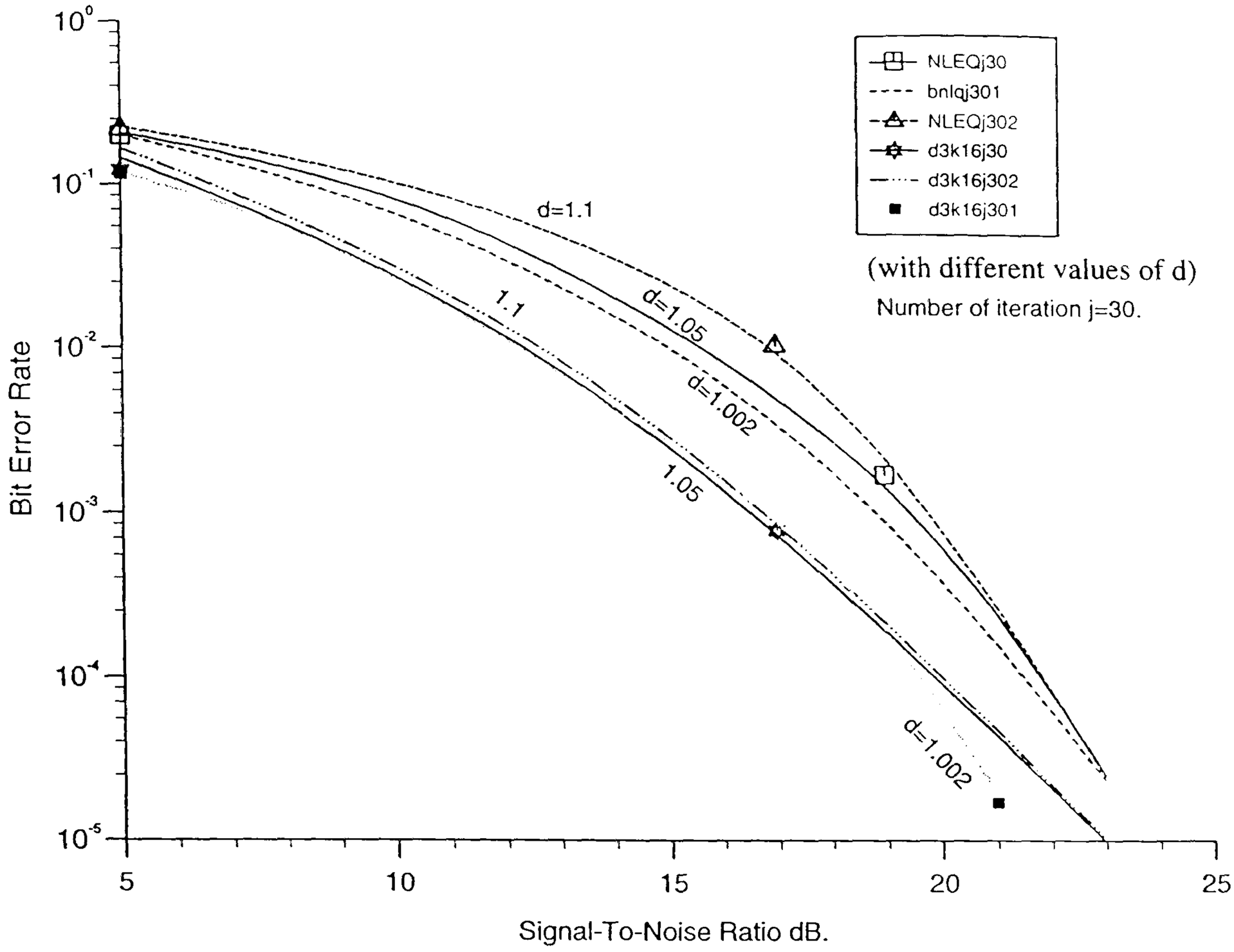


Fig. 5.6.18 Performance of Detector3 and Non-linear Equaliser over Channel1.

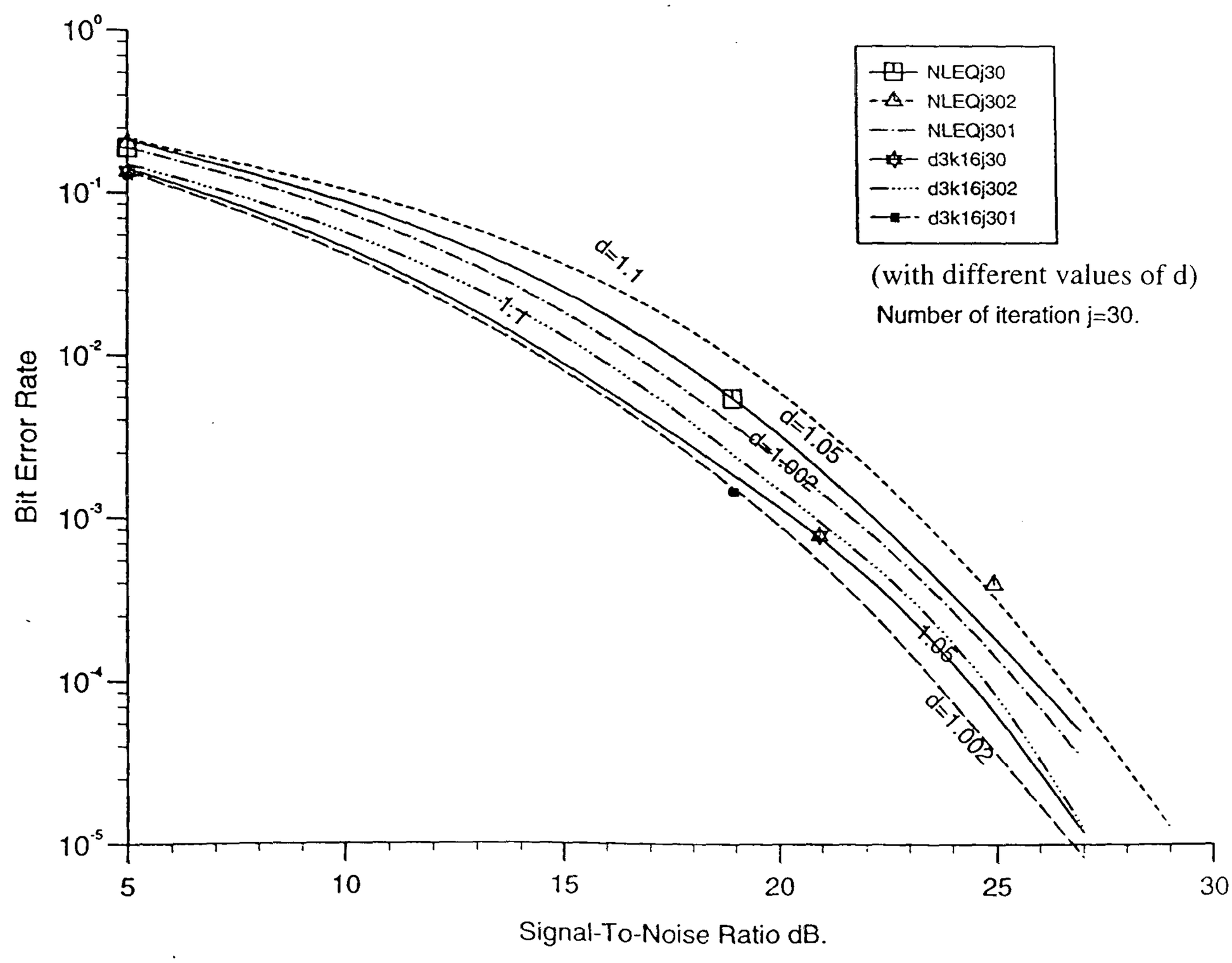


Fig. 5.6.19 Performance of Detector3 and Non-linear Equaliser over Channel2.

CHAPTER 6

CHANNEL ESTIMATION TECHNIQUES FOR HF RADIO LINKS

6.1 INTRODUCTION

In the adaptive adjustment of the NML detector, the receiver must continuously estimate the sampled impulse-response (SIR) of the channel and appropriately adjust (update) the stored estimate SIR, used by the NML detector or by the minimum phase algorithm. The required estimate may be obtained in a variety of different ways [27,28,61,62,65,67,73,76,92,78]. Since the estimation process is data-aided, using therefore the detected data symbols, a reasonably satisfactory estimate of the channel can be obtained by means of the conventional gradient algorithm, which is likely to form the basis of the most cost effective estimator, for a time-varying channel or where the receiver has only a limited knowledge of the correct model of the channel. The conventional gradient algorithm is in fact, a recursive solution to the least squares estimation problem, also termed as the least mean-square (LMS) error algorithm [1,61,62,65]. Furthermore, by suitably modifying the LMS algorithm itself, a much more accurate estimate of the HF channel can be obtained, by using an appropriate prediction technique [60,62,64].

Another approach is to use a Kalman filter as an estimator or equaliser for an unknown channel [1,23,27,28,48,63,67,69-74,78,80-84,88]. It is well known that, when appropriately designed, a Kalman estimator gives the least-squares estimate of the sampled impulse response of a time-invariant channel [63,64,74,97]. Unfortunately, this is considerably more complex than the corresponding gradient algorithm. Considerable advances have recently been achieved, in order to reduce the complexity of the Kalman algorithm [1,48,63,69,98]. The estimator technique can be divided into two separate classes, named the LMS and RLS algorithm. The choice of one algorithm over another is determined by various factors :

1.) Rate of convergence: This is defined as the numbers of iterations required for the algorithm, in response to stationary inputs, to converge "close enough " to the optimum value in the mean square sense. A fast rate of convergence allows the algorithm to adapt rapidly to a stationary environment of the unknown statistics.

2.) Mis-adjustment : This parameter provides a quantitative measure of the amount by which the final value of the mean-square error deviates from the minimum mean-squared error that is produced by the Wiener system [63].

3.) Tracking : when an adaptive algorithm operates in a non-stationary environment, the algorithm is required to track statistical variations in the environment. The tracking ability of the algorithm, however, is influenced by two contradictory features : (a) rate of convergence, and (b) steady-state fluctuation due to algorithm noise.

4.) Robustness : This refers to the ability of the algorithm to operate satisfactorily with poorly-conditioned input data.

5.) Computational requirements : This includes : (a) the number of operations, i.e. (multiplictions, division, and additions/subtractions) required to make one complete iteration of the algorithm, (b) the size of memory locations required to store the data and the program, and (c) the investment required to program the algorithm on a computer.

6.) Structure : This refers to the structure of information flow in the algorithm, determining the manner in which it is implemented in hardware form. For example an algorithm whose structure exhibits high modularity, parallelism, or concurrency is well suited for implementation using vary large scale integration (VLSI) technology.

7.) Numerical Properties : In numerical implementation of the algorithm, inaccuracies are produced due to quantization errors. The quantization errors are due to analog-to-digital conversion of the input data and digital representation of internal calculations. In particular, there are two basic issues of concern, numerical stability and numerical accuracy. Numerical stability : is an inherent characteristic of an adaptive estimation algorithm. Numerical accuracy, on the other hand, is determined by the number of bits used in the numerical representation of data samples and estimator coefficients (SIR components). An adaptive algorithm is said to be numerically robust where it is insensitive to variations in the word length used in its digital implementation.

A range of different channel estimators will next be studied, starting with the simple conventional gradient estimators and finishing with sophisticated estimators. This chapter describes three kind of estimation techniques, the gradient estimator with degree-1 prediction which is the simple estimator described in [25,27,30,60-62,65], the adaptive LMS estimator and the conventional square root kalman estimator.

Results of the computer simulation tests over a model of a data transmission system are given at the end of this chapter, with the suggestion of the most suitable estimator for the combined system (adaptive filter-NML detector-estimator) .

6.2 MODEL OF SYSTEM USED IN THE TESTS

Fig.6.2.1 shows the model of the system used in these tests. The system assumed here is the same as that described in section 3-2. The received signal $r(t)$ is sampled, once per received signal element at the time instants $\{iT\}$, to give the samples $\{r_i\}$, which are complex-valued and such that :

$$r_i = \sum_{h=0}^{L} s_{i-h} \cdot p_{i,h} + v_i = P_i \cdot S_i^T + V_i \qquad 6.2.1$$

Where r_i is sample value of the complex-valued resultant base-band signal $r(t)$ at time $t=iT$, and :

$$S_i = [s_i, s_{i-1}, s_{i-2} \ldots\ldots\ldots\ldots, s_{i-L}] \qquad 6.2.2$$

$$P_i = [p_{i,0}, p_{i,1}, p_{i,2} \ldots\ldots\ldots\ldots, p_{i,L}] \qquad 6.2.3$$

P_i and S_i are $(L+1)$-component row vectors, and S_i^T is the transpose of S_i.

P_i is considered to be the sampled impulse response, at time $t=iT$, of the linear base-band channel in Fig.6.2.1. The real and imaginary parts of the complex-valued noise components $\{v_i\}$ are statistically independent Gaussian random variables with zero mean and variance σ_v^2, and the linear demodulator is taken to be such that σ_v^2 is also the two-sided power spectral density of the additive white Gaussian noise introduced at the output of the HF channel in Fig.6.2.1 [17,25,30,61]. The detector in Fig.6.2.1 is assumed to give correct detection of all data symbols, even at low signal/noise ratios, so that :

$$S'_i = S_i \quad \text{for all value of } \{i\} \qquad 6.2.4$$

The above assumption is necessary for testing the estimator preventing errors in detection affecting the estimator performance. We are here concerned to evaluate the performance of the estimators when operating over a fading channel. The signal r_i and S'_i are fed to the channel estimator to give an estimate of the channel sampled impulse response P'_i at time $t=iT$, where :

$$P'_i = [p'_{i,0}, p'_{i,1}, p'_{i,2} \ldots\ldots\ldots\ldots, p'_{i,L}] \qquad 6.2.5$$

which is necessary for the detection process. In all tests the receiver is assumed to have no prior knowledge of the relative transmission delay over the three fading paths or even of the fact that there are three or two paths, so that none of the $\{p'_{i,h}\}$ for $h=0,1,\ldots\ldots,L$ are simply set to zero.

6.3 ESTIMATOR 1 (Least Mean-Square (LMS) "Gradient" Estimator)

The gradient channel estimator originally proposed for use with a maximum likelihood detector employing the Viterbi algorithm [65], after modification for use with complex-valued symbols, becomes the linear feed-forward estimator, and is also called the simple estimator [61].

The estimated sampled impulse response of the channel denoted as $\{P'_i\}$, is adjusted recursively to minimise the mean square error (MSE) between the actual received sequence $\{r_i\}$ and the output of the estimator (estimated received sequence) $\{r'_i\}$, where :

$$r'_i = \sum_{h=0}^{L} s_{i-h} \cdot p'_{i-1,h} = P'_{i-1} \cdot S_i^T \qquad 6.3.1$$

Assuming that eqn.6.2.4 holds. The error in estimated received signal next is determined by :

$$e_i = r_i - r'_i \qquad 6.3.2$$

Thus the MSE between the received signal and the estimate r'_i is given by :

$$J(p'_{i-1}) = E[|r_i - r'_i|^2] = E[|e_i|^2] = E[e_i \cdot e_i^*] \qquad 6.3.3$$

where * denotes the complex conjugate. From eqn.6.3.1 the function $J(p'_{i-1})$ can be written as :

$$J = E[(r_i - \sum_{h=0}^{L} s_{i-h} \cdot p'_{i-1,h})(r_i - \sum_{h=0}^{L} s_{i-h} \cdot p'_{i-1,h})^*] \Rightarrow$$

$$J = E[|r_i|^2 - \sum_{h=0}^{L} p'^*_{i-1,h} E[s^*_{i-h} \cdot r_i] - \sum_{h=0}^{L} p'_{i-1,h} \cdot E[s_{i-h} \cdot r_i^*]$$

$$+ \sum_{h=0}^{L} \sum_{k=0}^{L} p'^*_{i-1,h}\, p'_{i-1,k} \cdot E[s^*_{i-h} \cdot s_{i-k}]$$

Then:

$$J = \sigma_r{}^2 - \sum_{h=0}^{L} p'^*_{i-1,h} E[s^*_{i-h} \cdot r_i] - \sum_{h=0}^{L} p'_{i-1,h} \cdot E[s_{i-h} \cdot r_i^*]$$

$$+ \sum_{h=0}^{L} \sum_{k=0}^{L} p'^*_{i-1,h}\, p'_{i-1,k} \cdot E[s^*_{i-h} \cdot s_{i-k}] \qquad 6.3.4$$

where : E[.] is the expected value or (the statistical expectation operator).

and $\sigma_r^2 = E[|r_i|^2]$ 6.3.5

is the variance of the received signal r_i. Now let :

$$\Phi_{h-k} = E[s^*_{i-h} \cdot s_{i-k}] \qquad 6.3.6$$

where Φ_{h-k} is the auto-correlation function of the transmitted signal for lag (h-k), and

$$Q(-h) = E[s^*_{i-h} \cdot r_i] \quad , \quad Q^*(-h) = E[s_{i-h} \cdot r_i^*] \qquad 3.6.7$$

where $Q(-h)$ is the cross-correlation between the received signal and transmitted signal eqn.5.3.4 becomes :

$$J=\sigma_r^2 - \sum_{h=0}^{L} p'^{*}_{i-1,h} \cdot Q(-h) - \sum_{h=0}^{L} p'_{i-1,h} \cdot Q^{*}(-h) + \sum_{h=0}^{L}\sum_{k=0}^{L} p'^{*}_{i-1,h}\, p'_{i-1,k} \cdot \Phi_{k-h} \qquad 6.3.8$$

The cost function J, or " mean-square error" is precisely a second-order function of the impulse response vector P_{i-1} and the minimum value (optimum solution) can be obtained by the setting the gradient (L+1)-vector ∇ to zero [1,63,65], such that :

$$\nabla_h(J) = \frac{\partial J}{\partial P'_{i-1}} = [\frac{\partial J}{\partial p'_{i-1,0}}, \frac{\partial J}{\partial p'_{i-1,1}}, ..., \frac{\partial J}{\partial p'_{i-1,L}}] \qquad 6.3.9$$

$$\nabla_h(J) = 0 \quad , \quad \text{for } h = 0,1,.....,L \qquad 6.3.10$$

The h^{th} impulse response element is given in the complex form :

$$p'_{i-1,h} = \Re p'_{i-1,h} + j\,\Im p'_{i-1,h} \quad , \text{ for h} = 0,1,.....,\text{L} \qquad 6.3.11$$

From eqns.6.3.8 and 6.3.9 and after some mathematical operations $\nabla_h(j)$ can be written as:

$$\nabla_h(J) = -2Q(-h) + 2\sum_{h=0}^{L} p'_{i-1,h} \cdot \Phi_{k-h} \qquad 6.3.12$$

for $h=0,1,.....,L$

The optimum impulse response which minimises the MSE is thus the solution of a set of h simultaneous linear equations [1,63] :

$$\sum_{h=0}^{L} p'_{i-1,h} \cdot \Phi_{k-h} = Q(-h) \quad , \quad \text{for } h = 0,1,.....,L \qquad 6.3.13$$

In order to calculate the optimum impulse response P'_{i-1} of the channel response precise knowledge of the auto-correlation matrix and the cross-correlation vector is required. In practice it is difficult to calculate the auto-correlation and cross-correlation, therefore the task of the adaptive algorithm is to define the value of P'_{i-1} in an iterative manner. The steepest-descent algorithm is the simplest way to update the sample impulse response recursively [1,61,63,65,66,91] and there is no need to calculate the auto-correlation matrix and cross-correlation vector. This approach proceed as follows :

1.) Begin with an initial value $P'_{i-1}(0)$ for the estimated SIR, which provides an initial guess as to where the minimum point of the error-performance surface may be located. Typically, P'_0 is set equal to the zero vector.

2.) Using this initial or present guess, the gradient vector can be computed. The real and imaginary parts of which are defined as the derivative of the mean-squared error J, evaluated with respect to the real and imaginary parts of the SIR vector P'_i at time iT .

3.) The next estimated SIR can be computed by making a change in the initial or present guess in a direction opposite to that of the gradient vector.

4.) The process is next repeated from step 2.

It is intuitively reasonable that successive corrections of the estimated SIR vector P'_{i-1} in the direction of the negative of the gradient vector [63,66,87,91], i.e. (in the direction of the steepest descent of the error-performance surface) should eventually lead to the minimum mean-square error $J_{\min}$, at which point the SIR vector assumes its optimum value. This can be likened to releasing a ball at any point on the error surface. It will always fall in the direction of the steepest slope. If its motion is arrested at regular intervals of time, it will eventually reach the lowest point. The update sample impulse response P'_i is then given by [1,61,63,65,66,91] :

$$P'_i = P'_{i-1} + b' \cdot (-\nabla_i(J)) \qquad 6.3.14$$

Substitute eqns.6.3.6 and 6.3.7 in eqn.6.3.12, the gradient vector can be written as :

$$\nabla_h(J) = -2E[s^*_{i-h} \cdot r_i] + 2\sum_{h=0}^{L} p'_{i-1,h} \cdot E[s^*_{i-h} \cdot s_{i-k}] \qquad 6..3.15$$

From eqn.6.2.1 the cross-correlation matrix *Q(-h)* can be calculated as :

$$\begin{aligned} Q(-h) &= E[s^*_{i-h} \cdot r_i] = E[s^*_{i-h} \cdot (\sum_{h=0}^{L} s_{i-h} \cdot p_{i-1,h} + v_i)] = \\ &= \sum_{h=0}^{L} p_{i-1,h} \cdot |s_{i-h}|^2 \end{aligned} \qquad 6.3.16$$

where : $E[s^*_{i-h} \cdot v_i] = 0$ for $h = 0,1,\ldots\ldots,L$

Assuming that the information symbols are uncorrelated, i.e. $E[s_k \cdot s^*_j] = |s_k|^2$ and that the information sequence $\{s_k\}$ is uncorrelated with the additive noise sequence $\{v_i\}$.

From eqns.6.3.15 and 6.3.16 then :

$$\begin{aligned} \nabla_h(J) &= -2\sum_{h=0}^{L} p_{i-1,h} \cdot [s_{i-h} s^*_{i-h}] + 2\sum_{h=0}^{L} p'_{i-1,h} \cdot [s^*_{i-h} \cdot s_{i-h}] = \\ &= -2\, s^*_{i-h}[\sum_{h=0}^{L} p_{i-1,h} \cdot s_{i-h} - \sum_{h=0}^{L} p'_{i-1,h} \cdot s_{i-h}] = -2\, s^*_{i-h} \cdot e_i \end{aligned} \qquad 6.3.17$$

Eqn.6.3.17 can be derived directly as follows :

The gradient of the cost function (J) with respect to the impulse response vector P'_i can be estimated directly from the input data $\{r_i,\ r'_i\}$. Suppose we compute the gradient of the instantaneous squared error, that is :

$$\nabla J(i) = \nabla_{p'i-1}[e^2(i)] = 2e(i)\cdot\nabla_{p'i-1}\cdot e(i) = 2\,e_i\cdot\nabla_{p'i-1}[r_i - r'_i] =$$
$$= 2\,e_i\cdot\nabla_{p'i-1}[r_i - P'^{T}_{i-1}s^*_i] = -2\,e_i\cdot s^*_i$$

This derivation relies on the fact that neither r_i or s_i are affected by changes in P'_i [61,65,66,91]. This vector ∇J_i is termed the instantaneous gradient since it is based only on the value of e and s_i at time $t=iT$. When ∇J_i is used in place of true gradient, the result is the least mean squares (LMS) algorithm popularised by Widrow and Hoff [91,99]. Substitute eqn.6.3.17 into 6.3.14, then :

$$P'_i = P'_{i-1} + b\cdot e_i\cdot S^*_i \qquad \text{or in another form}$$
$$p'_{i,h} = p'_{i-1,h} + b\cdot e_i\cdot s^*_{i,h} \qquad \text{for } h = 0,1,\ldots\ldots,L \qquad 6.3.18$$

b is an appropriate small positive real-valued constant called the step size of LMS algorithm, and can often be ($b<1$). Eqn.6.3.18 is the conventional gradient algorithm, which for convenience is now referred to as estimator 1. This estimator assumes that the SIR of the channel varies only very slowly with time. The algorithm for estimator 1 is, in fact a recursive solution to the least squares estimation problem, also termed as the least mean-square LMS error algorithm [27,63,75,91]. This algorithm has become very popular since it may be implemented using only two multiply accumulate operations per tap coefficient in each sample interval. This is referred to as having a processing order of "$2L$". Fig.6.3.1 shows the digital implementation of the gradient estimator (estimator 1). The complete algorithm for estimator 1 is given by eqns.6.3.2 and 6.3.18 as :

$$r'_i = \sum_{h=0}^{L} s_{i-h}\cdot p'_{i-1,h} \quad ; \qquad e_i = r_i - r'_i$$
$$p'_{i,h} = p'_{i-1,h} + b\cdot e_i\cdot s^*_{i-h} \qquad \text{for h} = 0,1,\ldots\ldots,\text{L}$$

Note that the second term ($b\cdot e_i\, s^*_{i-h}$) represents the correction factor that is applied to the current estimate of SIR of the channel. It can be seen that, in the absence of noise, e_i is proportional to the error (discrepancy from the ideal) in the value of the orthogonal projection of P'_{i-1} onto the one-dimensional subspace spanned by s^*_i.

6.4 ESTIMATOR 2

This system is based on the simple conventional gradient estimator (LMS algorithm with fixed step size) [51,61,65], and using a degree-1 prediction [23,60,64] to improve the algorithm's ability to track the variation of the time-varying channel. Hence the performance of estimator 1 can be improved when an appropriate prediction technique is employed with the gradient algorithm [61]. Bear in mind that the near-maximum likelihood detector (see Fig.6.2.1) introduces an inherent delay of several sampling intervals. Thus the detected data symbols $\{s'_i\}$ at the output of the detector (Fig.6.2.1) have a delay in detection of $(n-1)$, i.e. $\{s'_i\}$ is detected after the reception of r_{i+n-1}. Thus a feed-forward estimator makes an estimate P'_i of P_i and is only available to the NML detector on the receipt of r_{i+n}. Therefore, there is a delay of n sampling intervals in the estimation of P_i. Eqn.6.3.18 now can be replaced by :

$$P'_i = P'_{i,i-1} + b \cdot e_i \cdot S_i^* \qquad 6.4.1$$

where $P'_{i,i-1}$ is the one-step prediction of P_i at time $t=(i-1)T$ and is given by :

$$P'_{i,i-1} = [p'_{i,i-1,0}, p'_{i,i-1,1}, p'_{i,i-1,2} \cdots\cdots\cdots\cdots, p'_{i,i-1,L}] \qquad 6.4.2$$

The estimator uses one-step prediction $P'_{i,i-1}$ to form the updated estimate of P_i, given by eqn.6.2.5 and the one-step prediction of P_{i+1}, is given by $P'_{i+1,i}$. As has been mentioned before in (chapter 4 and 5), an important advantage gained by using the adaptive linear filter in Fig.6.2.1 is that it avoids the need for prediction over many sampling intervals such as must be used in the absence of the filter [23,30,31]. prediction over many sampling intervals can increase considerably the error in the prediction [23,30,61,64]. Where the adaptive filter is used to make the SIR of the channel and filter to be near minimum phased. Further details on the adaptive filter and detector are given in chapter 4 and 5. The estimator in Fig.6.2.1 is fed by the early detected data symbol $\{s'_i\}$, and the correct detection of all data symbols is assumed, even at low signal/noise ratio, so that eqn.6.2.4 holds. This assumption gives us the freedom to test the estimator performance without being affected by detection errors.

A Least-Square fading memory prediction is used to make a one-step prediction of the channel SIR. This is done by determining a set of $(L+1)$ polynomials of degree-1, from the sequence of vectors $P'_i, P'_{i-1}, \ldots\ldots\ldots$, each of which gives the weighted least-square fit to the components in the corresponding location in the vectors $P'_i, P'_{i-1}, \ldots\ldots\ldots$, using the values of the polynomial, at time $t=(i+1)T$ to determine the $(L+1)$ components of $P'_{i+1,i}$. It has been shown [31,61] that the degree-1 polynomial gives the best performance overall, for different versions of the prediction process. The chosen polynomial is such

that it gives the best fit to the sequence of past observations and the exponentially weighted sum of the squares of the error function is minimised [60,64]. In [64], the technique is applied to the prediction of a value of a variable parameter, derived from past (noisy or inaccurate) observations of the parameter, the observations being unaffected by the prediction. The technique can also be applied to prediction from past updated estimates of the parameter, the prediction here influencing the subsequent updated estimate. Extensive tests have shown that this technique has improved the estimator performance without any sign of instability [61]. The estimator determines the estimated error in the prediction of SIR of the channel using

$$\varepsilon_i = P'_i - P'_{i,i-1} \qquad 6.4.3$$

where P'_i is the updated estimate of P_i, which is given by eqn.6.3.18 and is formed using the conventional gradient algorithm. $P'_{i,i-1}$ is the one-step prediction of P_i given by eqn.6.4.2. Bear in mind that the actual error in $P'_{i,i-1}$ is $(P_i - P'_{i,i-1})$.

From eqn.6.3.17 the estimator determines the correction matrix Δ_i, where :

$$\Delta_i = b \cdot e_i \cdot S_i^* \qquad 6.4.4$$

whose element is :

$$\delta_{i,h} = b \cdot e_i \cdot s^*_{i-h} \quad ; \quad \text{for } h = 0,1,\ldots\ldots,L \qquad 6.4.5$$

The correction vector is next fed to the appropriate least squares fading memory polynomial filter with degree-1 to give the prediction of (P_{i+1}) as follows :

$$P''_{i+1,i} = P''_{i,i-1} + (1-\lambda)^2 \cdot \Delta_i \qquad 6.4.6$$

$$P'_{i+1,i} = P'_{i,i-1} + P''_{i+1,i} + (1-\lambda^2) \cdot \Delta_i \qquad 6.4.7$$

where $(P''_{i+1,i})$ is the prediction of the rate of change with i of (P_{i+1}), $(L+1)$-components row vector, and $(P'_{i+1,i})$ is the degree-1 least square fading memory prediction of (P_{i+1}), $(L+1)$-components row vector [61,64]. The symbol λ in eqns.6.4.6 and 6.4.7 is a real-valued constant in the range *0* to *1*, usually close to *1*. At the start of the estimation process the estimator sets :

$$P''_{1,0} = [0.0] \quad \& \quad P'_{1,0} = [0.0] \qquad 6.4.8$$

The constants b & λ are optimised in combination so that the error in the one-step prediction of the sampled impulse response of the channel is minimised. This estimator was developed by [60] and fully examined over HF channels, and its performance is presented here for comparison purposes. The operation of estimator 2 are now as follows.

$$S_i = [s_i, s_{i-1}, s_{i-2} \ldots\ldots\ldots\ldots, s_{i-L}] \qquad 6.4.9$$

$$r'_i = P'_{i,i-1} \cdot S_i^T \qquad 6.4.10$$

$$e_i = r_i - r'_i$$

$$\Delta_i = b \cdot e_i \cdot S_i^* \qquad 6.4.11$$

$$P''_{i+1,i} = P''_{i,i-1} + (1-\lambda)^2 \cdot \Delta_i$$

$$P'_{i+1,i} = P'_{i,i-1} + P''_{i+1,i} + (1-\lambda^2) \cdot \Delta_i \qquad 6.4.12$$

The estimator to be studied next, is a development of estimator 2 and it employs the degree-1 least squares fading-memory prediction algorithm to evaluate ($P'_{i+1,i}$), with the step-size b of the LMS algorithm to update the correction matrix being adaptively adjusted.

6.5 ESTIMATOR 3 " Adaptive LMS Estimator "

This estimator is a development of the conventional gradient estimator [27,26,31,60-62,67], but now the step size of LMS algorithm is adaptive [63,66,90], in order to improve the algorithm's ability to track more rapidly fading channels. This system does not exploit any prior knowledge of the number of separate paths (sky-waves nor of their relative transmission delays).

Estimator 3 uses the linear feed-forward transversal filter (Fig.6.3.1) as in estimator 1. The estimator holds in store the detected data symbols :

$$S'_i = [s'_i, s'_{i-1}, s'_{i-2} \ldots\ldots\ldots\ldots, s'_{i-L}] \qquad 6.5.1$$

Assuming correct detection, i.e. eqn.6.2.4 is held, then eqn.6.5.1 can be written as :

$$S'_i = [s_i, s_{i-1}, s_{i-2} \ldots\ldots\ldots\ldots, s_{i-L}] \qquad 6.5.2$$

The estimated received signal r'_i is next determined using :

$$r'_i = \sum_{h=0}^{L} s_{i-h} \cdot p'_{i,i-1,h} = P'_{i,i-1} \cdot S_i^T \qquad 6.5.3$$

where ($P'_{i,i-1}$) is the one-step prediction of P_i given by eqn.6.4.2. The error in the estimated received signal is given by :

$$e_i = r_i - r'_i \qquad 6.5.4$$

The one-step prediction of P_i, given by ($P'_{i,i-1}$) in eqn.6.4.2, is obtained using the conventional gradient algorithm [51,61,65] and is given by :

$$P'_i = P'_{i,i-1} + b \cdot e_i \cdot S_i^* = P'_{i,i-1} + \Delta_i \qquad 6.5.5$$

where : $(\Delta_i = b \cdot e_i \cdot S_i^*)$ is the correction vector.

The error from the prediction process can be obtained from :

$$\varepsilon_i = P'_i - P'_{i,i-1} \qquad 6.5.6$$

Bearing in mind the actual error in $P'_{i,i-1}$ is $(P_i - P'_{i,i-1})$.

The prediction of P_{i+1} is now formed using eqns.6.4.6 and 6.4.7, where in the prediction technique, it is assumed that the rate of change of P_i with (i) is constant or only slowly varies with (i). Thus a considerable source of error in a prediction $(P'_{i,i-1})$ is likely to be the acceleration (variation in the rate of change) in P_i. Since the latter process uses the correction vector with fixed step size b, therefore the performance of the system can be improved by letting the step-size vary with (i), i.e. the step-size now becomes adaptive.
The correction vector is now replaced by :

$$\Delta_i = C_i \cdot e_i \cdot S_i^* \qquad 6.5.7$$

where Δ_i is the correction $(L+1)$-components row vector :

$$\Delta_i = [\delta_{i,0}, \delta_{i,1}, \delta_{i,2} \ldots\ldots\ldots\ldots, \delta_{i,L}] \qquad 6.5.8$$

whose h^{th} element is given by :

$$\delta_{i,h} = c_{i,h} \cdot e_i \cdot s^*_{i-h} \quad ; \quad \text{for } h = 0,1,\ldots\ldots\ldots, L \qquad 6.5.9$$

C_i is now a time-varying step-size $(L+1)$-components row vector and is given by :

$$C_i = [c_{i,0}, c_{i,1}, c_{i,2} \ldots\ldots\ldots\ldots, c_{i,L}] \qquad 6.5.10$$

The time varying step size is determined as follows :

The estimator first calculates the absolute value of the updated estimate of SIR of the channel given by :

$$z_{i,h} = |p'_{i,i-1,h}| \quad ; \quad \text{for} \quad h = 0,1,\ldots\ldots\ldots, L \qquad 6.5.11$$

using the fact that the greater the maximum absolute value of any $p_{i,h}$ the greater is likely to be its maximum acceleration and hence the greater the probable value of largest error in

the corresponding prediction $p'_{i,i-1,h}$ [23]. $p'_{i,i-1,h}$ is complex-valued and its absolute value is determined by :

$$zi,h = | p'_{i,i-1,h} | = \sqrt{[(\Re p'_{i,i-1,h})^2 + (\Im p'_{i,i-1,h})^2]} \qquad 6.5.12$$

for $h=0,1,\ldots\ldots\ldots\ldots,L.$

Where $\Re p'$ and $\Im p'$ are the real and imaginary parts of ($p'_{i,i-1,h}$). The step size vector C_i is now given by [23] :

$$C_i = f(Z_i) \qquad 6.5.13$$

whose hth element $c_{i,h}$ is given by :

$$c_{i,h} = f(z_{i,h}) \quad \text{for} \quad h = 0,1,\ldots\ldots,L \qquad 6.5.14$$

where $f(z_{i,h})$ is a monotonically non-decreasing positive real-value function of $z_{i,h}$ [23]. This will prevent $c_{i,h}$ from becoming too large, and in order to avoid possible instability of the algorithm given by eqn.6.5.7, the value of $c_{i,h}$ should be limited such that :

$$d_1 < c_{i,h} < d_2 \qquad 6.5.15$$

where (d_1 & d_2) are appropriate positive real-value constants. Computer simulation tests have shown that the best performance of the estimator is obtained, when $c_{i,h}$ varies non-linearly with ($z_{i,h}$) over the range (d_1 to d_2) [23,28], which prevents $c_{i,h}$ from becoming too large. Therefore the parameter ($c_{i,h}$) is chosen as a non-linear function of ($z_{i,h}$) that satisfies the following conditions :

$$\text{when:} \quad \begin{array}{ll} z_{i,h} > d_0 & c_{i,h} = (z_{i,h})^{0.5} \\ z_{i,h} < d_0 & c_{i,h} = d_1 = 10^{-6} \end{array} \qquad 6.5.16$$

The parameter (d_0) is adjustable to suit the channel characteristics and tests have shown that the algorithm is stable and gives a good performance over the whole range of signal-to-noise ratio when ($d_0 = 0.003$) for the three channel model considered in these tests [114]. After the step size is determined, the estimator uses the degree-1 prediction process to form the prediction of P_{i+1} of the SIR as in estimator 2. The whole algorithm for estimator 3 can be summarised as follows :

First the estimator forms the estimated received signal given by :

$$r'_i = \sum_{h=0}^{L} s_{i-h} \cdot p'_{i,i-1,h} = P'_{i,i-1} \cdot S_i^T \qquad 6.5.17$$

The error in estimated received signal is next formed using :

$$e_i = r_i - r'_i \qquad 6.5.18$$

The estimator determines the correction vector given by :

$$\Delta_i = C_i \cdot e_i \cdot S_i^* \qquad 6.519$$

Or in the element form :

$$\delta_{i,h} = c_{i,h} \cdot e_i \cdot s^*_{i-h} \quad ; \quad \text{for } h = 0,1,\ldots\ldots,L \qquad 6.5.20$$

$c_{i,h}$ is the time-varying step size given by eqn.6.5.12 and 6.5.16. The correction vector Δ_i is next passed through a degree-1 prediction process to give the prediction of P_{i+1} as :

$$P''_{i+1,i} = P''_{i,i-1} + (1-\lambda)^2 \cdot \Delta_i$$
$$P'_{i+1,i} = P'_{i,i-1} + P''_{i+1,i} + (1-\lambda^2) \cdot \Delta_i \qquad \text{or :}$$
$$p''_{i+1,i,h} = p''_{i,i-1,h} + (1-\lambda)^2 \cdot \delta_{i,h} \qquad \text{for } h=0,1,\ldots,L \qquad 6.5.21$$
$$p'_{i+1,i,h} = p'_{i,i-1,h} + p''_{i+1,i,h} + (1-\lambda^2) \cdot \delta_{i,h}$$

$P'_{i+1,i}$ is the degree-1 least-square fading memory prediction of P_{i+1} and the vector $P''_{i+1,i}$ is a prediction of the rate of change with respect to (i) of P_{i+1}.

At the start of estimation process :

$$P''_{1,0} = [0.0] \quad \& \quad P'_{1,0} = [0.0] \qquad 6.5.22$$

The results of computer simulation tests on a model of a digital data transmission system are given at the end of this chapter.

6.6 RECURSIVE LEAST-SQUARE (RLS) KALMAN ESTIMATION

The Kalman filter as an estimator has become the most widely used algorithm for an unknown channel and in particular for the HF radio links [1,27,48,50,64,69-72,78,80-84,98], for which under the appropriate conditions, it is the optimum estimator. A Kalman algorithm gives the least-squares estimate of the SIR of a time invariant channel, that introduces additive white Gaussian noise and was completely unknown at the start of estimation process [63,64,74,97]. Since the result holds for any time after the start of the process, it follows that a Kalman estimator provides the most rapid rate of convergence. The main problem with the Kalman algorithm is its complexity, both in its theoretical analysis and in its practical implementation.

The RLS Kalman algorithm gives a convergence rate that is far superior to that of the least mean squares (LMS) algorithm [1,63,84,97]. But this superior convergence rate of the RLS Kalman algorithm is at the expense of increased computation. Furthermore the RLS Kalman algorithm has been found to be numerically unstable [48] for particular channel environments (such as the HF channel), therefore anther version of RLS Kalman algorithm has been proposed by F. Hsu [48,68,85], which has inherently better stability and numerical accuracy than the conventional Kalman algorithm. This is done by square root formulations of the Kalman algorithm. It is necessary for convenience to describe first the method of the RLS algorithm towards the description of the square root Kalman estimator (SRK), and assess this algorithm for HF channel estimator, referred to as estimator 4. This chapter concludes with a modified version of SRK estimator that employs an appropriate prediction process. The latter takes into account the rate of change in the estimate of the SIR. This is referred to as estimator 5

Now we briefly restate the RLS criterion applied to the channel estimation. The sample of the received signal $r(t)$ at time $t=iT$ is given by :

$$r_i = \sum_{h=0}^{L} s_{i-h} \cdot p_{i,h} + v_i = P_i \cdot S_i^T + v_i \qquad 6.6.1$$

and the sample of the estimated received signal r'_i formed by the estimator is given by :

$$r'_i = \sum_{h=0}^{L} s_{i-h} \cdot p'_{i,h} = P'_i \cdot S_i^T \qquad 6.6.2$$

Where : P'_i is the estimated SIR of the channel which is given by $(L+1)$-components row vector.

$$P'_i = [p'_{i,0}, p'_{i,1}, p'_{i,2} \ldots\ldots\ldots\ldots, p'_{i,L}] \qquad 6.6.3$$

and S_i^T is the transpose of the row vector S_i and is given by eqn.6.2.2. Bear in mind that correct detection is assumed even at a low signal-to noise ratio. The error in r'_i is given by

$$e_i = r_i - r'_i = r_i - P'_i \cdot S_i^T \qquad 6.6.4$$

The recursive least-square (RLS) estimation of r'_i may be formulated as follows :

The update estimate of the SIR is determined such as to minimise the time average square error given by [1,63,75] :

$$J_i = \sum_{h=0}^{i} w^{i-h} |e_i|^2 \qquad 6.6.5$$

where w represents a weighting factor in the range ($0 < w < 1$). The quantity J_i is the weighted square error in the $\{r'_h\}$, starting with r'_0. The algorithm to be studied here is recursive in the sense that, on the receipt of each sample r_i, it repeats a sequence of operations on a set of parameter values determined after the receipt of the previous sample, to evaluate the corresponding parameter values, ready for the next received sample, and so on. The algorithm is least square in the sense that it minimises the weighted squared error in the $\{r'_i\}$. The quantity minimised by the RLS algorithm is the actual squared error rather than the expected value of the squared error that is minimised by the LMS algorithm. From eqns. 6.6.2, 6.6.4 and 6.6.5 the least square cost function J_i at time $t=iT$ becomes :

$$J_i = \sum_{h=0}^{i} w^{i-h} \cdot |r_h - P'_i \cdot S_h^T|^2 \qquad 6.6.6$$

The algorithm determines the estimated SIR of the channel P'_i at time $t=iT$, which minimises the cost function J_i. Eqn.6.6.6 can be written as :

$$J_i = \sum_{h=0}^{i} w^{i-h} \cdot (r_h - P'_i \cdot S_h^T) \cdot (r_h - P'_i \cdot S_h^T)^{\mathrm{H}} \qquad 6.6.7$$

Where $(\cdot)^{\mathrm{H}}$ is the transpose conjugate of $(\cdot)$. The cost function J_i and the element of the vector P'_i can then be seen to have a bowl-shaped surface with a unique minimum [1,63]. At the bottom or minimum point the gradient of J_i with respect to P'_i is

$$\nabla J_i = [\frac{\partial J_i}{\partial p'_{i,0}}, \frac{\partial J_i}{\partial p'_{i,1}}, ..., \frac{\partial J_i}{\partial p'_{i,L}}] \qquad 6.6.8$$

Where $p'_{i,h}$ is complex-valued, for $h=0,1,2,.....,L$.

Differentiating eqn.6.6.7 with respect to P'_i and setting the gradient of J_i equal to zero, yields the set of linear equations [1,23,63] :

$$P'_i \cdot \sum_{h=0}^{i} w^{i-h} \cdot S_h^* \cdot S_h^T = \sum_{h=0}^{i} w^{i-h} \cdot r_h \cdot S_h^* \qquad 6.6.9$$

Or: $$P'_i \cdot R_i = Q_i \qquad 6.6.10$$

Where $$R_i = \sum_{h=0}^{i} w^{i-h} \cdot S_h^* \cdot S_h^T \qquad 6.6.11$$

and: $$Q_i = \sum_{h=0}^{i} w^{i-h} \cdot r_h \cdot S_h^* \qquad 6.6.12$$

R_i is the signal correlation vector and Q_i is the cross-correlation vector. From eqn.6.6.10 the optimum SIR is given by :

$$P'_i = R_i^{-1} \cdot Q_i \qquad 6.6.13$$

To determine P'_i from eqn.6.6.13, would mean enormous computational complexity, therefore, P'_i, R_i, Q_i can be obtained recursively from P'_{i-1} , R_{i-1}, Q_{i-1}.
Eqn.6.6.11 can be written as :

$$R_i = w \cdot R_{i-1} + S_h^* \cdot S_h^T \qquad 6.6.14$$

After some mathematical operation, the P'_i can be shown to be updated recursively according to the relation :

$$P'_i = P'_{i-1} + K_i \cdot e_i \qquad 6.6.15$$

Where K_i is the Kalman gain vector which is given by :

$$K_i = (w + S_i^* \cdot \Phi_{i-1} \cdot S_i^T)^{-1} S_i^* \cdot \Phi_{i-1} \qquad 6.6.16$$

Where Φ_i is given by : $\Phi_i = R_i^{-1}$ and the update of the inverse of the correlation-matrix is given by :

$$\Phi_i = w^{-1}(\Phi_{i-1} - \Phi_{i-1} \cdot S_i^T \cdot K_i) \qquad 6.6.17$$

The algorithm described by eqns.6.6.15, 6.6.17 and 6.6.4 is called the RLS direct form or Kalman algorithm. The derivation of the above algorithm is given in Appendix D.
The classic Kalman filter presented in the previous discussion provides a tracking rate sufficient for a fading HF radio channel. Unfortunately the number of computations or operations (multiplications, divisions, and subtractions) in computing the variables in eqn.6.6.15 is proportional to (L^2). Most of these operations are involved in the updating of Φ_i. This part of the computation is also susceptible to round off noise [1,48,68]. Also it has been found by [48] that the covariance update formula (eqn.6.6.17) is numerically unstable. The main reason for this instability is that the algorithm forms the matrix Φ_i, which is computed as a difference of two positive semidefinite matrices. Therefore numerical accuracy makes the algorithm sensitive to computer round-off error, which some-times degrades the performance of the estimator [48]. The accuracy degeneration of the algorithm may result in a Φ_i matrix, which is indefinite (having both positive and

negative eigenvalues). To remedy that problem, an alternative algorithm has been developed which avoids the computation of Φ_i according to eqn.6.6.17. The basis of this algorithm lies in the decomposition of Φ_i in the form :

$$\Phi_i = U_i^* \cdot D_i \cdot U_i^T \qquad 6.6.18$$

This decomposition is called a square-root factorisation. The Square-Root (S-R) algorithm has been implemented in a decision-feedback equalised Phase Shift Keying (PSK) modem to transmit at high speed over HF channels. This algorithm has been found to be numerically stable and exhibits good numerical properties [48].

The next two estimators to be described will depend on the SRK algorithm, where the latter will be modified to be used as a channel estimator.

6.6.1 ESTIMATOR 4 " SRK Estimator "

Estimator 4 is a simple modification of the square-root Kalman algorithm, which has been developed by Hsu [48] to be implemented in a decision feedback equaliser to update the tap-coefficients of the equaliser.

The basic principle of this algorithm is based on the decomposition of the error covariance matrix Φ_i given by eqn.6.6.17 in the form given by eqn.6.6.18. Here we will derive briefly the estimation algorithm based on the square-root Kalman algorithm (SRKA). Let the error covariance matrix given by :

$$\Phi(i) = U^*(i) \cdot D\ (i) \cdot U^T(i) \qquad 6.6.19$$

Where U is an upper triangular matrix with unit diagonal elements and non-equal off-diagonal elements μ_{ij} for $i = 1,2,.....,L-1$; $j = i+1, i+2,.....,L$.

and D is diagonal matrix with real diagonal elements $[d_1, d_2, ..., d_L]$.

Eqn.6.6.19 can be written as :

$$\begin{bmatrix} 1,\mu_{12}^*,\mu_{13}^*,.....,\mu_{1L}^* \\ 0,1,\mu_{23}^*,.....,\mu_{2L}^* \\ . \\ 0,0,0,.....,1 \end{bmatrix} \cdot \begin{bmatrix} d_1,0,0,.....,0 \\ 0,d_2,0,.....,0 \\ . \\ 0,0,0,.....d_L \end{bmatrix} \cdot \begin{bmatrix} 1\ ,\ \ 0\ ,\ \ 0,.....,0 \\ \mu_{12},\ 1\ ,\ \ 0,.....,0 \\ . \\ \mu_{1L},\mu_{2L},\mu_{3L},.....,1 \end{bmatrix} \qquad 6.6.20$$

The matrix $\Phi(i)$ is given by eqn.6.6.17 as

$$\Phi_i = w^{-1}(\Phi_{i-1} - K_i \cdot S_i^T \cdot \Phi_{i-1}) \qquad 6.6.21$$

and : $$K_i = (w + S_i^T \cdot \Phi_{i-1} \cdot S_i^*)^{-1} \cdot \Phi_{i-1} S_i^* \qquad 6.6.22$$

Where w^{-1}, w are now given by [48] :

$$w^{-1} = (1+q) \quad \& \quad w = \eta$$

Then eqns.6.6.21 and 6.6.22 becomes :

$$\Phi_i = (1+q)\cdot(\Phi_{i-1} - K_i \cdot S_i^T \cdot \Phi_{i-1}) \qquad 6.6.23$$

and : $$K_i = (\eta + S_i^T \cdot \Phi_{i-1} \cdot S_i^*)^{-1} \cdot \Phi_{i-1} S_i^* \qquad 6.6.24$$

Let : $$\alpha = (\eta + S_i^T \cdot \Phi_{i-1} \cdot S_i^*) \qquad 6.6.25$$

Then eqn.6.6.24 becomes :

$$K_i = \alpha^{-1} \cdot \Phi_{i-1} S_i^* \qquad 6.6.26$$

Now let the vector F and B be defined by :

$$F_{i-1} = U_{i-1}^T \cdot S_i^* \qquad 6.6.27$$

and : $$B_{i-1} = D_{i-1} \cdot F_{i-1} = D_{i-1} \cdot U_{i-1}^T \cdot S_i^* \qquad 6.6.28$$

Then: $$K_i = \alpha^{-1} \cdot U_{i-1}^* \cdot D_{i-1} \cdot U_{i-1}^T \cdot S_i^* = \alpha^{-1} \cdot U_{i-1}^* \cdot B_{i-1} \qquad 6.6.29$$

Substituting eqns.6.6.29, 6.6.19 into eqn.6.6.23 then :

$$\begin{aligned} U_i^* \cdot D_i \cdot U_i^T &= (1+q)\cdot[U_{i-1}^* \cdot D_{i-1} \cdot U_{i-1}^T - \\ &\quad -\alpha^{-1} \cdot U_{i-1}^* \cdot D_{i-1} \cdot U_{i-1}^T \cdot S_i^* \cdot S_i^T \cdot U_{i-1}^* \cdot D_{i-1} \cdot U_{i-1}^T] = \\ &= (1+q)\cdot U_{i-1}^* \cdot [D_{i-1} - \alpha^{-1} \cdot D_{i-1} \cdot U_{i-1}^T \cdot S_i^* \cdot S_i^T \cdot U_{i-1}^* \cdot D_{i-1}] \cdot U_{i-1}^T \\ &= (1+q)\cdot U_{i-1}^* \cdot [D_{i-1} - \alpha^{-1} \cdot B_{i-1} \cdot B_{i-1}^{\mathrm{H}}] \cdot U_{i-1}^T \qquad 6.6.31 \end{aligned}$$

where : $$B_{i-1}^{\mathrm{H}} = B_{i-1}^{*T} = (D_{i-1} \cdot U_{i-1}^T \cdot S_i^*)^{*T} = S_i^T \cdot U_{i-1}^* \cdot D_{i-1}$$

Now let the factor $\overline{U}$, $\overline{D}$ be :

$$\overline{U}_{i-1}^* \cdot \overline{D}_{i-1} \cdot \overline{U}_{i-1}^T = D_{i-1} - \alpha^{-1} \cdot B_{i-1} \cdot B_{i-1}^{\mathrm{H}} \qquad 6.6.32$$

Then eqn.6.6.31 becomes :

$$U_i^* \cdot D_i \cdot U_i^T = (1+q) \cdot [U_{i-1} \cdot \overline{U}_{i-1}]^* \cdot \overline{D}_{i-1} \cdot [U_{i-1} \cdot \overline{U}_{i-1}]^T \qquad 6.6.33$$

From eqn.6.6.33 it can be identified the terms :

$$U_i = U_{i-1} \cdot \overline{U}_{i-1} \quad ; \quad D_i = (1+q) \cdot \overline{D}_{i-1} \qquad 6.6.34$$

Thus the updated U-D factors are determined in terms of the U-D factor of $(D_{i-1} - \alpha^{-1} \cdot B_{i-1} \cdot B_{i-1}^{H})$ and U_{i-1}.

The full derivation of the square root Kalman algorithm is given in ref. [48], and for convenience is explained in Appendix E. The mechanism of this algorithm is summarised as follows :

At time (iT) we have the data-symbol vector (S_i) which is given by eqn.6.2.2, the received sample (r_i), and want to update the estimated SIR vector (P'_{i-1} to P'_i). The SRKA proceeds as follows :

With $\{s_i\}$, $\{p'_{i-1}\}$, $\{d_i\}$ and μ_{ij}, representing the elements of S_i , P'_{i-1}, diagonal of D_{i-1} and the upper triangular portion (excluding diagonal of U_{i-1}) respectively.

The computational procedure of the algorithm is given by the following equations :

$$\varepsilon_i = r_i - r'_i \qquad 6.6.35$$

$$f_1 = s_1^* \qquad 6.6.36$$

$$K_1 = d_1 \cdot f_1 \qquad 6.6.37$$

$$\alpha_1 = \eta + K_1 \cdot f_1 \qquad 6.6.38$$

For j = 2, L

$$f_j = \sum_{m=1}^{j-1} \mu_{m\,j} \cdot s_m^* + s_j^* \qquad 6.6.39$$

$$K_j = d_j \cdot f_j \qquad 6.6.40$$

$$\alpha_j = \alpha_{j-1} + K_j \cdot f_j \qquad 6.6.41$$

Next j

$$\gamma = 1/\alpha_1 \qquad 6.6.42$$

$$d_1 = d_1 \cdot h_q \cdot \eta \cdot \gamma \qquad 6.6.43$$

where $h_q = (1+q) = 1/w$

For j = 2, L

$$\beta = \alpha_{j-1} \qquad 6.6.44$$

$$\lambda_j = -f_j \cdot \gamma \qquad 6.6.45$$

$$\gamma = 1/\alpha_j \qquad 6.6.46$$

$$d_j = d_j \cdot h_q \cdot \beta \cdot \gamma \qquad 6.6.47$$

For m = 1, j-1

$$\beta_1 = \mu_{m\,j} \qquad 6.6.48$$

$$\mu_{m\,j} = \beta_1 + K_m^* \cdot \lambda_j \qquad 6.6.49$$

$$K_m = K_m + K_j \cdot \beta_1^* \qquad 6.6.50$$

Next m

Next j

$$e = \varepsilon_i \cdot \gamma \qquad 6.6.51$$

for j=1,2,. . . . . ,L

$$p'_j = p'_{j-1} + K_j \cdot e \qquad 6.6.52$$

Next j

where : $\{p'j\}$ are the elements of the estimated SIR vector (P'_j). All quantities are complex except w, η, h_q, γ, β, $\{ dj \}$ $(1<= j <= L)$ and $\{\alpha j\}$ $(1<= j <= L)$.

At the start of the process (at $t=0$)

$$d_j = 1.0 \qquad \text{for } j = 1,2,. . . .,L \qquad 6.6.53$$

$$\mu_{mj} = 0.0 + 0.0 \qquad \text{for } m=1,2,.....,(L-1) \qquad 6.6.54$$

$$\text{and} \qquad j= m+1, m+2,. . . . .,L$$

The computational requirement of the SRK estimator is $(6L^2+11L)$ real multiplications and $(6L^2+6L)$ real additions and (L) divisions with $(L^2+8L+14)$ storage locations [48]. Equations (6.6.35) to (6.6.54) represent the full SRK algorithm.

6.6.2 ESTIMATOR 5 " SRK Estimator with Prediction "

This arrangement is a simple modification of estimator 4, where now the degree-1 prediction is incorporated with SRKA to track the variation of the HF channel.

Tests have been shown [114] that when a channel changes very rapidly, and the SIR of the channel has more than (10) components, i.e. $L > 10$, the SRKA does not give satisfactory operation especially at a high signal to noise ratio for the reason that minimisation of mean square error is done for the channel model that obeys a first order Markov process [27], that (P_i) is a function of (P_{i-1}) only. But the nature of HF channel suggests that (P_i) may effectively be a function of previous vectors as high as a 2 to 5 intervals [61,64], i.e. a (2 to 5)th order Markov process, as in the case considered in this study. It has been found by [61,64] that one step prediction is adequate for HF channel applications. Therefore the performance of SRKA can be improved by incorporating a degree-1 prediction to track the most rapidly fading HF channel.

Estimator 5 uses the SRKA given by estimator 4 to update the estimate of the SIR of the channel given by (P'_i) in eqn.5.6.52 as :

$$p'_i = p'_{i,i-1} + \varepsilon \cdot K_i \qquad 6.6.55$$

where (P'_i) is determined by eqns.6.6.35 to 6.6.62. The estimator next forms an estimate of the error in the prediction of $(P'_{i,i-1})$, which is given by :

$$e'_i = P'_i - P'_{i,i-1} \qquad 6.6.56$$

where $(P'_{i,i-1})$ is the one step prediction of (P_i), bearing in mind that the actual error in $(P'_{i,i-1})$ is $(P_i - P'_{i,i-1})$. From eqn.6.6.55 and 6.6.56, then :

$$e'_i = \varepsilon \cdot K_i = \Delta_i \qquad 6.6.57$$

(Δ_i) is now called the correction matrix. The error (e'_i) is next, fed through the polynomial filter to give the prediction of (P_{i+1}) [61,64], which is given by :

$$\begin{aligned} P''_{i+1,i} &= P''_{i,i-1} + \lambda_1 \cdot e'_i \\ P'_{i+1,i} &= P'_{i,i-1} + P''_{i+1,i} + \lambda_2 \cdot e'_i \end{aligned} \qquad 6.6.58$$

where $(P'_{i+1,i})$ is the degree-1 least square fading memory prediction of (P_{i+1}), $(L+1)$-components row vector, and $(P''_{i+1,i})$ is a prediction of the rate of change with (i) of (P_{i+1}).The symbols (λ_1, λ_2) in eqn.6.6.58 are positive real-valued scalar constants. $(\lambda_1$ and $\lambda_2)$ are optimised here, by means of the computer simulation to minimise the error in prediction of the channel response. The one-step prediction used by estimator 5 eqn.6.6.58, is slightly different from that used by estimators 2 and 3, where $(\lambda_1$ and $\lambda_2)$ in eqn.6.6.58 no longer bear any relationship.

At start of the estimation process the estimator sets :

$$P''_{1,0} = [0.0] \quad , \quad P'_{1,0} = [0.0] \qquad 6.6.59$$

The full algorithm of estimator 5 is given by eqns. 6.6.35-6.6.59. Tests have shown some useful improvement in the performance of estimator 5, where eqn.6.6.58 allows greater flexibility and improved performance of the estimator over the whole range of signal-to-noise ratios. Tests have not shown any kind of instability in the algorithm. The performance of this estimator is given at the end of this chapter. A comparison of various estimators is also presented.

6.7 SIMULATION RESULTS AND DISCUSSION

Extensive series of computer simulation tests have been carried out to assess the performance of various channel estimators described in section (6.3-6.6), in order to select the best estimator for a combined system (Modem).

The model of the data transmission system used is that shown in Fig 6.2.1 where a QPSK signal is employed to transmit the information of a rate *4800 bits/sec*, over fading multipath channels 1, 2 and 3, whose parameters are given in Chapter 2. All assumptions

made here are given in section 6-2. The results of the tests are given in Tables 6.7.1-6.7.3, and Figs. 6.7.1-6.7.10.

The performance of the estimators is measured in terms of the average mean-square error in the estimation of the SIR of the channel, where the latter is measured in *dB* and is given by :

$$\zeta_1 = 10 \cdot \log_{10} \cdot [\frac{1}{57000} \cdot \sum_{i=1641}^{58640} |P_i - P'_{i,i-1}|^2] \quad \text{in } dB. \qquad 6.7.1$$

where ζ_1 is called the average mean-square estimation error. For comparison purposes another measurement is given by :

$$\zeta_2 = 10 \cdot \log_{10} \cdot [\frac{1}{57000} \cdot \sum_{i=1641}^{58640} \frac{|P_i - P'_{i,i-1}|^2}{|P_i|^2}] \quad \text{in } dB. \qquad 6.7.2$$

where ζ_2 is called the average mean square normalised estimation error, and is a measure of the normalised or relative error in $P'_{i,i-1}$, which is given in *dB*. $|P_i|$ is the unitary length of the vector P_i. The signal-to-noise ratio (ψ) in all tests is given by eqn.5.6.1 in *dB*, as :

$$\psi = 10 \log_{10} \frac{E_b}{0.5 N_0} \qquad 6.7.3$$

where E_b is the average transmitted signal energy per bit at the input to the HF channel Fig.6.2.1, and $N_0/2$ is the two-sided power spectral density of the additive white Gaussian noise at the output of transmission path. In all tests a given number of (*58640*) received samples are involved to evaluate ζ_1 and ζ_2, where the first (*1640*) received samples are taken to be the warm-up period in any test carried out here, and are ignored, so that the fading and noise processes can be stabilised and the initial transient behaviour of the estimator does not affect ζ_1 and ζ_2. Thus ζ_1 and ζ_2 give a measure of the steady-state performance of the estimator. In all tests, (*L+1*), the total number of components in the overall sampled impulse response (SIR) is taken to be *20*.

In each Table (6.7.1-6.7.3), the adjustable parameters, such as b, λ, w, λ_1 and λ_2 are here optimised at each value of SNR (ψ) as far as reasonably possible to minimise ζ_1 in eqn.6.7.1. The other parameters, such as d_1, d_0, η, are kept fixed. For estimator 3 the parameter d_0 is set to *0.003* during all tests over the three channels and by optimising λ, good results are obtained as shown in the Tables and Figs.67.1-6.7.2. It should be also noted from the tables that the value of 'λ' stays approximately the same for every individual channel at the given value of SNR, and the adjustment of λ is not critical. This makes the system more flexible to implement in practice. In estimator 4 the parameters w, η are fully examined in ref. [48]. During the tests, all parameters do not create any

instability in the algorithm. Figs.6.7.1-6.7.2 show the performance of all estimators, and a short symbol is written beside each plot, where for example *EST3WP* refers to estimator 3 with degree-1 prediction. Also the performance of estimator 3 without any prediction is included in the tables and the graphs in Figs.6.7.1-6.7.2, and is denoted as (*EST3NP*), for comparison purposes. A wide range of SNR (ψ=*10-65 dB*) has been used in the tests. At low values of SNR up to *35dB*, a significant number of errors in detection of the received data symbols is likely to be caused by the additive noise mainly and fading, whereas at high SNR up to *65dB*, the fading predominates over the noise.

Tables 6.7.1-6.7.3 and Figs.6.7.1-6.7.2 show the performance of the various estimators starting with LMS algorithm and finishing with SRK algorithm when operating over the three HF channels. It can be seen from the tables and figures, that the best performance of the estimators is given by estimator 3 and estimator 5, with degree-1 prediction. The poorest performance is given by estimator 1 (LMS) and estimator 4 (SRK) without prediction, especially at high SNR because minimisation of ζ_1 is done for a channel that is time-invariant or varies with time very slowly, which means that it obeys a first order Markov process, that is P_i is a function of P_{i-1} only. But the nature of HF channels suggests that P_i may effectively be a function of previous vectors as high as 2-5 intervals away [27,60]. It is clear from the tables and figures, that employing degree-1 prediction improves the performance of the estimators over the whole range of SNR. The adaptive step-size estimator (estimator 3) gives good results and is very close to estimator 5 (SRK with degree-1 prediction), over the whole range of SNR. At low SNR estimator 3 is the best, which is of the greatest practical importance. The improvement in estimator 3 is due to changing the step-size of the LMS algorithm from remaining fixed during the estimation process to being adaptively adjusted. This improves the ability of the algorithm to track time varying channels. In other words estimator 3 is better able to correct an error in $p'_{i,i-1}$ caused by an acceleration in P_i. Furthermore the computational complexity required by estimator 3 is much less than that required by the SRK estimator (estimator 4 and 5). The channel estimators, listed in order of complexity and starting with the simple estimator, are estimator 1, 2, 3, 4, and 5. Therefore the most promising as the compromise in terms of performance and complexity, estimator 3 appears to be the preferred system.

Further tests have been carried out on estimator 3 when operating over channel 1 and 2, and the results of computer simulation are presented in Figs.6.7.3-6.7.9. Fig.6.7.3 and Fig.6.7.4 show the plots of computer simulation tests over channel 1 and 2 for the evaluation of the most optimum value of the parameter λ for estimator 3, that would result in the best mean-square error values ζ at different SNR. Figs.6.7.3-6.7.4 confirm that, in these particular channels, the adjustable parameter λ for estimator 3 is not unduly critical at the given value of SNR, so that the correct adjustment of the estimator should be achievable in practice. During these tests all adjustable parameters such as d_0, d_1, d_2 are kept fixed at their optimum value.

Figs.6.7.5-6.7.9 show a segment of *50* sample intervals of the actual and estimated SIR of the overall linear base-band channel 1 and 2 in three-dimensional graphs, where the X-axis represents the number of components in the SIR of the channel (*L+1*). The Y-axis represents the time instant *i*, where *i* is taken to be *1,2,3,.....,50* sample intervals. The Z-axis represents the amplitude variation of the SIR of the channel.

The estimated SIR is obtained using estimator 3 with degree-1 prediction at SNR equal to *10* and *35dB*. Comparing Figs.6.7.7-6.7.8 with Fig.6.7.5 it can be seen that, as the SNR increases the estimated SIR reaches the actual SIR. The figures also indicate that, a very small distortion in the amplitude of the SIR, is caused by the estimator. Therefore estimator 3 with degree-1 prediction gives quite good performance over both channels, even at low SNR under the condition that perfect detection is assumed. It is clear from Figs.6.7.7-6.7.9, the distortion caused by the estimator is negligible and the estimator gives a very accurate estimate of the channel response.

Finally, the start-up convergence comparison between the LMS algorithm and SRK algorithm is presented in Fig.6.7.10, when both algorithms operate over channel 1 at *SNR=25dB*. The starting conditions of SRK algorithm are given in Appendix D, and the estimated vector (P'_i) has all zero components at time *i=0*. Each of the curves in Fig.6.7.10 is the average of *5* separate runs, each having a different sequence of Gaussian noise samples and data symbols, but with the same channel characteristics. The adjustable parameters for both algorithms are set to their optimum value for the corresponding steady-state, at the given value of SNR. The results in Fig.6.7.10 show how the error :

$$\zeta_i = 10\log_{10}(E\{|P_i - P'_i|^2\}) \qquad 6.7.4$$

in P_i varies with *i*. In eqn.6.7.4 $E\{|P_i - P'_i|^2\}$ is the measured value of the ensemble average of $|P_i - P'_i|^2$ over the *5* runs. The estimators start with no prior knowledge of the channel, therefore the estimators are not adjusted to achieve the most rapid convergence possible. It is clear from Fig.6.7.10 the SRK algorithm reaches the steady-state more rapidly. The LMS algorithm gives quite good convergence compared to the SRK, but the former LMS is less complex and very simple to implement.

This chapter concludes that, the most cost effective of the five estimators, for the given application studied here, is estimator 3 with degree-1 prediction in terms of performance and complexity. The good results from estimator 3 are due to its ability to track the acceleration in P_i more accurately than other estimators. Also in terms of complexity, it's simple to implement, where all parameters except λ can be set fixed and the value of λ is not critical for the stability of the algorithm. Therefore estimator 3 is a good candidate for a combined detector-estimator system. In the next chapter we will study the system when the detector, adaptive filter and estimator work together over a fading channel.

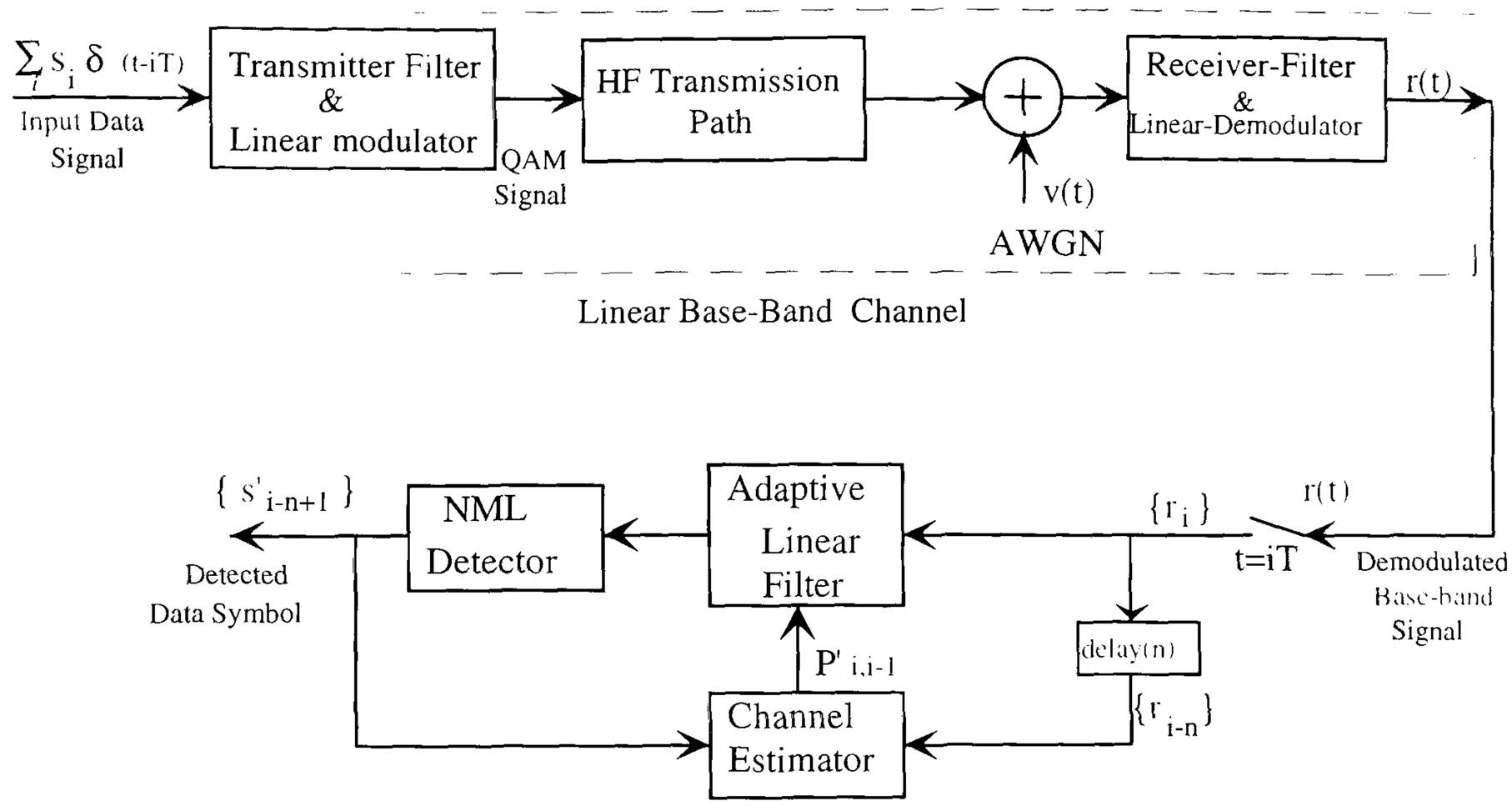


Fig.6.2.1 Model of Data Transmission System.

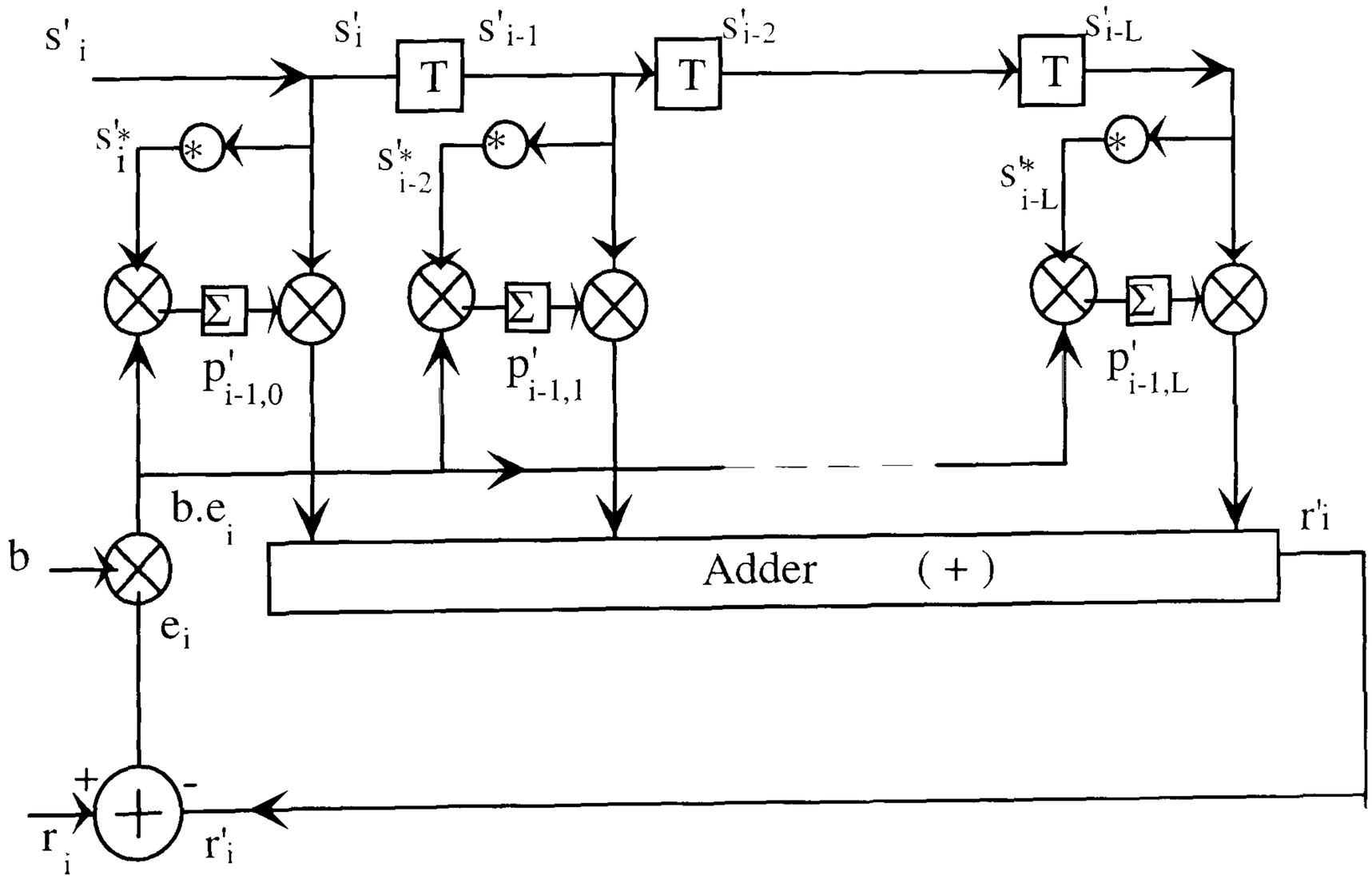


Fig.6.3.1 : Feedforward Estimator (Estimator 1).

SNR (dB)	B	ζ_1 (dB)	ζ_2 (dB)	SNR(dB)	B	λ	ζ_1 (dB)	ζ_2 (dB)
10	0.01	-13.98	-12.6	10	0.24	0.985	-14.24	-12.97
20	0.018	-19.64	-18.28	20	0.24	0.976	-21.58	-20.33
30	0.028	-23.99	-22.63	30	0.24	0.966	-28.5	-27.3
35	0.032	-25.22	-23.87	35	0.24	0.96	-31.6	-30.45
40	0.033	-25.832	-24.5	40	0.24	0.954	-34.3	-33.2
50	0.034	-26.15	-24.82	50	0.24	0.946	-37.77	-36.85
60	0.034	-26.2	-24.9	65	0.24	0.942	-38.88	-38.04

a- performance of estimator 1, and b-performance of estimator 2, (over channel1).

SNR (dB)	B	do	d1	d2	ζ_1 (dB)	ζ_2 (dB)
10	0.028	0.003	10^{-6}	α	-15.44	-14.34
20	0.052	0.003	10^{-6}	α	-21.65	-20.28
30	0.09	0.003	10^{-6}	α	-26.42	-25.08
35	0.118	0.003	10^{-6}	α	-28.02	-26.88
40	0.120	0.003	10^{-6}	α	-29.13	-27.81
50	0.131	0.003	10^{-6}	α	-29.65	-28.51
60	0.128	0.003	10^{-6}	α	-29.9	-28.6

c- performance of estimator 3 (no prediction), (over channel1).

SNR (dB)	λ	do	d1	d2	ζ_1 (dB)	ζ_2 (dB)
5	0.993	0.003	10^{-6}	α	-12.59	-11.24
10	0.99	0.003	10^{-6}	α	-16.1	-14.76
15	0.987	0.003	10^{-6}	α	-19.66	-18.34
20	0.983	0.003	10^{-6}	α	-23.25	-21.96
25	0.979	0.003	10^{-6}	α	-26.78	-25.48
30	0.974	0.003	10^{-6}	α	-30.22	-28.94
35	0.969	0.003	10^{-6}	α	-33.5	-32.21
40	0.963	0.003	10^{-6}	α	-36.5	-35.26
50	0.954	0.003	10^{-6}	α	-41.02	-39.82
60	0.951	0.003	10^{-6}	α	-42.73	-41.52
65	0.949	0.003	10^{-6}	α	-42.88	-41.9

d-performance of estimator 3 (with prediction), (over channel1).

SNR(dB)	η	w	$\zeta 1$ (δB)	ζ_2 (dB)
10	0.01	0.978	-14.18	-12.83
20	0.01	0.952	-20.08	-18.74
30	0.01	0.9	-25.07	-23.74
40	0.01	0.822	-27.87	-26.54
50	0.01	0.788	-28.5	-27.18
60	0.01	0.788	-28.6	-27.3

e- performance of estimator 4 (SRK), (over channel1).

SNR(dB)	η	w	λ1	λ2	ζ_1 (dB)	ζ_2 (dB)
5	0.01	0.988	0.005	0.999	-10.99	-9.75
10	0.01	0.980	0.007	0.78	-14.77	-13.52
15	0.01	0.974	0.009	0.70	-18.64	-17.4
20	0.01	0.972	0.013	0.70	-22.48	-21.25
25	0.01	0.962	0.015	0.70	-26.3	-25.1
30	0.01	0.952	0.018	0.70	-30.0	-28.82
35	0.01	0.942	0.024	0.7	-33.6	-32.43
40	0.01	0.928	0.028	0.70	-37.07	-35.92
45	0.01	0.905	0.034	0.70	-40.36	-39.22
50	0.01	0.88	0.04	0.70	-43.4	-42.28
60	0.01	0.88	0.066	0.88	-48.88	-47.8
65	0.01	0.84	0.07	0.86	-50.76	-49.74

f- performance of estimator 5 (SRK-D-1), (over channel1).

Table 6.7.1 : Performance of Estimators over Channel 1.

NR(dB)	B	ζ_1 (dB)	ζ_2 (dB)	SNR(dB)	B	λ	ζ_1 (dB)	ζ_2 (dB)
10	0.01	-13.76	-10.98	10	0.24	0.985	-14.13	-11.47
20	0.018	-19.37	-16.60	20	0.24	0.976	-21.5	-18.85
30	0.028	-23.58	-20.82	30	0.24	0.966	-28.36	-25.84
35	0.032	-24.73	-21.99	35	0.24	0.96	-31.46	-28.98
40	0.033	-25.28	-22.56	40	0.24	0.954	-34.15	-31.73
50	0.034	-25.7	-22.85	50	0.24	0.946	-37.55	-35.31
60	0.034	-25.72	-22.88	65	0.24	0.942	-38.55	-36.40

a- performance of estimator 1, and b-performance of estimator 2, (over channel2).

SNR (dB)	B	do	d1	d2	ζ_1 (dB)	ζ_2 (dB)
10	0.032	0.003	10^{-6}	α	-16.1	-13.65
20	0.057	0.003	10^{-6}	α	-21.75	-19.06
25	0.072	0.003	10^{-6}	α	-24.1	-21.36
30	0.092	0.003	10^{-6}	α	-26.0	-23.43
35	0.128	0.003	10^{-6}	α	-27.62	-25.58
40	0.124	0.003	10^{-6}	α	-28.53	-26.05
50	0.128	0.003	10^{-6}	α	-29.2	-26.6
60	0.128	0.003	10^{-6}	α	-29.0	-26.6

c- performance of estimator 3 (no prediction), (over channel2).

SNR (dB)	λ	do	d1	d2	ζ_1 (dB)	ζ_2 (dB)
5	0.992	0.003	10^{-6}	α	-12.76	-10.15
10	0.99	0.003	10^{-6}	α	-16.38	-13.6
15	0.987	0.003	10^{-6}	α	-19.98	-17.12
20	0.983	0.003	10^{-6}	α	-23.57	-20.72
25	0.979	0.003	10^{-6}	α	-27.13	-24.24
30	0.974	0.003	10^{-6}	α	-30.58	-27.72
35	0.969	0.003	10^{-6}	α	-33.92	-30.98
40	0.963	0.003	10^{-6}	α	-36.99	-34.06
50	0.954	0.003	10^{-6}	α	-41.67	-38.55
60	0.949	0.003	10^{-6}	α	-43.41	-40.52
65	0.949	0.003	10^{-6}	α	-43.62	-40.65

d-performance of estimator 3 (with prediction), (over channel2).

SNR(dB)	η	w	$\zeta 1$ (δB)	ζ_2 (dB)
10	0.01	0.978	-13.94	-11.14
20	0.01	0.952	-19.80	-17.0
30	0.01	0.9	-24.70	-21.90
40	0.01	0.822	-27.36	-24.6
50	0.01	0.788	-27.94	-25.21
60	0.01	0.788	-28.01	-25.3

e- performance of estimator 4 (SRK), (over channel2).

SNR(dB)	η	w	λ1	λ2	ζ_1 (dB)	ζ_2 (dB)
5	0.01	0.988	0.006	0.999	-10.88	-8.13
10	0.01	0.980	0.007	0.78	-14.70	-11.93
15	0.01	0.974	0.009	0.70	-18.57	-15.81
20	0.01	0.972	0.013	0.70	-22.44	-19.70
25	0.01	0.962	0.015	0.70	-26.24	-23.52
30	0.01	0.952	0.018	0.70	-29.92	-27.23
35	0.01	0.942	0.024	0.7	-33.52	-30.83
40	0.01	0.928	0.028	0.70	-36.98	-34.31
45	0.01	0.905	0.034	0.70	-40.26	-37.61
50	0.01	0.88	0.04	0.70	-43.28	-40.67
60	0.01	0.88	0.066	0.88	-48.67	-46.18
65	0.01	0.84	0.07	0.86	-50.50	-48.2

f- performance of estimator 5 (SRK-D-1), (over channel2).

Table 6.7.2 : Performance of Estimators over Channel 2.

NR(dB)	B	ζ_1 (dB)	ζ_2 (dB)	SNR(dB)	B	λ	ζ_1 (dB)	ζ_2 (dB)
10	0.008	-16.16	-13.27	10	0.24	0.99	-17.32	-14.71
20	0.014	-22.28	-19.37	20	0.24	0.985	-24.99	-22.41
30	0.025	-27.6	-24.66	30	0.24	0.978	-32.50	-29.94
35	0.03	-29.65	-26.65	35	0.24	0.972	-36.15	-33.56
40	0.034	-31.04	-27.98	40	0.24	0.966	-39.65	-37.1
50	0.038	-32.1	-28.95	50	0.24	0.954	-45.97	-43.45
60	0.038	-32.21	-29.08	65	0.24	0.938	-51.5	-49.3

a- performance of estimator 1, and b-performance of estimator 2, (over channel3).

SNR (dB)	B	do	d1	d2	ζ_1 (dB)	ζ_2 (dB)
10	0.027	0.003	10^{-6}	α	-17.88	-15.53
20	0.042	0.003	10^{-6}	α	-24.35	-21.57
30	0.076	0.003	10^{-6}	α	-30.06	-27.13
35	0.094	0.003	10^{-6}	α	-32.46	-29.4
40	0.117	0.003	10^{-6}	α	-34.04	-31.17
50	0.131	0.003	10^{-6}	α	-36.1	-32.81
60	0.135	0.003	10^{-6}	α	-36.44	-33.32

c- performance of estimator 3 (no prediction), (over channel3).

SNR (dB)	λ	do	d1	d2	ζ_1 (dB)	ζ_2 (dB)
5	0.995	0.003	10^{-6}	α	-15.4	-12.82
10	0.993	0.003	10^{-6}	α	-19.1	-16.56
15	0.991	0.003	10^{-6}	α	-22.92	-20.37
20	0.989	0.003	10^{-6}	α	-26.75	-24.1
25	0.986	0.003	10^{-6}	α	-30.5	-27.86
30	0.982	0.003	10^{-6}	α	-34.3	-31.63
35	0.980	0.003	10^{-6}	α	-38.03	-35.05
40	0.974	0.003	10^{-6}	α	-41.63	-38.85
50	0.963	0.003	10^{-6}	α	-48.41	-45.57
60	0.95	0.003	10^{-6}	α	-53.90	-51.2
65	0.949	0.003	10^{-6}	α	-55.78	-52.514

d-performance of estimator 3 (with prediction), (over channel3).

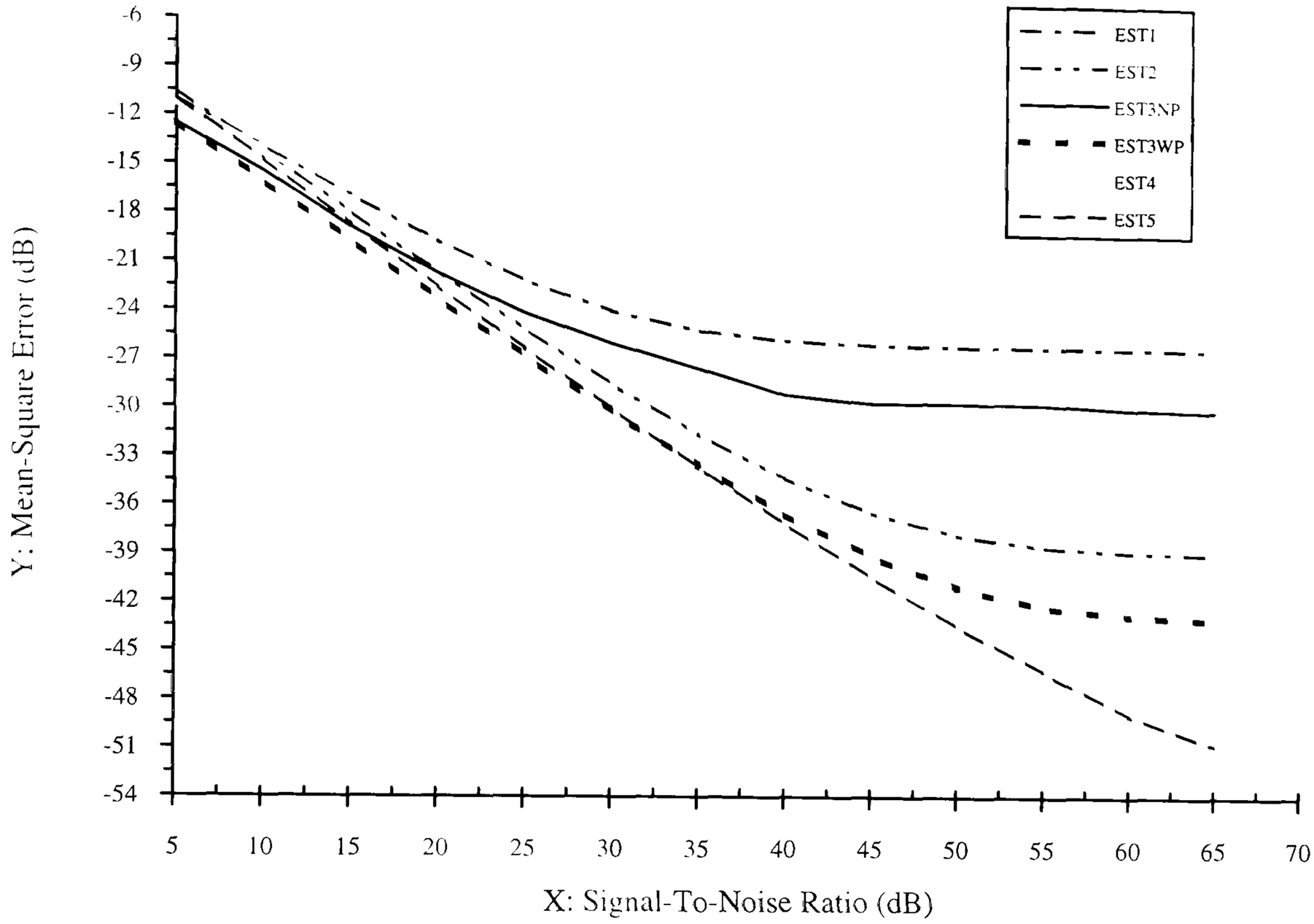
SNR(dB)	η	w	ζ1 (δB)	ζ_2 (dB)
10	0.01	0.984	-16.55	-13.65
20	0.01	0.966	-22.76	-19.86
30	0.01	0.926	-28.47	-25.58
40	0.01	0.838	-32.84	-29.91
50	0.01	0.784	-34.72	-31.7
60	0.01	0.784	-35.01	-31.96

e- performance of estimator 4 (SRK), (over channel3).

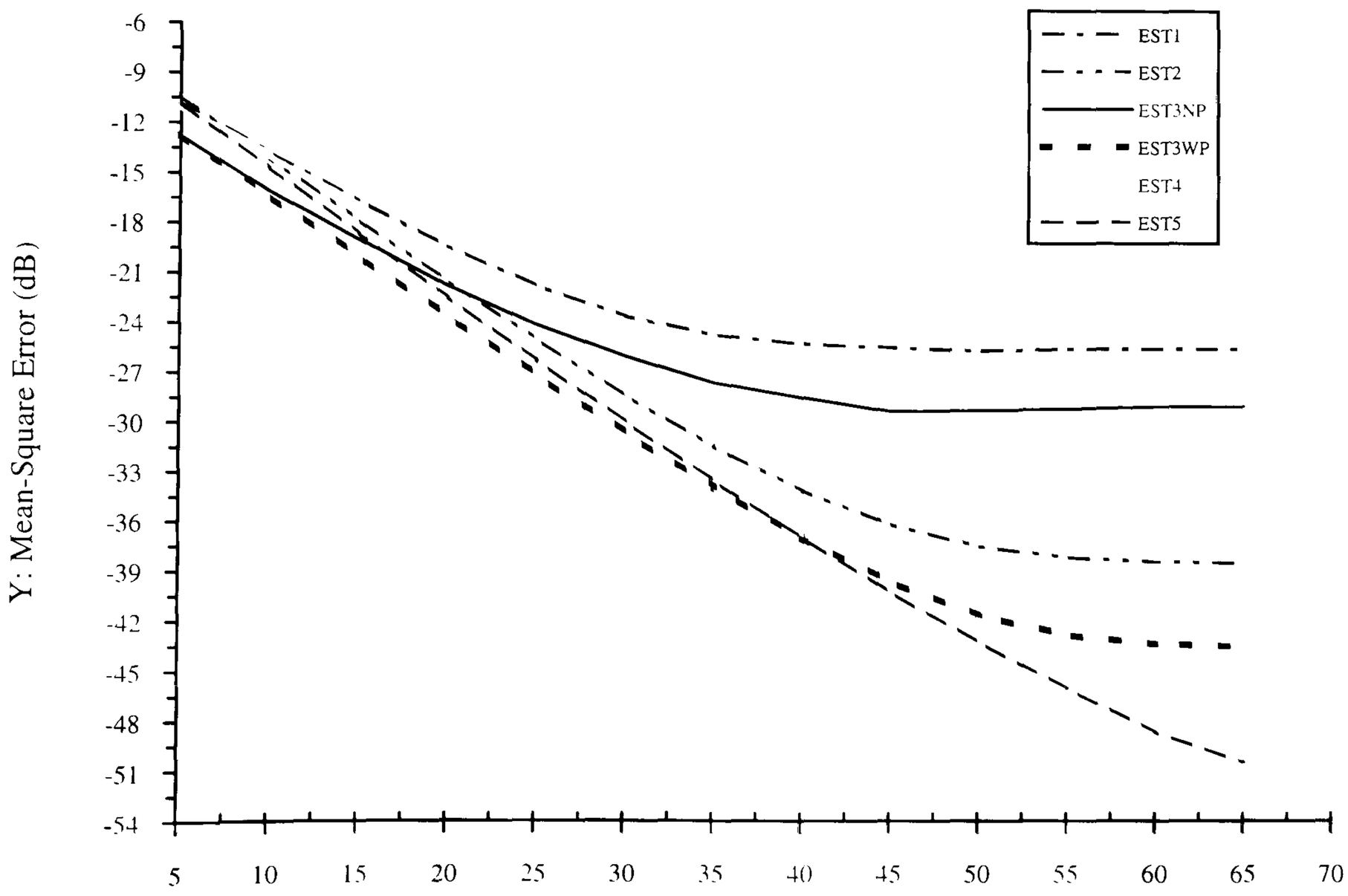
SNR(dB)	η	w	λ1	λ2	ζ_1 (dB)	ζ_2 (dB)
5	0.01	0.994	0.005	0.999	-13.85	-11.31
10	0.01	0.992	0.006	0.78	-17.69	-15.14
15	0.01	0.988	0.007	0.7	-21.66	-19.05
20	0.01	0.986	0.009	0.7	-25.54	-22.96
25	0.01	0.984	0.012	0.7	-29.4	-26.81
30	0.01	0.980	0.014	0.70	-33.3	-30.75
35	0.01	0.974	0.018	0.7	-37.13	-34.55
40	0.01	0.968	0.022	0.70	-40..80	-38.25
45	0.01	0.956	0.024	0.70	-44.57	-42.05
50	0.01	0.948	0.032	0.70	-48.10	-45.57
60	0.01	0.936	0.054	0.88	-54.85	-52.22
65	0.01	0.918	0.062	0.86	-57.89	-55.50

f- performance of estimator 5 (SRK-D-1), (over channel3).

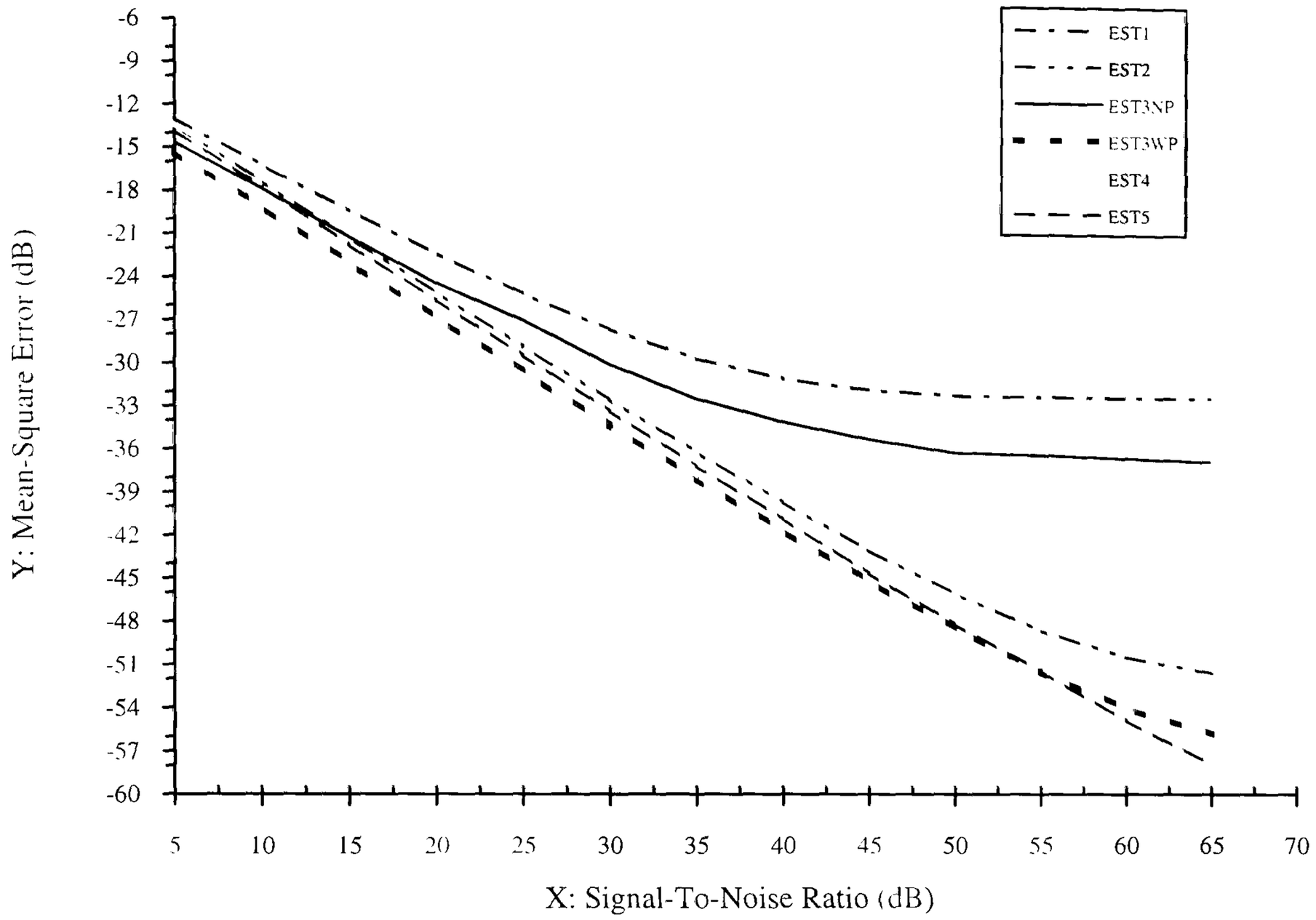
Table 6.7.3 : Performance of Estimators over Channel 3.



(ζ_1 with ψ)
a-over channel1.

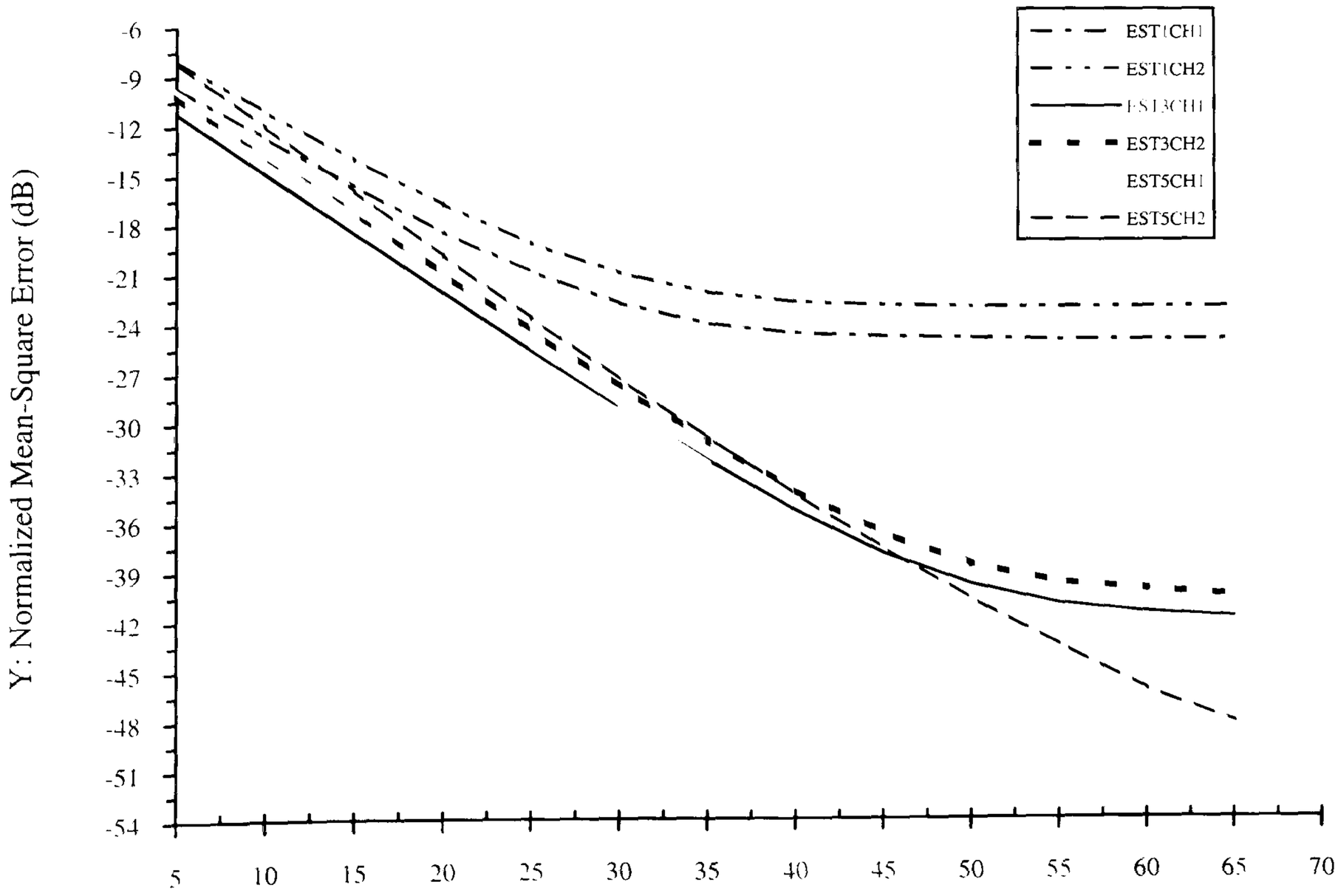


(ζ_1 with ψ)
b-over channel2.



c-over channel3. (ζ_1 with ψ)

Fig.6.7.1 Comparision of Estimators operating over Channel1, Channel2 and Channel3.



(ζ_2 with ψ)

Fig.6.7.2 Comparision of Estimator1,3 and 5 operating over Channel1 and Channel2.

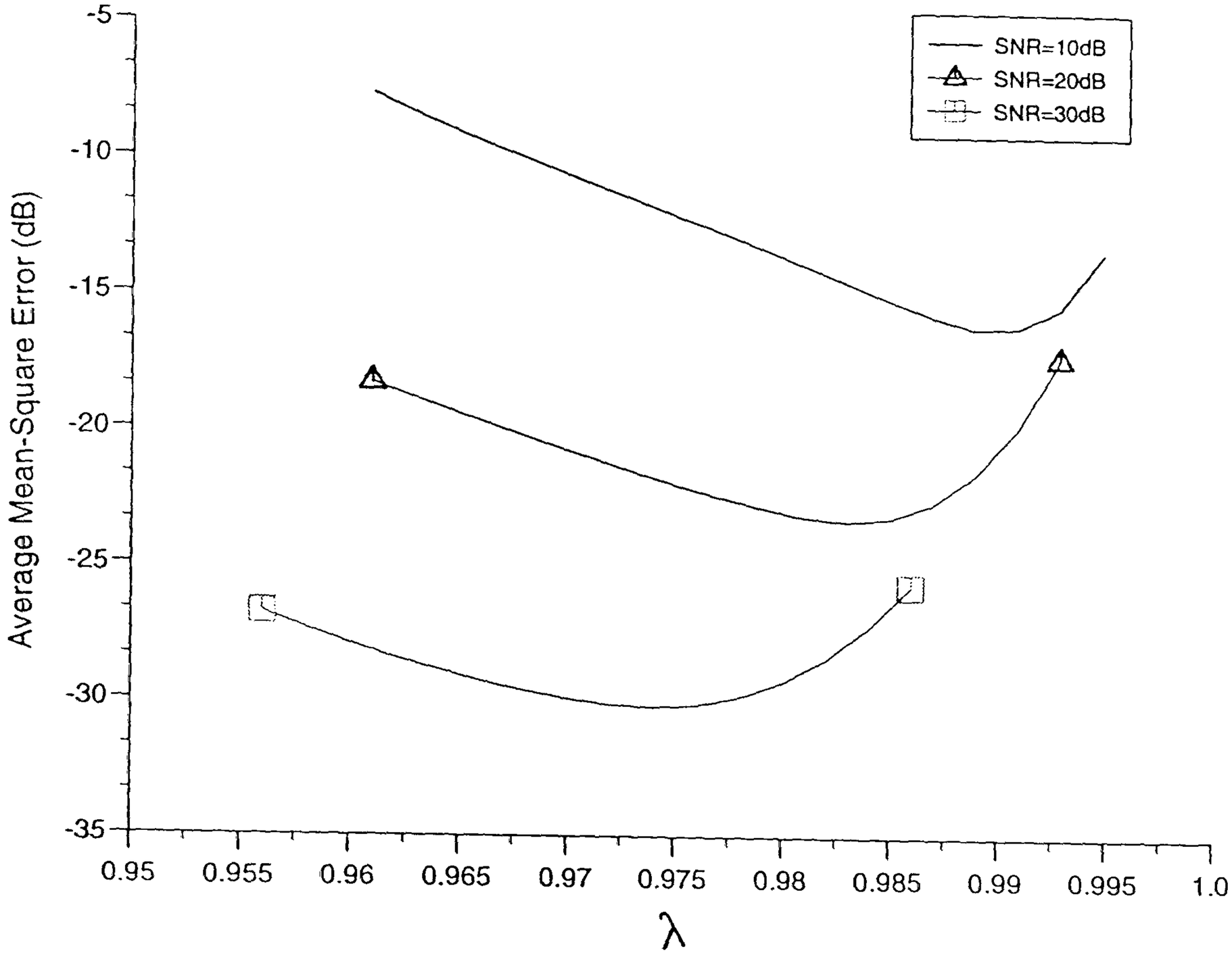


Fig.6.7.3 Variation of 'λ' with Mean-square Error 'ζ_1' for Estimator 3.

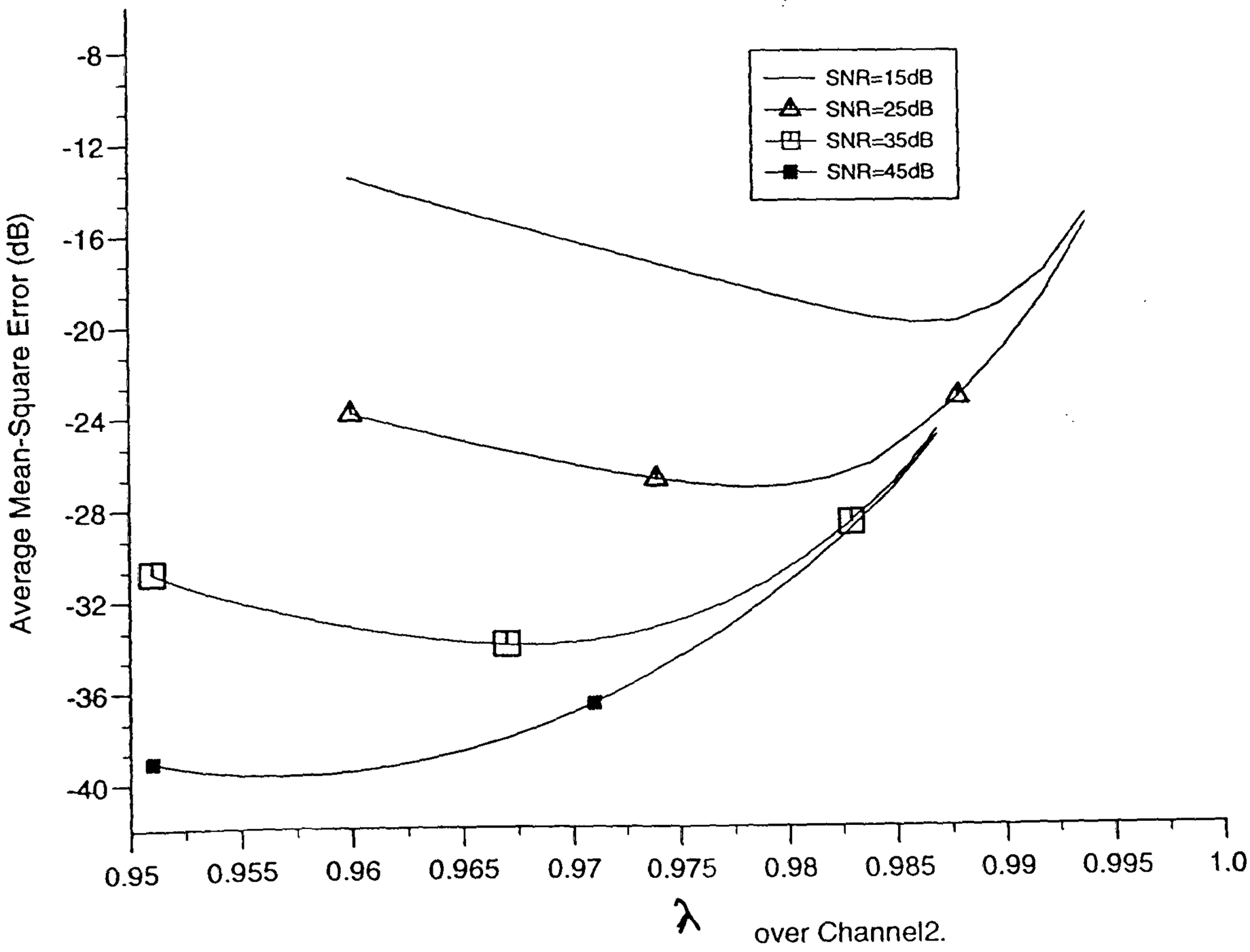
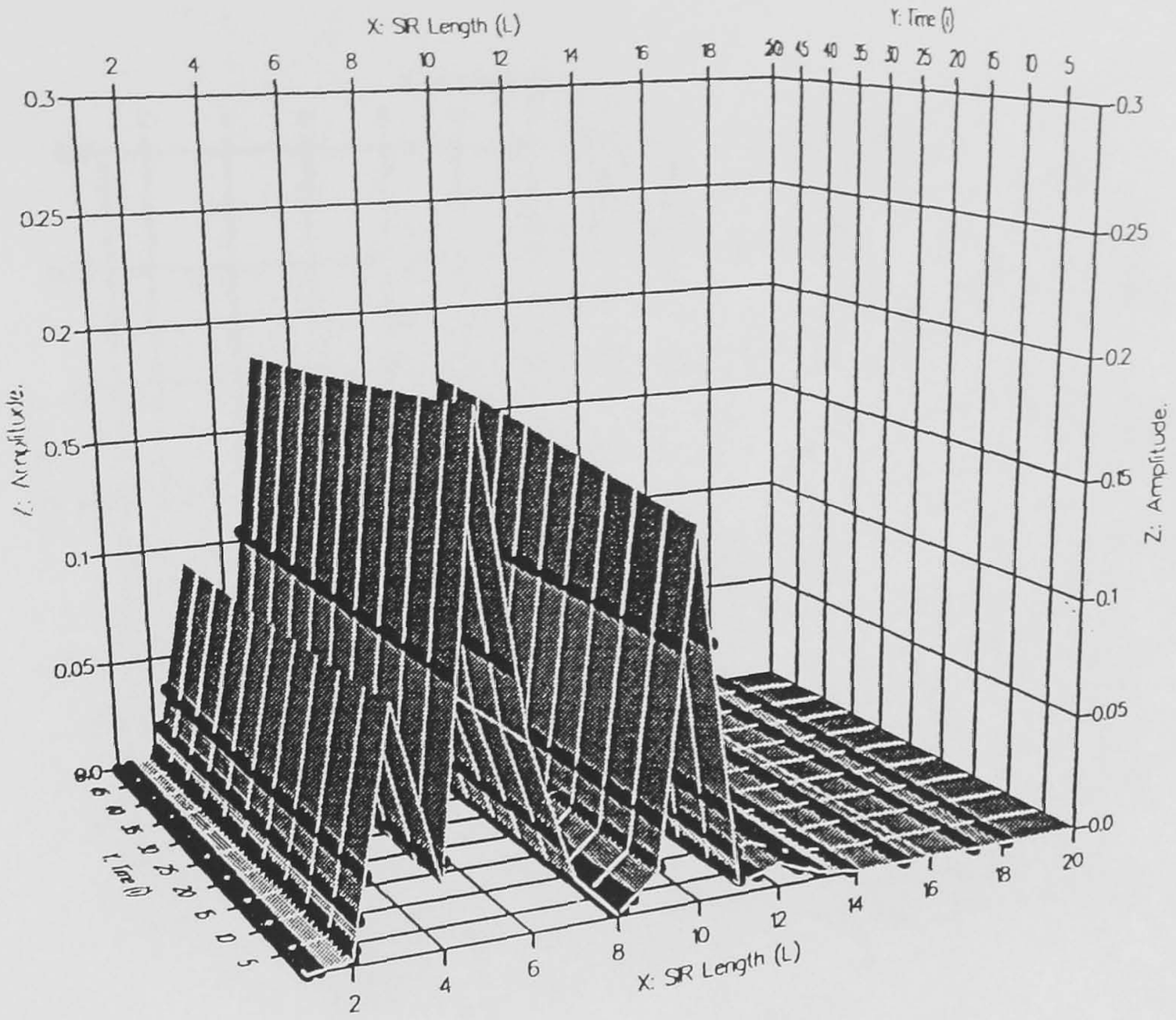
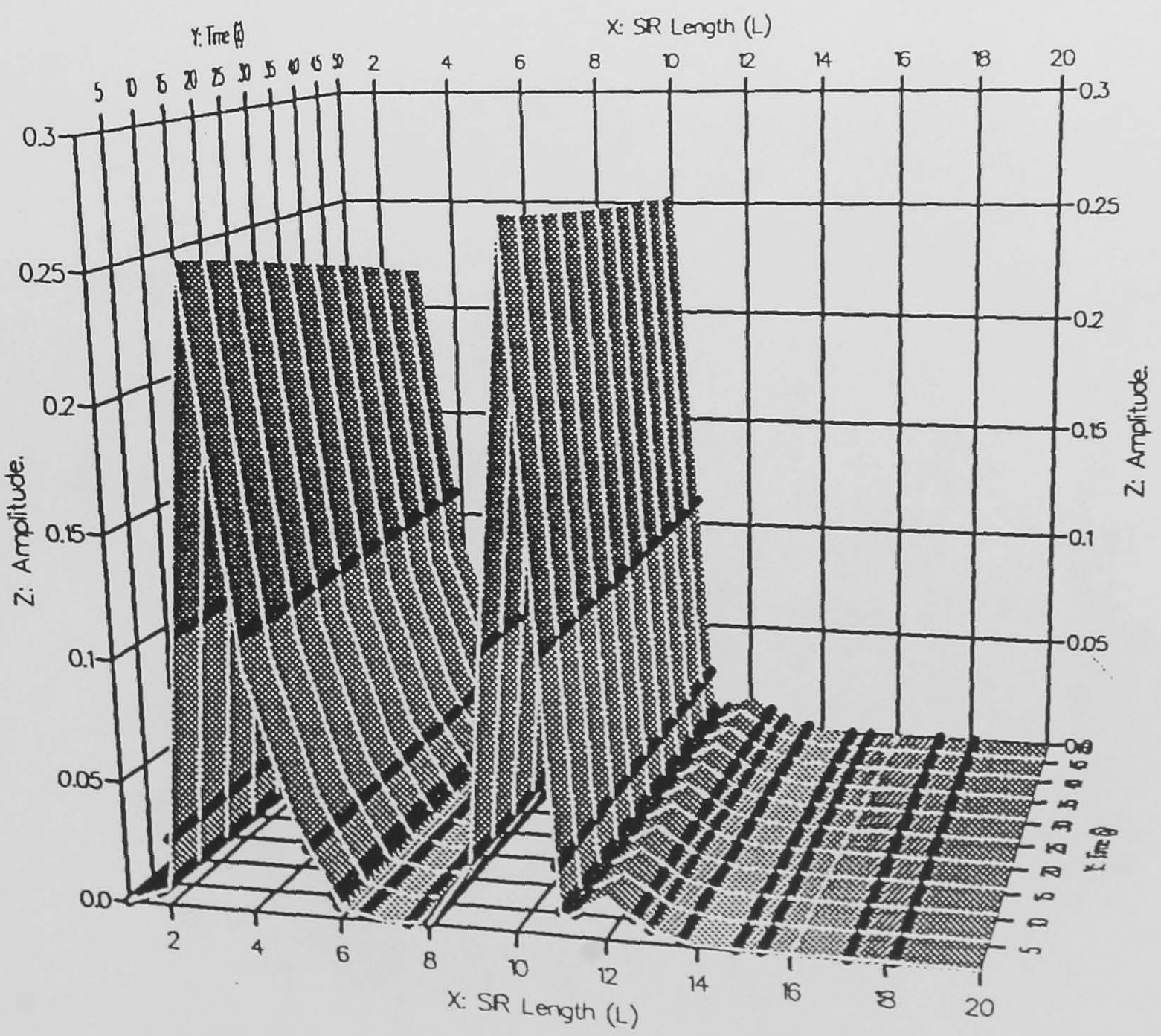


Fig.6.7.4 Variation of 'λ' with Mean-square Error 'ζ_1' for Estimator 3.



(i=9600-9650).

Fig. 6.7.5 A segment of Actual SIR of Channel1.



(i=9600-9650).

Fig. 6.7.6 A segment of Actual SIR of Channel2.

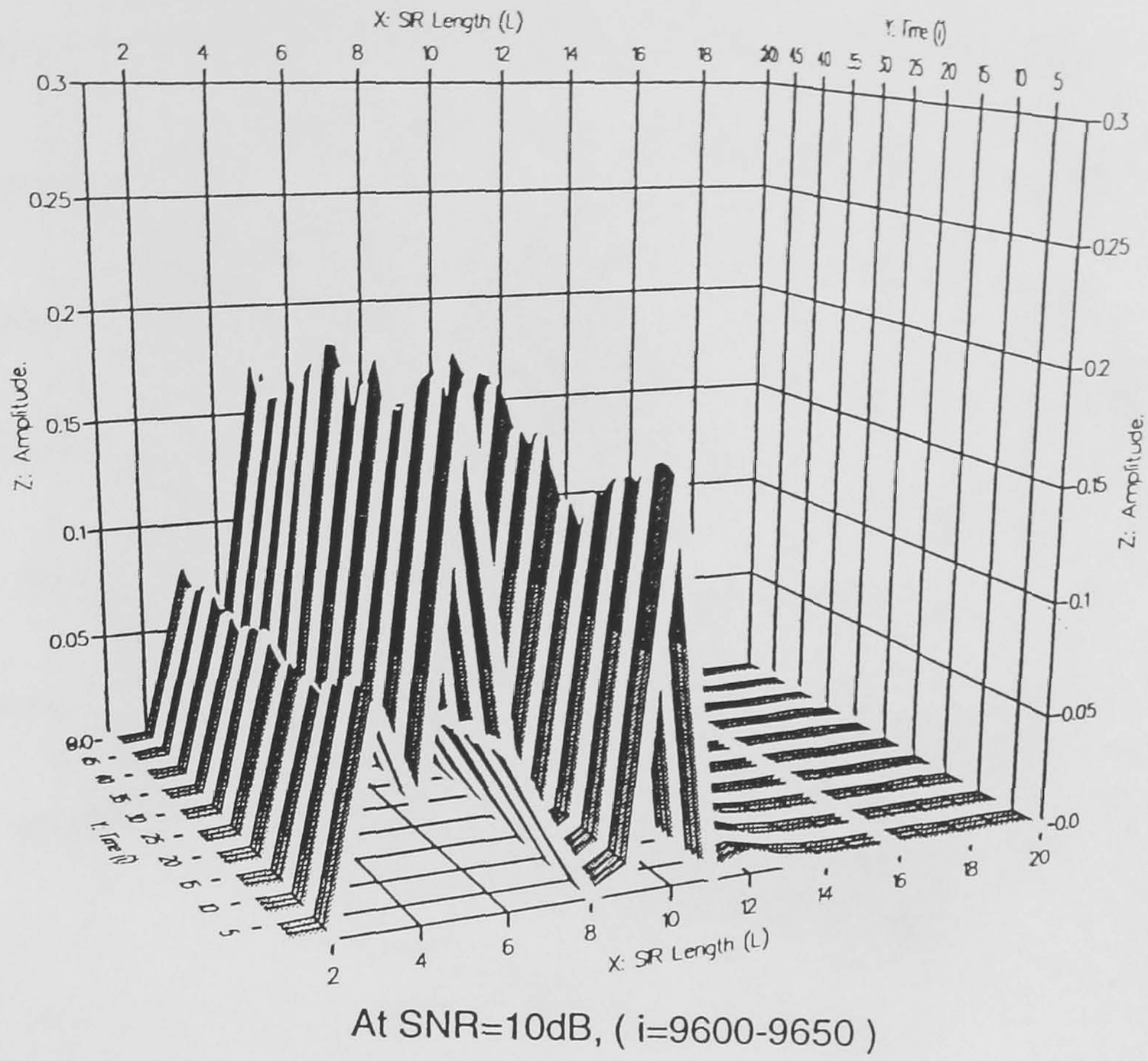


Fig. 6.7.7 A segment of Estimated SIR of Channel1 using Estimator3.

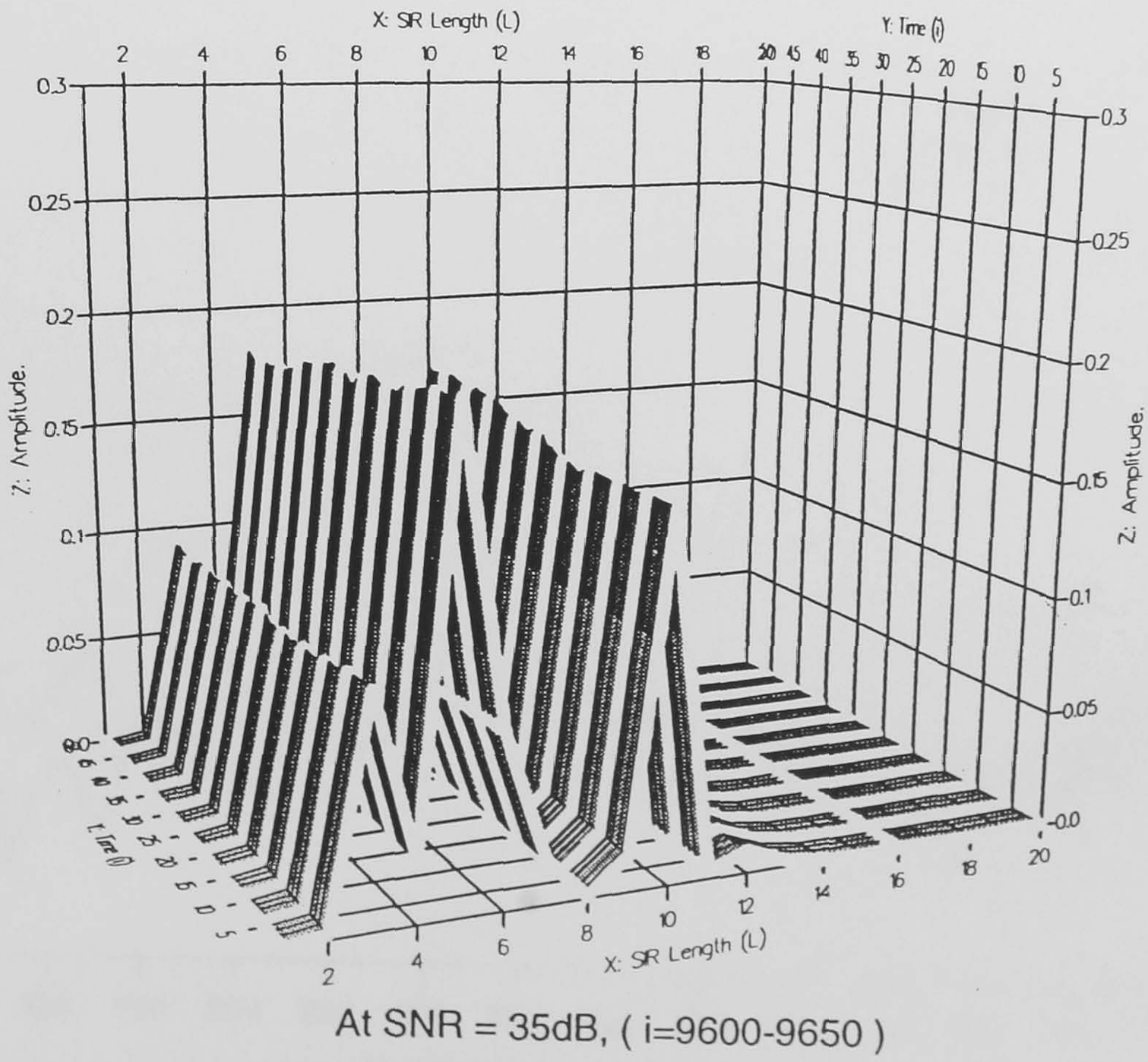


Fig. 6.7.8 A segment of Estimated SIR of Channel1 using Estimator3.

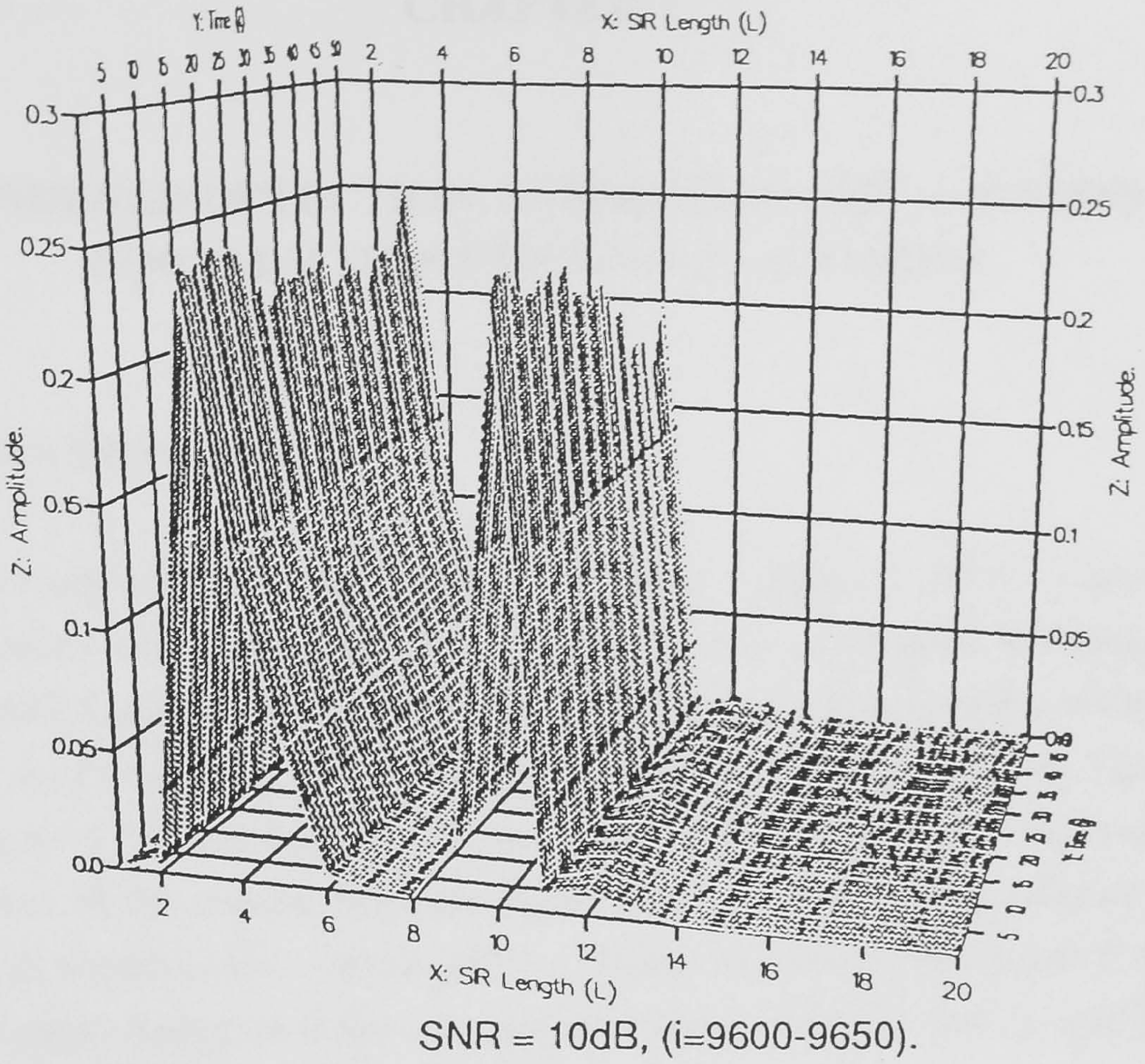


Fig. 6.7.9 A segment of Estimated SIR of Channel2 using Estimator3.

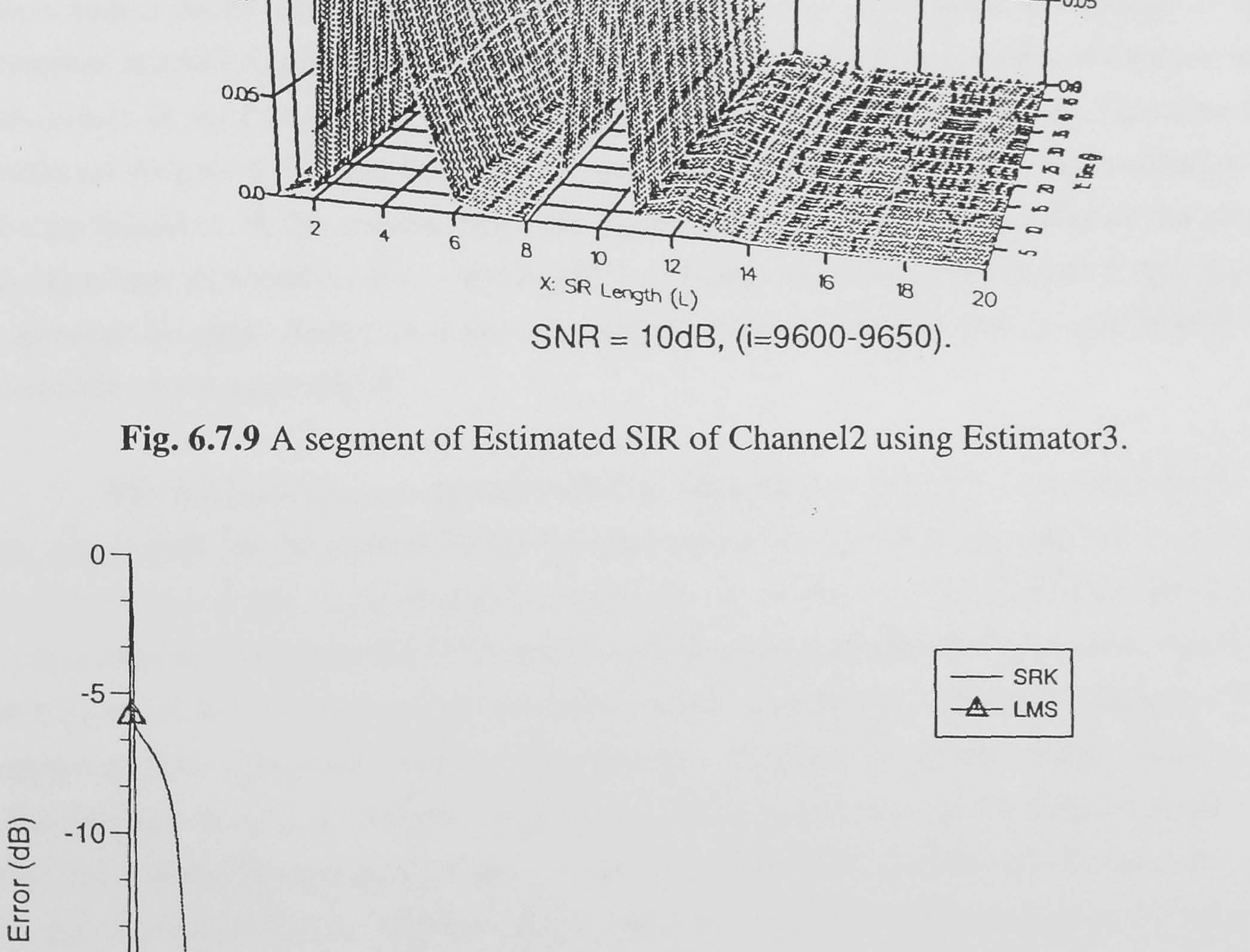


Fig. 6.7.10 Convergence Comparison of LMS and SRK Algorithm.

CHAPTER 7

COMBINED DETECTION, ESTIMATION AND ADAPTIVE EQUALISATION FOR 4800 bit/sec MODEM.

7.1 INTRODUCTION

In previous chapters, various near-maximum likelihood (NML) detectors have been tested under the assumption that the receiver has exact prior knowledge of the sampled impulse response (SIR) of the linear base-band channel (perfect estimation was assumed). In the case of channel estimators, perfect detection was assumed. Therefore the previous results of detectors and estimators may not give an accurate representation of the performance of the system, when the receiver has to operate without any of the above assumptions as would be the case in practice. Hence the results in Chapter 5 and 6 may represent an upper-bound in terms of the performance of the detectors and estimators, under the given assumption.

The minimum phase algorithm (MPA) described in Chapter 4 has been tested on the actual (SIR) of the channel rather than the estimated one. In practice the MPA operates on the estimated SIR of the channel provided by the channel estimator, as will be the case in this chapter. Therefore the MPA and the adjustment of the pre-detection filter described in Chapter 4 will be used in the combined system to make the channel response at least approximately minimum phase. The energy of each received signal element is concentrated around its earliest components. This is achieved by an iterative process to find most of the roots (zeros) of the Z-transform of the SIR, that lie outside the unit circle in the Z-plane. Next it replaces these roots by their complex-conjugate reciprocals, leaving the remaining roots unchanged. The adjustment method of the pre-detection filter uses the prior knowledge of these roots to update the tap-gains of the filter. The use of such a filter enables a near optimum performance to be achieved using simple detector. Therefore it is important to test the performance of the complete system when operating over a fading channel, without any of the above assumptions being made. The combined system considered here consists of three parts :

1. The detector

The results of Chapter 5 suggests that, the most cost effective NML detector in terms of performance and complexity is detector 3. Therefore the combined system will use detector 3 to detect the data symbols.

2. The estimator

Different types of estimator have been studied and tested in Chapter 6, and from the results estimator 3 with degree-1 prediction appears to be the candidate for the combined system. For comparison purposes, different types of estimators will be tested here.

3. Adaptive filter

This is the same as described in Chapter 4, where the filter is inserted between the sampler and NML detector, and it employs a minimum phase algorithm.

The above three parts are connected together to give the combined system.

In this chapter a brief description of the three basic parts of the system will be considered first. The chapter concludes with a discussion of the computer simulation tests and their results.

7.2 MODEL OF COMBINED SYSTEM

Fig.7.2.1 shows the model of the combined system that is used in the computer simulation tests, which is the same as that described in Chapter 3. For convenience the necessary description of the system will be considered here.

The linear base-band channel in Fig.7.2.1 consists of the transmitter filter, a linear modulator, a transmission path (HF channel), a coherent demodulator and a receiver filter. A sequence of regularly spaced impulses $(\sum_i s_i \cdot \delta\ (t - iT))$ representing the transmitted elements are fed to the input of a linear base-band channel. In the HF channel, the spectrum of the resulting voice-band QPSK signal is shifted to the HF band by a linear process of single side-band suppressed carrier amplitude modulation. This results in an HF radio signal, which is then transmitted via two or three independent sky waves with different fixed delays in transmission. The model of the HF channel with all parameters such as frequency spread and time delay that used in this tests were given in Chapter 2. At the output of the HF channel, stationary white Gaussian noise with zero mean and two-sided power spectral density of $0.5N_0$ is added to the distorted QPSK signal.

The faded, distorted and noisy received signal is passed through a linear coherent demodulator, which shifts its spectrum back to the voice band. It is assumed here that the reference carriers of the coherent demodulator are in phase quadrature and have the same frequency which is constant and equal to the average frequency of the received signal carrier, thus eliminating any constant frequency offset in the received QPSK signal. Since here we are not concerned to study the synchronisation of the system, ideal carrier phase tracking and symbol timing tracking is assumed. Synchronisation methods are given in [1,21,22,60,102,103].

The demodulated signal at the output of the coherent demodulator $r(t)$ is complex-valued, and is sampled once per data symbol, at the time instant $\{iT\}$, to give the corresponding received samples $\{r_i\}$ where :

$$r_i = \sum_{h=0}^{L} s_{i-h} \cdot p_{i,h} + v_i \qquad 7.2.1$$

where :

$$p_{i,h} = p_{i,h}(hT) \qquad 7.2.2$$

The $\{p_{i,h}\}$ are the components of the $(L+1)$-components row vector such that :

$$P_i = [p_{i,0}, p_{i,1}, \cdots\cdots\cdots, p_{i,L}] \qquad 7.2.3$$

P_i is the overall sampled impulse response (SIR) of the linear base-band channel. It is assumed that $p_{i,h} = 0$ for $0 > h > L$, where $(L+1)$ is taken to be 20 samples for both channels considered in these tests. The Z-transform of P_i is given by :

$$P_i(z) = p_{i,0} + p_{i,1} \cdot z^{-1} + \cdots\cdots\cdots + p_{i,L} \cdot z^{-L} \qquad 7.2.4$$

The noise components $\{v_i\}$ in eqn.7.2.1 are complex-valued Gaussian random variables with zero mean and variance σ_v^2.

The receiver in Fig.7.2.1 operates on the received samples $\{r_i\}$ to detect the values of the data symbols $\{s_i\}$, from the values of $\{\rho_i\}$ using a knowledge of the possible values of s_i. The output from the detector is the sequence $\{s_i'\}$, which is the sequence of detected data symbols. In order to start detection, the receiver must first form an accurate estimate (prediction) of the SIR of the channel and then it must adjust the coefficients of the feed-forward filter (C) and backward filter (G), in Fig.7.2.1. Therefore the received signal is fed to the adaptive filter C with $(N+1)$-taps, where the tap-coefficients are given by :

$$C_i = [c_{i,0}, c_{i,1}, \cdots\cdots\cdots, c_{i,N}] \qquad 7.2.5$$

with Z-transform :

$$C_i(z) = c_{i,0} + c_{i,1} \cdot z^{-1} + \cdots\cdots\cdots + c_{i,N} \cdot z^{-N} \qquad 7.2.6$$

The output signal from this filter at $(t=iT)$ is :

$$d_i = \sum_{h=0}^{N+L} s_{i-h} \cdot e_{i,h} + u_i \qquad 7.2.7$$

where u_i represents the sampled value of the Gaussian noise at the output of the linear filter C, and

$$E_i = [e_{i,0}, e_{i,1}, \cdots\cdots\cdots, e_{i,N+L}] \qquad 7.2.8$$

with Z-transform :

$$E_i(z) = e_{i,0} + e_{i,1} \cdot z^{-1} + \cdots\cdots\cdots + e_{i,N+L} \cdot z^{-N-L} \qquad 7.2.9$$

The sequence E_i is determined from the sequence C_i and the sequence P_i. In fact, the filter is such that $e_{i,h} = 0$; for $h=0,1,2,.....(N-1)$, (see Chapter 4 for more detail), and so introduces a delay of N sampling intervals.

The backward filter (filter G) in Fig.7.2.1 has L tap-gains and at time instant ($t=iT$) are given by the last (L)-components of the row vector Y_i :

$$Y_i = [y_{i,0}, y_{i,1}, \cdots\cdots\cdots, y_{i,L}] \qquad 7.2.10$$

Y_i can be considered as an estimate of the SIR of the channel and filter C, when the filter is correctly adjusted then :

$$y_{i,h} \approx e_{i,N+h} \quad ; \quad \text{for } h = 0,1,2,.....,L \qquad 7.2.11$$

and the sequence Y_i should be at least approximately minimum phased.

The sequence Y_i is determined by the minimum phase algorithm (MPA) in Fig.7.2.1, whose input is an estimate (prediction) of the SIR of the overall linear base-band channel ($P'_{i-N,i-N-1}$) at time instant $t=(i-N)T$. This prediction is given by :

$$P'_{i-N,i-N-1} = [p'_{i-N,i-N-1,0}, p'_{i-N,i-N-1,1}, \cdots\cdots\cdots, p'_{i-N,i-N-1,L}] \qquad 7.2.12$$

The output of the backward filter (filter G) is an accurate estimate of the intersymbol interference (ISI) present in d_i , where :

$$f_i = \sum_{h=1}^{L} s'_{i-h} \cdot y_{i,h} \qquad 7.2.13$$

The equalised signal is ρ_i, which is given by :

$$\rho_i = d_i - f_i \qquad 7.2.14$$

The detector in Fig.7.2.1 uses ρ_i to determine the detected value ($s'_{i-N-\mu}$) of the data symbol ($s_{i-N-\mu}$). Clearly, the NML detector itself introduces a delay of (μ) sampling intervals, so that the total delay in detection is ($N+\mu$) sampling intervals.

Finally, the detected symbol value ($s'_{i-N-\mu}$) is fed to a differential decoder, to give at its output the corresponding detected information ($\alpha'_{i-N-\mu}$).

The channel estimator in Fig.7.2.1 uses the 'early' detected data symbols $\{s'_{i-N-h}\}$, for ($h=0,1,2,.....L$ and $L < \mu$) together with the appropriate received samples, to form an estimate (or prediction) of the SIR of the channel at time instant $t=(i-N)T$, i.e. $P'_{i+1,i}$ which is needed by the minimum phase algorithm.

7.3 MINIMUM PHASE ALGORITHM AND ADJUSTMENT OF THE FILTER

The minimum phase algorithm and the adjustment method of the tap-gains of the linear filter used in the combined system is the same as described in Chapter 4. Therefore a brief description of the algorithm will be considered here, for simulation purposes.

At the time instant $t=iT$ and before the detection of $s'_{i-N-\mu}$, the MPA receives from the estimator a prediction of the SIR of the channel corresponding to time instant $t=(i-N)T$. The prediction of SIR is $P'_{i-N,i-N-1}$, which is given by eqn.7.2.12. The MPA algorithm operates on this prediction in an iterative manner to determine, in sequence, the zeros (roots of the Z-transform of $P'_{i-N,i-N-1}$), whose modulus greater than d, where d is set to 1.05. Then it replaces these roots by the complex conjugate of their reciprocals, all remaining roots (zeros) being left unchanged. The MPA first sets $Y=P'_{i-N,i-N-1}$ and then starts with the first starting point as follows :

In the first run at $i=1$, the algorithm starts with β_0 set to the first starting point number 1 in Fig.7.3.1, and starts to find the roots of the SIR, that are greater than 1.05. When the whole set of the nine starting points have been used and the algorithm has not converged to a root, the algorithm will be terminated and it will be assumed that all m roots, that are outside the required region in the unit circle in the Z-plane are located. The values of the negative of the reciprocals of the located roots $\{\gamma_m(i)\}$, will be stored in an array together with the initial nine starting points, to give a total of $m+9$ starting points. The new set of starting points is arranged for the next run as shown in Fig.7.3.2.

In the second run of the algorithm ($i=i+1$), the value of β_0 is set now to the first value in the new set of starting points in Fig.7.3.2 $\gamma_1(i)$, where $\gamma_1(i)$ here is the first root tracked in the previous run. This will speed the algorithm to converge much more quickly to its corresponding value $\gamma_1(i+1)$. If the roots corresponding to $\gamma_1(i+1)$ are greater than 1.05, the value of $\gamma_1(i+1)$ will replace $\gamma_1(i)$ in the starting point set, otherwise the value of $\gamma_1(i)$ will be omitted from the set. The algorithm next sets β_0 equal to $\gamma_2(i)$ and this of course, will converge to its corresponding value of $\gamma_2(i+1)$ and the process will be repeated m times. After m such repetitions, the algorithm will use the remaining nine starting points as described earlier. This of course first checks whether any more roots are left outside the required region in the unit circle in Z-plane, and at the same time attempts to locate the given roots. If this check results in a root with modulus greater than 1.05, the negative of its reciprocals will be stored together with the previous m $\{\gamma_m(i+1)\}$ in the

starting points array. Therefore, the starting points for any subsequent run are the roots tracked from the previous run, plus the initial nine starting points.

Fig.7.3.1 : Arrangement of the Initial Nine Starting Points for the Iteration Process.

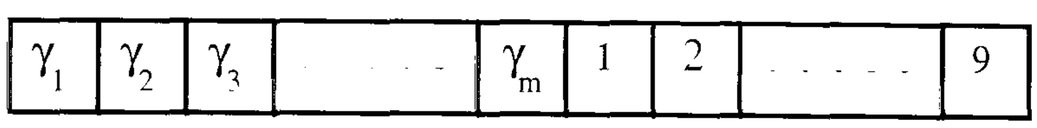


Fig.7.3.2 : Arrangement of New Starting Points for the Iteration Process.

Where the values of the initial starting points are given in Chapter 4. Therefore using the roots that have been found at $t=iT$ together with the initial starting points to locate the roots at time $t=(i+1)T$, will reduce the number of iteration process required by the algorithm to converge [39]. Bear in mind that the algorithm uses a one-tap feedback filter and two-tap feedforward filter as explained in Chapter 4 and Appendix C.
The procedure for allocating the roots is as follows :

At the start:

$L+1=20$, $\delta=10^{-10}$.

$d=(1/1.05)^2=0.90703$

$y_h=p'_{i-N,i-N-1,h}$, for $h=0,1,.....,L$

MAIN LOOP		number of real multiplication	number of addition	number of inverse
FOR $J=1,40$				
$e_{j,h} = y_h - \beta_j \cdot e_{j,h+1}$	(1)	4(L+1)	4(L+1)	0
for $h=L,L-1,.....,1,0$;$(e_{j,L+1} = 0)$.				
$\varepsilon_j = \sum_{h=1}^{L} e_{j,h} \cdot \beta_j^{h-1}$	(2)	4(L)	4(L)	0
$\varphi = e_{j,0}/\varepsilon_j$	(3)	6	3	2
$\beta_{j+1} = \beta_j + \varphi$	(4)	0	2	0
$\rho = \lvert \varphi \rvert^2$	(5)	2	1	0
IF $\rho < \delta$ THEN				
GOTO **CONV**				
ELSE				
$\theta = \lvert\beta_j\rvert^2$	(6)	2	1	0
Total number of operation:		8L+14	8L+11	2

IF $\theta > (d)$ THEN

GOTO **SSTP**

ENDIF

ENDIF

NEXT J

SSTP: "Find new starting point after diverge",

IF (all the starting points are not used) THEN

REPEAT THE MAIN LOOP

ELSE

GOTO **STORSTP**

ENDIF

CONV : " Using two-tap feedforward filter to alter the roots of the channel by the obtained roots from previous run" as:

$y_h = e_{k,h+1} + \beta_k^* \cdot e_{k,h}$ (7)	4(L+1)	4(L+1)	0

for $h=-1,0,1,2,.....,L$ $(e_{k,-1}=0)$

where $y_{-1}=e_{k,0}=0$

Total number of operation:	4(L+1)	4(L+1)	0

NEWSTP: " Find new starting points after convergence",

IF (all the starting points are not used) THEN

REPEAT THE MAIN LOOP

ELSE

GOTO **STORSTP**

ENDIF

STORSTP :

Add the new located roots to the initial nine starting points and store them in an array, for the next run.

TP-UPD : " Update the tap-gains of Filter C",

$c_h=0$; for $h=0,1,.....,N$

$c_{N-1}=1.0$

$e_h = c_h + \beta_k^* \cdot c_{h-1}$ (8)	4(N+1)	4(N+1)	0

for $h=N,N-1,.....,1,0.$ $(e_{N+1}=0)$

$c_h = e_h - \beta_k \cdot c_{h-1}$ (9)	4(N+1)	4(N+1)	0

for $h=-1,0,1,.....,N.$ $(e_{-1}=0)$

Total number of operation:	8(N+1)	8(N+1)	0

END.

The resultant Y and C are now taken to be the updated values of Y_i and C_i for the backward and forward filters C and G respectively, in the receiver Fig.7.2.1. It is obvious, that the above procedure may be carried out 'off line' in the sense that it is independent of all other activities in the receiver in Fig.7.2.1.

The number of arithmetic operations required by the algorithm is not fixed and it depends on the characteristics of the channel and the number of roots outside the required region in Z-plane. For one iteration, i.e. $j=1$, the number of elements in SIR ($L+1=20$), and the number of tap-gains of filter C is ($N+1=50$), the number of operation required by the process is :

	real multiplication's	additions or subtraction	inverse
one-iteration (j=1)	8L+14	8L+11	2
Locating of K roots	4(L+1)K	4(L+1)K	0
Update C	8(N+1)K	8(N+1)K	0

For example when $j=40$, and the number of roots that lie outside the required region is $k=10$, the process requires a total number of real multiplications about *11440*, a total number of additions or subtractions about *11320*, and *80* inverse operations.

Since the impulse response of a slowly time-varying channel varies smoothly with time, such that there are no discontinuities in the rate of change or acceleration with time of a sample of the impulse response, a relatively long period can be allowed between successive adjustments of the adaptive filter [23]. The above procedure can be carried out once every 2, 4, 8, or 16 samples. In the tests considered here the combined system will use the MPA to update Y and C every 1, 2, 4, 8, and 16 sampling intervals and the results of the computer simulation test will be presented later. Therefore a useful reduction in the average number of arithmetic operations involved in the process can be achieved. This is the key to the simple implementation of the algorithm in practice.

Finally, the number of operations required by the algorithm with 1, 2, 4, 8, and 16 sampling intervals are given in table 7.1, assuming $K=10$ roots outside the required region in the unit circle.

7.4 THE DETECTOR

The detector used in the combined system is detector 3, that described in Chapter 5. The number of stored vectors is $k=16$ and the delay introduced by detector 3 is $\mu=32$. This detector gives the best compromise between the performance and complexity as stated in Chapter 5 for the application studied here. The operation of this detector is exactly as described in Chapter 5, except that in the combined system considered here it

operates on the output sample (d_i) of the adaptive pre-detection filter (Filter C) in Fig.7.2.1, rather than on the received sample (r_i), where (d_i) is given by :

$$d_i = \sum_{h=0}^{N} r_{i-h} \cdot c_{i,h} \qquad 7.4.1$$

Also the detector, here uses the vector (Y_i), which is an estimate of the cascade of the sampled impulse response (SIR) of the channel at $t=(i-N)T$, and the linear feedforward filter (Filter C), which is provided by the minimum phase algorithm. The detector uses (Y_i) to adjust the tap-gains of the feed back filter (Filter G). The signal (f_i) at the output of adaptive filter G is an accurate estimate of inter-symbol interference present in (d_i), which is given by :

$$f_{i,j} = \sum_{h=1}^{L} s'_{i-h,j} \cdot y_{i,h} \qquad 7.4.2$$

The detector calculates eqn.7.4.2 for $j=1,2,.....k$.

The signal at the input of detector (ρ_i) is now given by :

$$\rho_i = d_i - f_i \qquad 7.4.3$$

The detector operates on ρ_i to detect the data symbol s_i in the following manner :

Just prior to receipt of the sample (d_i) at time $t=iT$, the detector holds in store ($k=16$) different μ-components vectors $\{Q_{i-N-1}\}$ with their costs $\{B_{i-N-1}\}$, where :

$$Q_{i-N-1} = [x_{i-N-\mu}, x_{i-N-\mu+1}, \cdots\cdots\cdots, x_{i-N-1}] \qquad 7.4.4$$

and

$$B_{i-N-1} = \sum_{j=N}^{i-1} |d_j - \sum_{h=0}^{L} x_{j-N-h} \cdot y_{j,h}|^2 \qquad 7.4.5$$

where $|x|$ is the absolute value of the scalar quantity x and $x_i = 0$; for $i <= 0$.

The $k=16$ vectors are divided into three groups, where in group 1, group 2 and group 3 there are k_1, k_2 and k_3 $\{Q_{i-N-1}\}$ vectors respectively. The value of k_1, k_2 and k_3 are *2*, *4* and *10* respectively.

On receipt of sample (d_i), the detector expands the $k=16$ vectors into the corresponding k_{tot} $\{Q'_{i-N}\}$ vectors in the following way :

Each of the $k_1=2$ vectors $\{Q_{i-N-1}\}$ in group 1 is expanded into *4* vectors to give $4k_1=8$ $(\mu+1)$-components vectors $\{Q'_{i-N}\}$, where :

$$Q'_{i-N} = [x_{i-N-\mu}, x_{i-N-\mu+1}, \cdots\cdots\cdots, x_{i-N-1}, x_{i-N}] \qquad 7.4.6$$

stored with their associated cost $\{B_{i-N}\}$, which is now given by :

$$B_{i-N} = B_{i-N-1} + |d_i - \sum_{h=0}^{L} x_{i-N-h} \cdot y_{i,h}|^2 = B_{i-N-1} + |q_i|^2 \qquad 7.4.7$$

The first μ components of the vector Q'_{i-N} are the same as in vector Q'_{i-N-1}, and the last component (x_{i-N}) takes one of the four possible values of (s_{i-N}). At this stage we have 8 vectors with their associated costs.

Each of the $k_2=4$ vectors $\{Q_{i-N-1}\}$ in group 2 is expanded into 2 vectors to give $2k_2=8$ (μ+1)-component vectors $\{Q'_{i-N}\}$with smallest cost, as described in Chapter 5. Again the first μ components of the vector Q'_{i-N} are the same as in vector Q'_{i-N-1}, and the last component (x_{i-N}) is chosen to give the corresponding two smallest cost vectors. This is done such that, for each vector Q_{i-N} the detector calculates :

$$q_{i-N} = d_{i-N} - \sum_{h=1}^{L} x_{i-N-h} \cdot y_{i,h} - x_{i-N} \cdot y_{i,0} = \rho'_{i-N} - x_{i-N} \qquad 7.4.8$$

where $y_{i,0} = 1.0$, which is set by the MPA. The value of x_{i-N} for which $|q_{i-N}|^2$ and therefore $|B_{i-N}|^2$, are smallest is determined first by selecting the possible values of the real and imaginary parts of x_{i-N} that are closest, respectively to the real and imaginary parts of ρ'_{i-N}. This selection process is done by applying a simple threshold level comparison, without evaluating the $|q_{i-N}|^2$ itself. The value of x_{i-N} just determined is now used to form the quantity q_{i-N} eqn.7.4.8. The second value of x_{i-N}, which gives the second smallest vector Q'_{i-N} in group 2, is determined as explained in Chapter 5.

Each of the $k_3=10$ vectors $\{Q_{i-N-1}\}$ in group 3 is expanded into one vector to give $k_3=10$, (μ+1)-components vectors $\{Q'_{i-N}\}$, where x_{i-N} is now determined by threshold level comparison to the corresponding smallest cost vector as described previously.
At this stage the detector holds $k_{tot}=4k_1+2k_2+k_3=26$ expanded vectors $\{Q'_{i-N}\}$,together with their associated costs $\{B_{i-N}\}$.

The detector now selects the smallest cost vector from the set of 26 vectors $\{Q'_{i-N}\}$, and takes its first component $x_{i-N-\mu}$ to be the detected data symbol $s'_{i-N-\mu}$, and the rest of the operation is carried out exactly as described in Chapter 5.
In addition to the final detected data symbol $s'_{i-N-\mu}$, the detector provides the early detected data symbols $\{s'_{i-N-h}\}$; for $h=0,1,2,\ldots\ldots,L$, which is needed by channel estimator.

Finally, the number of operations required by detector 3 for each step is given below :

START:

1. Calculation of ISI in the signal (d_i), using eqn7.4.2 :

FOR j=1,k	Real multiplication	real addition	Comparison
$f_{i,j} = \sum_{h=1}^{L} x_{i,\mu-h,j} \cdot y_{i,h}$	4kL	4kL	0
NEXT J			

2. Pass received sample r_i via the linear filter C to form d_i :

$d_{i-N} = \sum_{h=0}^{N} r_{i-N-h} \cdot c_{i,h}$	4(N+1)	4(N+1)	0

3. Calculation of the costs $B_{i\text{-}N}$ using eqn.7.4.7, and applying threshold level comparison, and calculate $q_{i\text{-}N}$ using eqn.7.4.8.

2k	6k	2k

4. Forming the expanded vectors $\{Q'_{i\text{-}N}\}$ and their costs in group 1, (k_1=*2*):

$8k_1$	$4k_1$	0

5. Forming the expanded vectors $\{Q'_{i\text{-}N}\}$ and their costs in group 2, (k_2=*4*):

$2k_2$	$4k_2$	$3k_2$

6. Selecting the smallest cost vector and determine the detected data symbol :

0	0	$k_{tot}=4k_1+2k_2+k_3$

7. Preparation for next detection, which includes :

Omitting all vectors whose first component not equal to the first component ($x_{i\text{-}N\text{-}\mu}$) of the smallest cost vector. 0 0 k_{tot}

Selecting of (*k-1=15*) lowest cost vectors from the remaining vectors :

0	0	$(k-1)k_{tot}$

Storing the final *k* vectors with their associated cost for next run and subtract costs to bring the smallest cost to zero : 0 k 0

END.

The total number of operations required by detector 3 are :

Real multiplication's : $4k(L)+4(N+1)+2k+8k_1+2k_2 = 2k(2L+1)+2(4k_1+k_2)+4(N+1)$.

Real addition : $4k(L)+4(N+1)+6k+4k_1+4k_2+k=k(4L+7)+4(k_1+k_2)+4(N+1)$.

Comparison : $2k+3k_2+2k_{tot}+(k-1)k_{tot}=2k+3k_2+k_{tot}+k.k_{tot}$.

The total number of arithmetic operations required by detector 3 per one run, for $k=16$, $k_1=2$, $k_2=4$, $k_3=10$, then $k_{tot}=26$, $L+1=20$ and $N+1=50$ or *30*, are given in table 7.1.

7.5 THE CHANNEL ESTIMATOR

The channel estimator used in combined system Fig.7.2.1 is the adaptive step size least mean square estimator with degree-1 prediction (ALMS-D-1), that described in

Chapter 6 and referred to estimator 3. The reason for choosing estimator 3 is that, estimator 3 appears to be the most cost effective estimator in terms of performance and complexity as stated in Chapter 6. In this simulation test five estimators, that described in Chapter 6, will be examined for comparison purposes and in order to find the combined system most suitable for data transmission over HF channel. Therefore the results of all arrangements of combined system will be presented at the end of this chapter, and for convenience, the operation of estimator 3 will be given here.

The function of the channel estimator is to provide, for each received sample r_i a 1-step prediction of the complex-valued channel response P_{i+1} denoted by $P'_{i+1,i}$, where :

$$P'_{i+1,i} = [p'_{i+1,i,0}, p'_{i+1,i,1}, \cdots\cdots\cdots, p'_{i+1,i,L}] \qquad 7.5.1$$

In addition to the received sample r_i the estimator requires the "early detected" data symbols $[s'_i, s'_{i-1}, \ldots\ldots\ldots, s'_{i-L}]$, which are provided by the detector. Therefore at time $t=iT$, the estimator requires two input signals :

1. The received signal corresponding to the time instant $t=(i-N)T$, where

$$r_{i-N} = \sum_{h=0}^{L} s_{i-N-h} \cdot p_{i-N,h} + v_{i-N} \qquad 7.5.2$$

2. The "early detected" data symbols S'_{i-N}, where :

$$S'_{i-N} = [s'_{i-N}, s'_{i-N-1}, \cdots\cdots\cdots, s'_{i-N-L}] \qquad 5.7.3$$

According to the model of the combined system in Fig.7.2.1, the estimator determines $P'_{i+1,i}$ by the following steps :

	number of multiplication	number of addition	number of comparison	number of SQRT	
STEP1 : Calculating r'_{i-N} :					
$r'_{i-N} = \sum_{h=0}^{L} s'_{i-N-h} \cdot p'_{i-N,i-N-1,h}$;	4(L+1)	4(L+1)	(0)	(0)	7.5.4
STEP2 : Calculation e_{i-N} using :					
$e_{i-N} = r_{i-N} - r'_{i-N}$	(0)	(2)	(0)	(0)	7.5.5
STEP3 : Calculating Φ_{i-N} using :					
$z_{i-N,h} = \sqrt{(\Re\, p'_{i-N,i-N-1,h})^2 + (\Im\, p'_{i-N,i-N-1,h})^2}$ for $h=0,1,\ldots\ldots L$	2(L+1)	(L+1)	(0)	(L+1)	7.5.6

$z_{i-N,h} > d_0 \quad \phi_{i-N,h} = (z_{i-N,h})^{0.5}$
$z_{i-N,h} < d_0 \quad \phi_{i-N,h} = d_1$ (0) (0) (L+1) (L+1) 7.5.7

STEP4 : Forming the correction vector $\Delta_{i\text{-}N}$
whose h[th] elements $\delta_{i\text{-}N,h}$ is given by :

$$\delta_{i-N,h} = \phi_{i-N,h} \cdot e_{i-N} \cdot s'^{*}_{i-N-h} \quad 6(L+1) \quad 2(L+1) \quad (0) \quad (0) \quad 7.5.8$$

for $h=0,1,2,.....,L$

STEP5 : Forming the one-step prediction using :

$$p''_{i-N+1,i-N,h} = p''_{i-N,i-N-1,h} + (1-\lambda)^2 \cdot \delta_{i-N,h}$$

for $h=0,1,2,.....,L$ 2(L+1) 2(L+1) (0) (0) 7.5.9

$$p'_{i-N+1,i-N,h} = p'_{i-N,i-N-1,h} + p''_{i-N+1,i-N,h} + (1-\lambda^2) \cdot \delta_{i-N,h}$$

for $h=0,1,....,L$ 2(L+1) 4(L+1) (0) (0) 7.5.10

Total number of operations : 16(L+1) 13(L+1)+2 (L+1) 2(L+1)

END.

Where :

$r'_{i\text{-}N}$ is the estimated complex-valued received sample.

$P'_{i-N,i-N-1} = [p'_{i-N,i-N-1,0}, p'_{i-N,i-N-1,1}, \cdots\cdots\cdots, p'_{i-N,i-N-1,L}]$ is the prediction of the SIR of the overall linear base-band channel corresponding to the time instant $t=(i\text{-}N)T$, which was determined by the estimator at time $t=(i\text{-}1)T$.

$e_{i\text{-}N}$ is the complex-valued error sample in the estimate of the received sample.

$\Delta_{i-N} = [\delta_{i-N,0}, \delta_{i-N,1}, \cdots\cdots\cdots, \delta_{i-N,L}]$ is the correction vector.

$\Phi_{i\text{-}N}$ is the adaptive step size vector real-valued, and is determined as described in Chapter 6.

$z_{i-N,h} = |p'_{i-N,i-N-1,h}|$; for $h = 0,1,.....,L$ is the absolute value of the updated estimate of the SIR of the channel.

RP', IP' are the real and imaginary part of P'_i.

$\phi_{i\text{-}N,h}$ for $h=0,1,2,.....L$ are the elements of the adaptive step-size vector $\Phi_{i\text{-}N}$, given by : $\phi_{i-N,h} = F(z_{i-N,h})$; for $h=0,1,2,.....,L$.

$F(z_{i\text{-}N,h})$ is a monotonically non-decreasing positive real valued function of $(z_{i\text{-}N,h})$, such that :

$$\phi_{i-N,h} = \sqrt{z_{i-N,h}} \quad \text{for } h=0,1,2,.....,L \qquad 7.5.11$$

In order to avoid possible instability of the algorithm given by eqn.7.5.8. the value of $\Phi_{i-N,h}$ should be limited as in eqn.7.5.7.

d_0 and d_1 are an appropriate positive real valued constants.

(s'^{*}_{i-N}) is the complex conjugate of (s'_{i-N}).

$P''_{i-N+1,i-N}$ is the prediction of the rate of change of P_{i-N+1}, and λ a is real-valued constant in the range $(0---1)$ and usually close to 1.

$P'_{i-N+1,i-N} = [p'_{i-N+1,i-N,0}, p'_{i-N+1,i-N,1}, \cdots\cdots\cdots, p'_{i-N+1,i-N,L}]$ is the one-step prediction of the SIR of the channel corresponding to the time instant $t=(i-N+1)T$, which is needed by the MPA, at time instant $t=(i+1)T$, i.e. at the next time instant.

At the start of estimation process the estimator sets :

$P''_{1,0} = [0]$ and $P'_{1,0}=P'_1$, where P'_1 is determined by an appropriate synchronising signal that precedes the transmission data [23,101].

The adjustable parameters d_0, d_1 are kept fixed as given in Chapter 6, where the value of d_0, d_1 are set to 0.003 and 10^{-6} respectively over both channels considered in these tests. The adjustable parameter λ is optimised at the given signal-to-noise ratio to minimise the average mean square error. Since λ is used in Eqns.7.5.11-7.5.12, it's the most important parameter used by the estimator as it controls the rate of adaptation. The value of λ will be discussed later.

Finally, the total number of arithmetic operations required by the estimator are $16(L+1)$ multiplications, $13(L+1)+2$ additions, $L+1$ comparisons and $2(L+1)$ square roots. $L+1$ is taken to be 20 for both channels. The total number of arithmetic operations thus required by the estimator is given in table 7.1.

7.6 RETRAINING PROCESS :

In order to avoid complete failure (collapse) of the estimator/detector process, due to the severe interference or deep fade, (p_i becomes very small in certain parts of the sequence $\{P_i\}$) during prolonged transmission of data symbols, the receiver requires the transmission of a training sequence at periodic intervals. A retraining process at the receiver then resets the receiver (modem) to its correct position regularly.

The failure of the system will give a high error rate in $\{s_i\}$, which reduces the accuracy of the channel estimator, since the latter depends on the "early" detected data symbols $\{s'_i\}$and the previous estimate of the channel, as can be seen from Fig.7.2.1 and eqns.7.5.11-7.5.12. This inaccuracy in the estimates of the channel gives as a result an inaccurate estimate of the channel and filter $\{Y_i\}$, used by the detector. Consequently, there will be more errors in the $\{s'_i\}$, even when the signal is not in a deep fade, ($|P_i|$ becomes high again), and the system will collapse. As the result, the system will lose the accurate information of the channel, which is important for the correct operation of the

detection process. Such a failure in the combined system can be recovered by providing the system with an accurate estimate of the channel. This may be achieved by sending a retraining data sequence periodically [23,26,101]. It was decided to use a simple and ideal retraining process as follows :

The data is now transmitted by frames, where each frame contains *960* symbols. After the receipt of every frame, the receiver in the combined system, whether there has been a collapse or not, carries out the following operations :

1. The detector is reset to its start up position where all values in each $k=16$ stored vector $\{Q_{i-1}\}$ are made correct by having $x_{i-N-h}= s_{i-N-h}$, for $h=1,2,3,.....\mu$, ($\mu=32$), and all associated costs $\{B_{i-N-1}\}$ are given large positive value, except the first vector whose cost is assigned to zero. The detector now operates normally on the following received samples $\{r_i\}$, until the receipt of the next frame.

2. Correct estimation is assumed after each frame of transmission, where the prediction vector $P'_{i-N,i-N-1}$ eqn.7.5.5 is set to the actual SIR of the channel, i.e. $P'_{i-N,i-N-1}=P_{i-N}$ and the estimator is reset to its start up position, where $P''_{1,0}=[0]$.

Therefore at the end of every frame, (960^{th} sample), the receiver resets the detection and estimation process to their start up positions. The MPA in the combined system does not need any special operation (initialisation) during the retraining process.

7.7 RESULTS OF THE COMPUTER SIMULATION TESTS

Computer simulation tests have been carried out on the combined system shown in Fig.7.2.1, when operating at *4800 bit/sec*. The system is a combined detector/estimator, employing the minimum phase algorithm (MPA) to adjust the tap-gains of the adaptive feedforward and backward filter. The system was tested over channel 1 and 2, whose parameters are given in Chapter 2.

The detector used in the combined system is detector 3 with a *16* stored vectors and *2*, *4* and *10* vectors in group1, group2 and group3 respectively. The delay introduced by the detector is ($\mu=32$) sampling intervals. In all tests the adaptive backward filter (G) has L taps, where the feed-forward filter C has $(N+1)=50$ or *30*, so the total delay in detection is $N+\mu=81$ or *61 sampling intervals*. The number of components in the SIR of channel 1 and 2 is *20*, i.e. $L+1=20$, $L=19$. The number of iteration processes J for the MPA, used by the receiver, is *40* or *30* and the value of d is set to *1.05* in all tests, except for two cases, where d is set to *1.002* and *1.1*. Around *60,000* received samples $\{r_i\}$ have been involved in each test. The signal-to-noise ratio (SNR) Ψ is as defined by eqn.5.6.1 where :

$$\Psi = 10\log_{10}(\frac{E_B}{0.5N_0}) \qquad 7.7.1$$

The average mean-square error in the estimation of the SIR of the channel is measured in dB and is exactly as given in eqn.6.7.1 as

$$\zeta_1 = 10\log_{10}[\frac{1}{57600}\cdot \sum_{i=1041}^{58640} |P_i - P'_{i,i-1}|^2] \qquad 7.7.2$$

The estimator used in the combined system is estimator 3 with degree-1 prediction, which gives the best performance overall as shown in Chapter 6, under perfect conditions, i.e. correct detection was assumed. To confirm this fact and for comparison purposes with other estimators, five estimators have been tested in the combined system without assuming the correct detection. Therefore the following seven arrangements of combined system have been tested by computer simulation. These are :

1. Combined detector, adaptive equaliser and gradient estimator without prediction referred to as (*FLMS:D-0*).
2. Combined detector, adaptive equaliser and gradient estimator with degree-1 prediction referred to as (*FLMS:D-1*).
3. Combined detector, adaptive equaliser and adaptive step-size gradient estimator without prediction referred to as (*ALMS:D-0*).
4. Combined detector, adaptive equaliser and adaptive step-size gradient estimator with degree-1 prediction referred to as (*ALMS:D-1*).
5. Combined detector, adaptive equaliser and square-root Kalman estimator without prediction referred to as (*SRK:D-0*).
6. Combined detector, adaptive equaliser and square-root Kalman estimator with degree-1 prediction referred to as (*SRK:D-1*).
7. Combined non-linear (decision feedback) equaliser referred to as (*CNLEQ*).

The results of the simulation tests are as follows :

STAGE 1

The first test is carried out on the combined detector 3, estimator 3 with degree-1 prediction and the adaptive equaliser with and without the retraining process. In the case of no training, the only condition is that at $i=1$, the estimated SIR of the channel is set to the actual SIR, i.e. $P'_1=P_1$. The results is shown in Figs.7.7.1-7.7.2. Figs.7.7.1-7.7.2 show the error in the estimated SIR where :

$$ERR_i = 10\log_{10}[|P'_i - P_i|^2] \qquad 7.7.3$$

It can be seen from Fig.7.7.1, when no retraining process is used, the system collapses after the deep fade over both channels and gives a high error rate. During the test it was

observed that the system collapses at SNR less than *15dB* and *20dB* over channel 1 and channel 2 respectively. This is due to the errors in the $\{s'_{i\text{-}N\text{-}\mu}\}$ over a fairly long period of time, reducing the accuracy of the estimates of the channel, since the estimates of the channel depends on $\{s'_{i\text{-}N\text{-}\mu}\}$ and the previous estimates of the channel. When the system collapses due to severe ISI or deep fade, it will never recover again as it can be seen from Fig.7.7.1. In fact after a certain period of time, the estimates of SIR provided by the channel estimator become quite different from the actual SIR of the channel. Thus the system loses the accurate information of the channel, which is important for the correct operation of the detection process. Therefore due to the collapse of the system, it was decided to use the retraining process, described in section 7.6 in all tests. The error of the estimate of SIR (when retraining process included) is shown in Fig.7.7.2 over both channels at SNR equal to *15* and *20dB*. Comparing the graphs without and with retraining process, it can be seen that the combined system recovers from a collapse after the retraining. It is clear from Fig.7.7.2, when the error is very large due to the deep fade, that the system recovers again from this high error after the retraining process. Therefore the combined system without the retraining process will collapse and never recover again as illustrated in Fig.7.7.1, but when using the retraining process and after the deep fade, the system may collapse, but then it recovers after the next frame as shown in Fig.7.7.2.

STAGE 2

The results of the 6 arrangements of combined detector 3, adaptive equaliser and channel estimator with retraining process are shown in Figs.7.7.3-7.7.4 over channels 1 and 2. Fig.7.7.3 shows the bit error rate measurement of the combined systems with retraining process over channel 1. The plot error rate of the 'upper bound' is also shown in Fig.7.7.3 and is denoted (*PE*), where perfect estimation is assumed. The adjustable parameters of all estimators, such as (b, w, μ, λ, λ_1 and λ_2) are set to their optimum value at the given signal-to-noise ratio. It can be seen from the figure that the best performance is given by the combined systems 4 and 6. The combined system 4 (ALMS:D-1) gives the best result over the whole range of signal-to-noise ratios *5-25dB* and requires fewer operations than combined system 6 (SRK:D-1). Also it can be seen that the degradation caused by combined system 4 is, at error rates of 10^{-3} and 10^{-4}, of *4-5dB* and *1-2dB* respectively relative to the 'upper bound'.

Fig.7.7.4 shows the performance of the combined systems with retraining process when operating over channel 2. Again the best performance is given by system 4 (ALMS:D-1). Comparing the bit error rate of the system 4 with the 'upper bound' (*PE*), it can be seen that, there is a degradation, at error rates of 10^{-3} and 10^{-4}, of *4-5dB* and *1.5-2.5dB* respectively, relative to the 'upper bound'. The combined system 6 (SRK:D-1) gives close performance to system 4 at high SNR, where at low SNR system 6 is inferior.

It can also be seen from the previous figure that all combined systems perform well over channel 1 and worst over channel 2 with the degradation, at error rate of 10^{-4}, of some 5-6dB relative to the performance over channel 1. The inferior performance of the system over channel 2 is essentially due to the increase in signal distortion. Channel 2 exhibits a more rapidly fading rate, since it has two-paths with a large frequency spread and time delay, whereas channel 1 with three paths with the same frequency spread and time delay would exhibit less fading rate than channel 2.

It is clear from previous observation that the estimator performs well under perfect conditions (correct detection is assumed), but still gives good performance without the latter assumption. The result obtained here agrees with the result obtained in Chapter 6.
Therefore the adaptive step-size gradient estimator with degree-1 prediction (estimator 3) is selected as the channel estimator in the combined system. Towards the end of this chapter, all the tests will be carried out using the previous system. The minimum phase algorithm (MPA) and the adjustment of the feed-forward and backward equaliser are carried out every sampling interval in this stage of test. The number of iteration processes used by MPA to allocate the roots of SIR that lie outside the required region in the unit circle in Z-plane (greater than d) is set to 40 and the value of d is set to 1.05. The number of the tap-gains of the feed-forward filter is set to 50, i.e. $N+1=50$.
Another test has been carried out using the combined system 4, where now the number of iterations is set to 30 and the value of d is set to 1.1, 1.05 and 1.002. The results of computer simulation are shown in Fig.7.7.5. It can be seen that the performance of the system is improved as the value of d is reduced. Bear in mind that the smaller the value of d, the more processes required by the algorithm per received sample r_i to adjust the tap-gains of both equalisers. Thus increasing d reduces the complexity of the algorithm but increases the degradation of the system. Therefore the system with $d = 1.05$ is selected to be the best compromise in terms of complexity and performance as indicated in Fig.7.7.5

For comparison purposes the combined non-linear (decision feedback) equaliser (CNLEQ) has been tested over both channels and the result is shown in Fig.7.7.6. From the figure an advantage of 4-6dB can be gained by using the NML detector over the decision feedback equaliser system. Bear in mind that the tap-gains of the feed-forward and backward equaliser are adjusted as before using estimator 3 and MPA. Thus the non-linear equaliser system operates under the same condition, as the NML detector.
An interesting observation at this stage of tests is that, using combined detector 3, estimator 3 with degree-1 prediction and adaptive equaliser, the system operates well over both channels and it gives close result to the 'upper bound' (PE) case at high SNR (greater than $20dB$). Therefore further tests will be carried out on this system, in order to reduce the computational complexity without unduly affecting the performance. The most critical parts of the receiver in terms of complexity are the minimum phase algorithm and the

channel estimator. Since they required a high number of operations per received sample r_i, a useful reduction in complexity can be achieved by using the MPA once every *2, 4, 8* or *16* sampling intervals (*T*). In the next stage of tests the combined system is tested over both channels, where now the MPA is employed once every *2, 4, 8* or *16 T*.

STAGE 3

Computer simulation tests are carried out on combined system 4 with the number of iteration processes, used by MPA, set to *40* and the value of *d* set to *1.05*. The MPA is now carried out once every *2, 4, 8* and *16 T*, and the results are presented in Figs.7.7.7-7.7.8. It can be seen from the graphs that the degradation in performance is about *0.5-1dB*, when the MPA is carried once every *2, 4*, and *8T* where it is about *1-2dB* when MPA is carried out once every *16T* (*T: sampling interval*) relative to the system when MPA is carried out every *T*, over both channels. As the result of this test the combined system 4, that employ MPA once every *8T*, appears to be the most cost effective system in terms of complexity and performance overall the systems tested in this stage.

Another test has been carried out on the combined system 4, where the MPA is carried out once every *8T* with the number of tap-gains of the feed-forward equaliser reduced to *30*, i.e. *N+1=30*. The result is given in Fig.7.7.9. It can be seen from the figure that the system still performs well over both channels and gives very close result relative to that when *N+1=50*.

STAGE 4

From the results in previous stage it is observed that the combined system 4, employing MPA once every *8T* with *N+1=30* operates well over both channels with less computational requirement per received sample r_i. Therefore further tests were carried out on the system in order to reduce the arithmetic operations required by channel estimator. Since the channel studied here is a slowly time-varying channel, i.e. the SIR of the channel does not change significantly over *4* or *8* sampling intervals, the calculation of the step-size can be carried out once every *4* or *8* sampling intervals. This reduces the number of operations required by the channel estimator, since the calculation of the step-size involves the square root (SQRT) operation in the order of *2(L+1)* per received sample as shown in section 7.5. Therefore a useful reduction can be achieved using this method, and the total number of arithmetic operations required by the estimator is now given in Table 7.1-d

The result of the computer simulation test is presented in Fig.7.7.10, and this system is denoted as *CSC1M8N30ES8*. It is clear from the figure there is no degradation in the performance caused by calculating the step size once every *8T* relative to the system *CSC1M8N30*, where the calculation of step size is carried out every *T*. In the

graph is shown also the combined system 4 that employs MPA once every sampling interval and *N+1=50* and referred to as *CSC1M1N50*. It can be seen from the previous figure that *CSCM8N30ES8* gives some improvement in the performance over the system *CSCM8N30*, close to the performance of *CSCM1N50* and it requires fewer operations than both.

The adjustable parameters of estimator 3 such as d_0, d_1 and λ are set to their optimum value. Since the influence of d_0, d_1 is small in comparison with that of λ, therefore d_0, d_1 are kept fixed during all tests and their values are *0.003* and 10^{-6} respectively as given in Chapter 6.

The only parameter that needs to be optimised is λ. Unfortunately there is no optimum value of λ for the signal-to-noise ratio (SNR) varying from *5-30dB*. However in all tests carried out previously, the value of λ is set to its optimum value at the given SNR and the performance of the system with the optimum value of λ is given in Table 7.2 (a,b) for both channels. It is clear from the table the value of λ can be fixed for high SNR *23-28dB*. It was observed during the tests that the most interesting value of λ for SNR varying from *20-28dB* is in the range *0.982-0.967*. As the value of λ decreases from *0.982* to 0.967 the performance of the system improves for SNR *23-28dB*, but for SNR *20-23dB*, it is worse especially over channel 2. The influence of λ is more noticeable on channel 2 than on channel 1, since channel 2 exhibits a more rapidly fading rate. As it is impossible to fined one optimum value of λ for optimising the system over SNR varying from 20 to 28 and at the same time over channel 1 and 2, we have to accept some degradation if the value of λ is preferred to be fixed. For this particular application two interesting values of λ seem to be *0.977* and *0.973* can be used. The performance of the system for λ=*0.973* is given in Table 7.2 (c-d) over both channels.

STAGE 5

Since the retraining process reduces the actual data transmission rate due to the training sequence at the start of each frame, it is interesting to test the system without a retraining process using a high signal-to-noise ratio. Thus computer simulation tests have carried out on combined system 4 that employs MPA once every *8T* and the adaptive estimator calculates the step-size once every *8T*. The number of tap-gains of the feed-forward equaliser is set to *30,* i.e. *N+1=30* and the only condition assumed here is that, at the start (*i=1*) of the detection the receiver is assumed to have the correct information of the channel response, i.e. the estimated SIR of the channel is set to the actual SIR such that :

$$p'_{1,h} = p_{1,h} \quad \text{for} \quad h=0,1,2.....,L$$

The adjustable parameters d_0 and d_1 are set to *0.003* and 10^{-6} respectively as before, λ is selected to be *0.973*. The SNR used in this test is varying from *20* to *30dB*. The result of

computer simulation test is given in Table 7.3. It can be seen from the previous table that the system collapses at the SNR equal to *20* and *23dB* over channel 1 and 2 respectively, where at SNR greater than *23dB*, the system performs quite well and gives an acceptable result.

Increasing the value of λ to *0.976* improves the performance over channel 1 where over channel 2 there is a small degradation, as shown in Table 7.3. It is clear from the table that the system does not collapse at *SNR=20dB* when operating over channel 1. This is a very interesting result, since the system still works without a retraining process provided with adequate transmission power. The advantage gained here is no loss in the transmission rate and the complexity of the system is also reduced, since there is no need for the retraining process. Bear in mind that the fading predominates over the noise in the case of high SNR, i.e. most the errors that occur in detection are caused mainly by the fading.

Concluding Remarks

1. The combined system that uses estimator 3 with degree-1 prediction gives the best performance over the six systems considered here, and this result agrees with that of Chapter 5 and 6. Also the system gives quite close performance to 'upper bound' over both channels with the degradation on average about *1-3dB* and *3-5dB* for channel 1 and 2 respectively. A useful reduction in complexity of the MPA and estimator has been achieved without unduly changing the performance of the system. This result appears to be very encouraging for practical implementation.

2. The non-linear equaliser system has been tested under the same conditions and it has been found that its performance is inferior to the system that uses a NML detector.

3. Tests have indicated that, at error rates of around 10^{-3}, most of the errors occur in just one or three frames, where these frames correspond to the times when the signal is in the deepest fades, particularly when the system operates over the worst channel (channel 2). Moreover, during the deep fades any small noise will result in errors, and the errors starting to occur may continue to occur until the end of the frame. This is due to the low signal level, i. e. at low SNR *(10-20dB)*. Therefore, the errors can be reduced to a low value by using a high value of SNR, greater than *20* and *25dB*, for channel 1 and 2 respectively.

4. Using the adaptive estimator 3 with degree-1 prediction does improve the performance of the system, as shown in Figs.7.7.3-7.7.4 due to its ability to track the variation of time-varying channels better than other systems. Moreover the complexity of this estimator after calculation of the step size once every *8T* is not too large compared with the same estimator that uses a fixed step size. It also shows that the performance of the systems which use a channel estimator without prediction are much poorer than those with prediction, even in the case of square root Kalman estimator. The performance of the

latter estimator is improved by incorporating degree-1 prediction as shown in Figs.7.7.3-7.7.4, but this increases the complexity of the system even more.

5. When MPA and the calculation of the adaptive step size is carried out once every $8T$, it does not degrade the system bit error rate performance, but considerably reduces the complexity and makes the practical implementation of the system simpler.

6. The total number of the arithmetic operations has been calculated for every part of the receiver, and these are summarised in Table 7.1.

7. Finally, error-free performance can be achieved over both channels with adequate SNR. The error-free performance is obtained over channel 1 and 2 at SNR equal to *25* and *32 dB* respectively as indicated in Table 7.2.

The final conclusion of this chapter is that using the combined system 4 a very encouraging result has been obtained, since to the author's knowledge, there has been no system proposed which could give a similar result for *4800 bit/sec* data transmission over similar HF channels considered here with the same characteristics, and of course without using any sophisticated coding, diversity, ARQ and any form of interleaving.

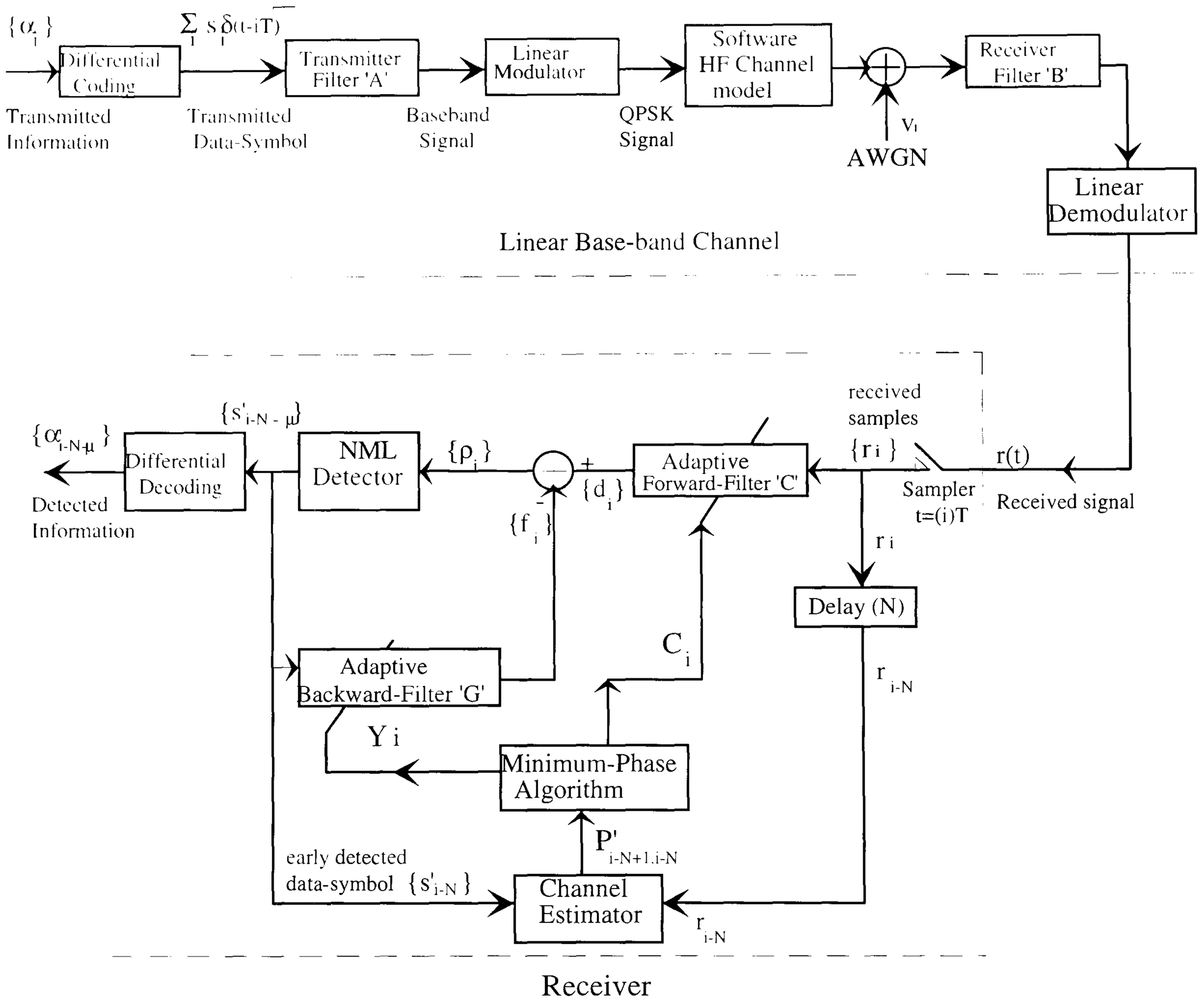


Fig.7.2.1 : The Model of Combined System used in the Tests.

	real multiplication	real addition or Sub.	inverse
every samples	11440	11320	80
every 2 samples	5720	5660	40
every 4 samples	2860	2830	20
every 8 samples	1430	1415	10
every 16 samples	715	708	5

'j=40, L+1=20, N+1=50, K=10 roots outside the required region in the unit circle'.

	real multiplication	real addition or sub.	inverse
every samples	9840	9720	80
every 4 samples	2460	2430	20
every 8 samples	1230	1215	10

'j=40, L+1=20, N+1=30, K=10 roots outside the required region in the unit circle'

a-Total number of arithmetic operations required by MPA.

	real multiplication	real addition or sub.	comparison
N+1=50,(L+1=20), k=16,k_{tot}=26	1472	1552	486
N+1=30,(L+1)=20, k=16,k_{tot}=26	1392	1472	486

b-Total number of arithmetic operations required by Detector3.
(including the calculation of the sample d_i and f_i).

real multiplication	real addition or sub.	comparison	SQRT
16(L+1)	13(L+1)+2	L+1	2(L+1)
320	262	20	40

c-Total number of arithmetic operations required by Estimator3.
(step-size is calculated every T)

real multiplication	real addition or sub.	comparison	SQRT
14(L+1)+2(L+1)/8	12(L+1)+((L+1)/8)+2	(L+1)/8	2(L+1)/8
285	245	3	5

d-Total number of arithmetic operations required by Estimator3.
(Step-size is calculated once every 8T)

TABLE 7.1 The Total Number of Arithmetic Operations Required by the Combined System.

SNR (dB)	λ	ζ_1 CS4M1N50	BER CS4M1N50	ζ_1 CS4M8N30ES8	BER CS4M8N30ES8
5	0.999	-0.370	$3.2*10^{-1}$	-0.4	$3.2*10^{-1}$
10	0.991	-2.52	$1.8*10^{-1}$	-2.14	$1.87*10^{-1}$
15	0.987	-11.06	$3.3*10^{-2}$	-11.153	$3.1*10^{-2}$
17	0.984	-15.22	$1.3*10^{-2}$	-14.37	$1.55*10^{-2}$
20	0.977	-22.5	$1.35*10^{-4}$	-22.44	$2.7*10^{-4}$
23	0.977	-25.2	$9*10^{-6}$	-25.12	$9*10^{-6}$
25	0.977	-26.8	0.0	-26.71	0.0

a-over channel 1.

SNR (dB)	λ	ζ_1 CS4M1N50	BER CS4M1N50	ζ_1 CS4M8N30ES 8	BER CS4M8N30ES8
5	0.999	0.265	$3.45*10^{-1}$	0.34	$3.45*10^{-1}$
10	0.991	-1.83	$2.3*10^{-1}$	-1.42	$2.47*10^{-1}$
15	0.987	-6.47	$8.6*10^{-2}$	-5.53	$1.05*10^{-1}$
18	0.982	-13.77	$2.36*10^{-2}$	-8.12	$4.0*10^{-2}$
20	0.977	-14.26	$1.8*10^{-2}$	-14.51	$1.6*10^{-2}$
23	0.977	-17.63	$5.9*10^{-3}$	-17.96	$6.47*10^{-3}$
25	0.970	-26.2	$1.8*10^{-4}$	-26.14	$1.9*10^{-4}$
28	0.970	-29.2	$4.5*10^{-5}$	-28.82	$5.4*10^{-5}$
32	0.970	-31.9	0.0	-31.7	0.0

b-over channel 2.

SNR (dB)	ζ_1 CS4M1N50	BER CS4M1N50	ζ_1 CS4M8N30ES8	BER CS4M8N30ES8
20	-19.0	$3.5*10^{-3}$	-19.3	$2.5*10^{-3}$
23	-25.2	$9*10^{-6}$	-24.4	$9*10^{-6}$
25	-26.8	0.0	-26.7	0.0

c-over channel 1, λ=0.973.

SNR (dB)	ζ_1 CS4M1N50	BER CS4M1N50	ζ_1 CS4M8N30ES8	BER CS4M8N30ES8
20	-11.6	$2.2*10^{-2}$	-11.5	$2.4*10^{-2}$
23	-14.0	$7.8*10^{-3}$	-13.4	$6.9*10^{-3}$
25	-26.7	$3.3*10^{-4}$	-26.6	$6.8*10^{-4}$
28	-29.2	$4.5*10^{-5}$	-29.16	$9.0*10^{-5}$
32	-31.8	0.0	-30.8	0.0

d-over channel 2, λ=0.973.

TABLE 7.2 Performance of the Combined System 4 over Channel 1 and 2.

λ	SNR (dB)	ζ_1 CS4M8N30ES8	BER CS4M8N30ES8	λ	ζ_1 CS4M8N30ES8	BER
0.973	20	system	collapse	0.976	-22.0	$3.9*10^{-4}$
0.973	23	-24.2	$8.7*10^{-6}$	0.976	-25.0	$8.7*10^{-6}$
0.973	25	-26.0	0.0	0.976	-26.6	0.0

a-over channel 1

λ	SNR (dB)	ζ_1 CS4M8N30ES8	BER CS4M8N30ES8	λ	ζ_1 CS4M8N30ES8	BER
0.973	20	system	collapse	0.976	system	collapse
0.973	23	system	collapse	0.976	system	collapse
0.973	25	-26.7	$3.3*10^{-4}$	0.976	-26.9	$4.6*10^{-4}$
0.973	28	-29.2	$4.5*10^{-5}$	0.976	-29.3	$1.1*10^{-4}$
0.973	30	-31.8	0.0	0.976	-30.5	$6.9*10^{-5}$

b-over channel 2.

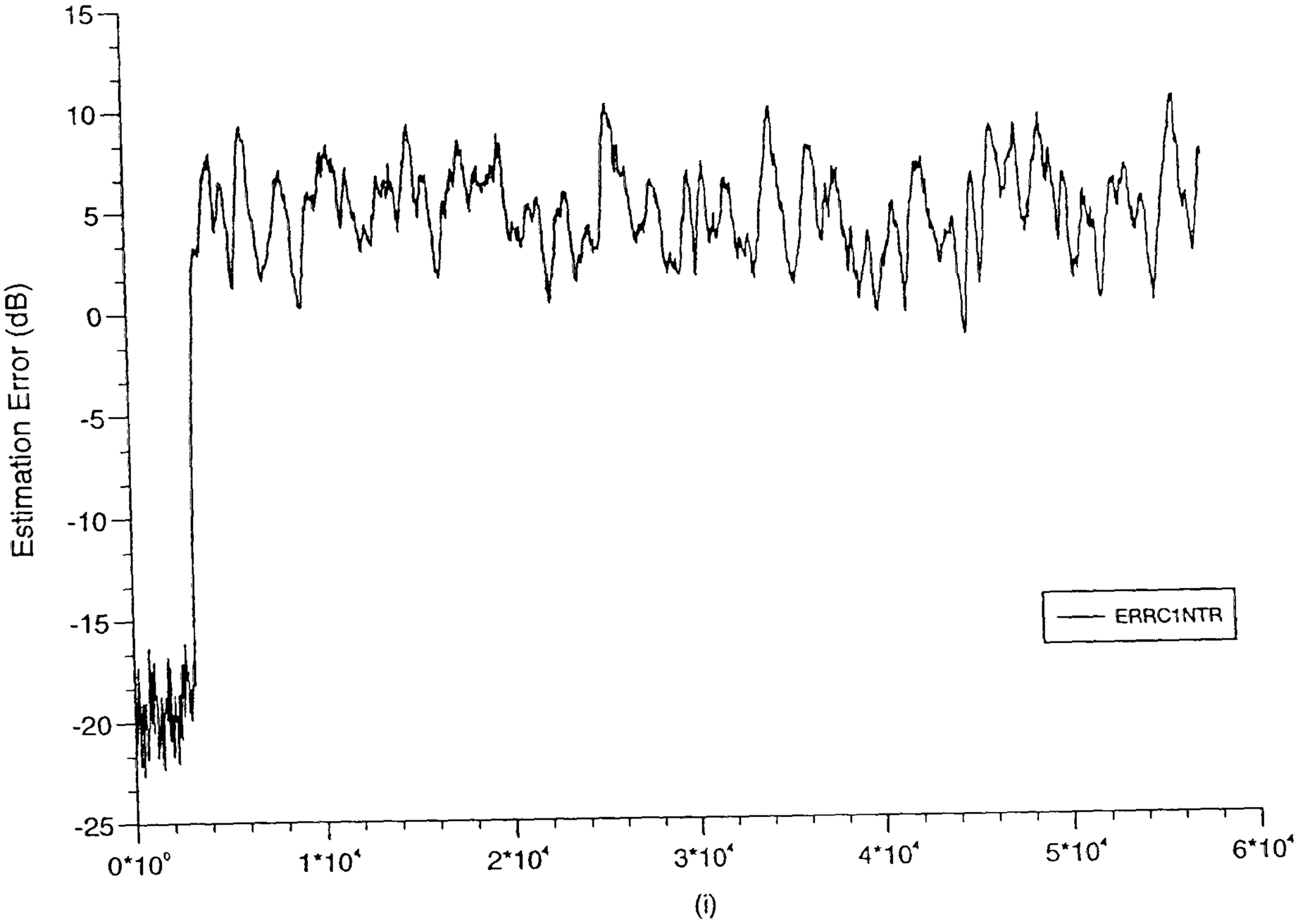
TABLE 7.3 Performance of the Combined System 4 over Channel 1 and 2. without retraining process.

Note:

The symbol in the table is denoted to the system under tests as :

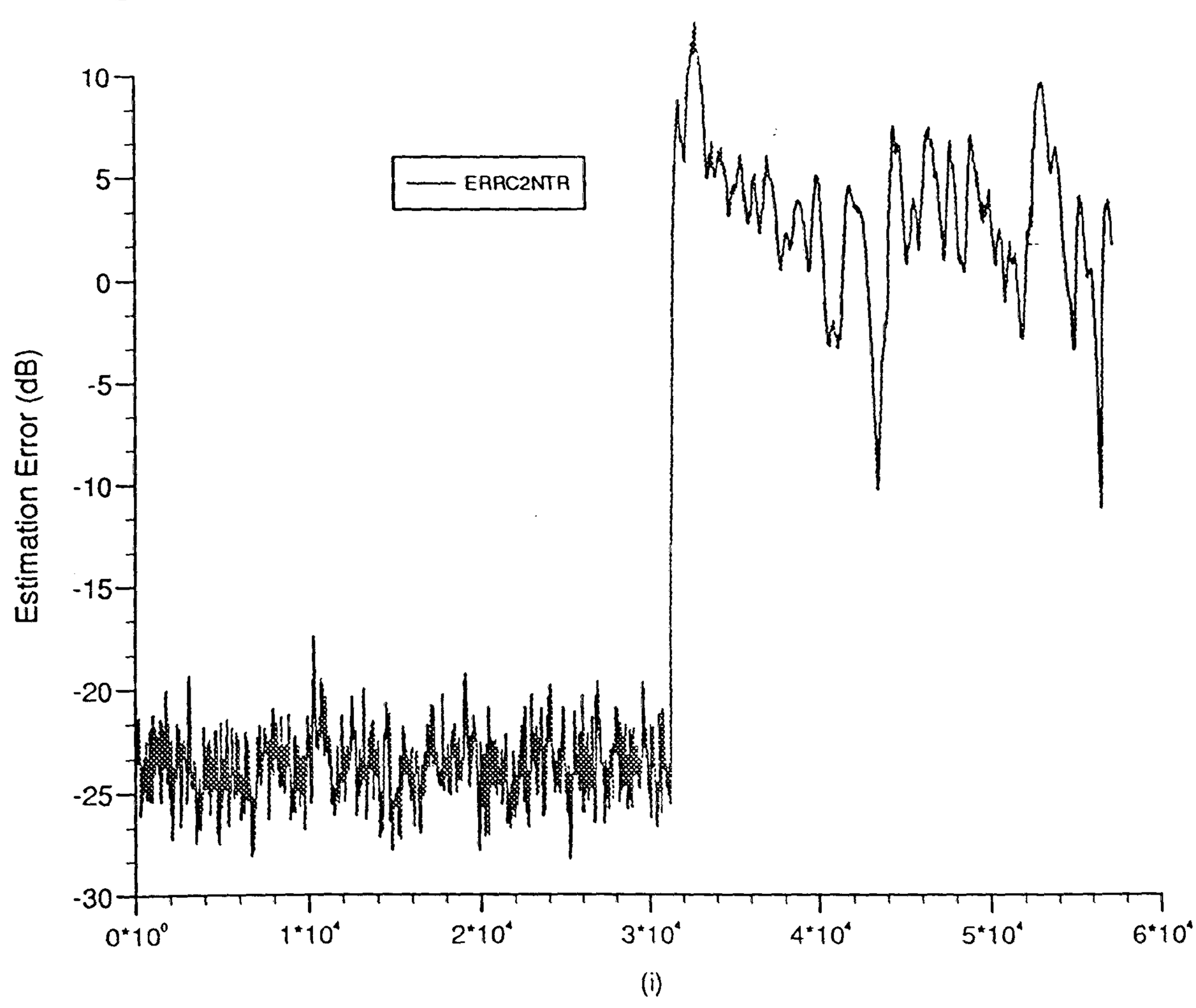
CS4M1N50: Combined detector 3, estimator 3 with degree-1 prediction and adaptive equaliser with (j=40, d=1.05), where MPA and estimator is carried out every sampling interval.

CS4M8N30ES8: Combined detector 3, estimator 3 with degree-1 prediction and adaptive equaliser with (j=40, d=1.05), where MPA and calculation of the step-size is carried out once every 8 sampling intervals.



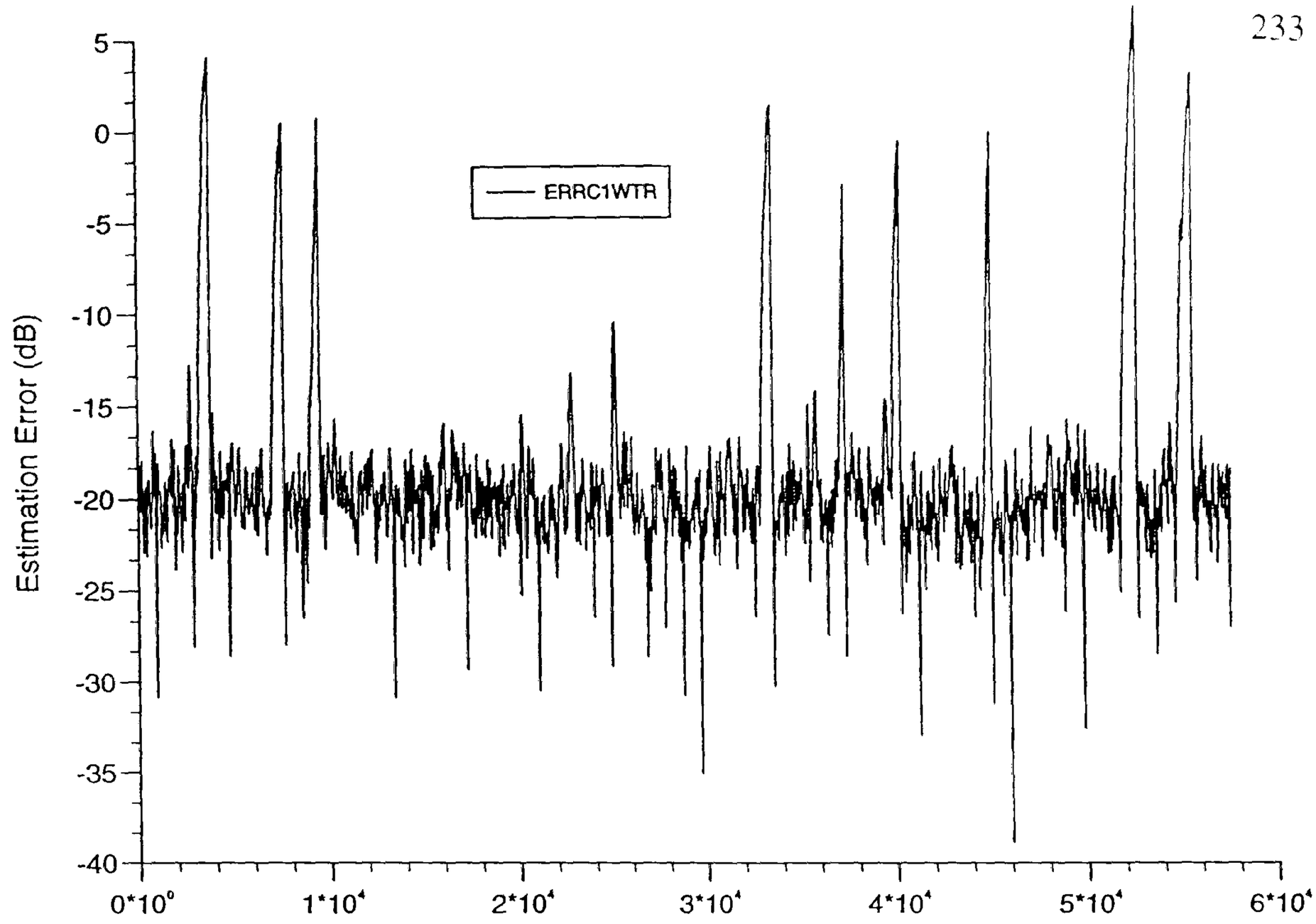
(SNR=15dB, λ=0.987 and without training).

Fig.7.7.1-a Error in the Estimation of SIR of Channel 1, using Estimator3.



(SNR=20dB, λ=0.983 and without training).

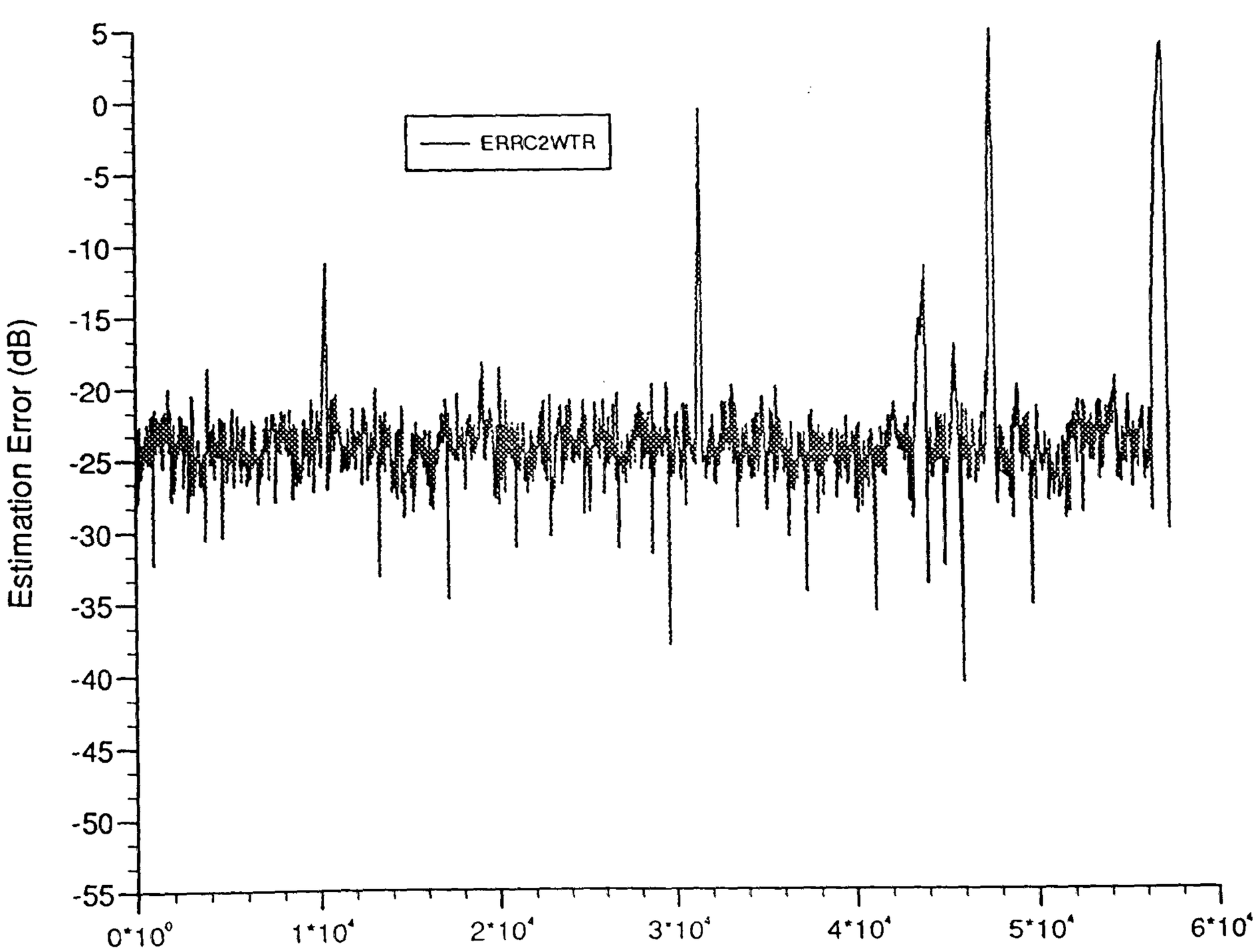
Fig.7.7.1-b Error in the Estimation of SIR of Channel 2, using Estimator3.



(i)

(SNR=15dB, λ=0.987 and with training).

Fig.7.7.2-a Error in the Estimation of SIR of Channel 1, using Estimator3.



(i)

(SNR=20dB, λ=0.983 and with training).

Fig.7.7.2-b Error in the Estimation of SIR of Channel 2, using Estimator3.

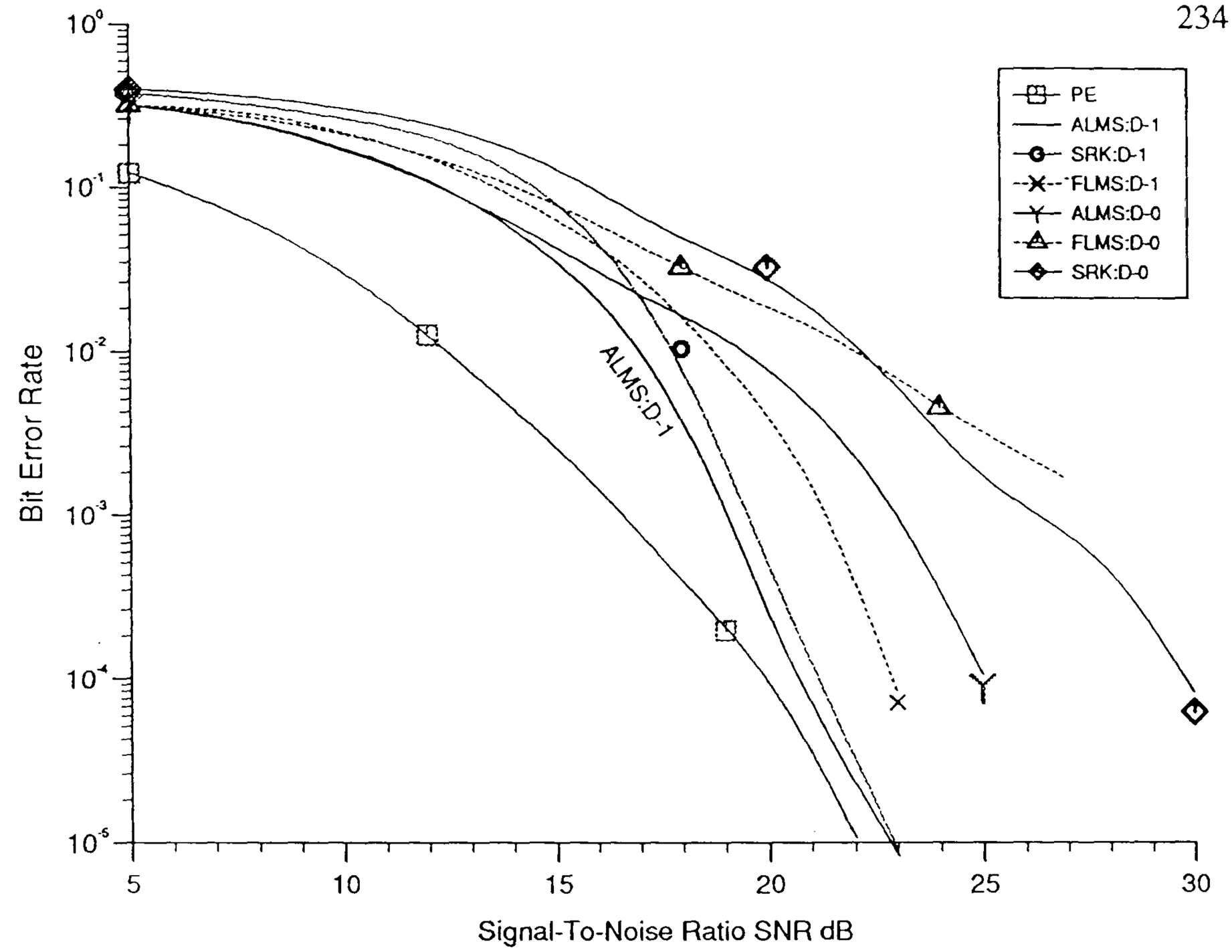


Fig.7.7.3 Performance of Combined Detector 3, Estimators and Adaptive Equaliser over Channel1, with (d=1.05, iter. number (j)=40 and MPA every T)

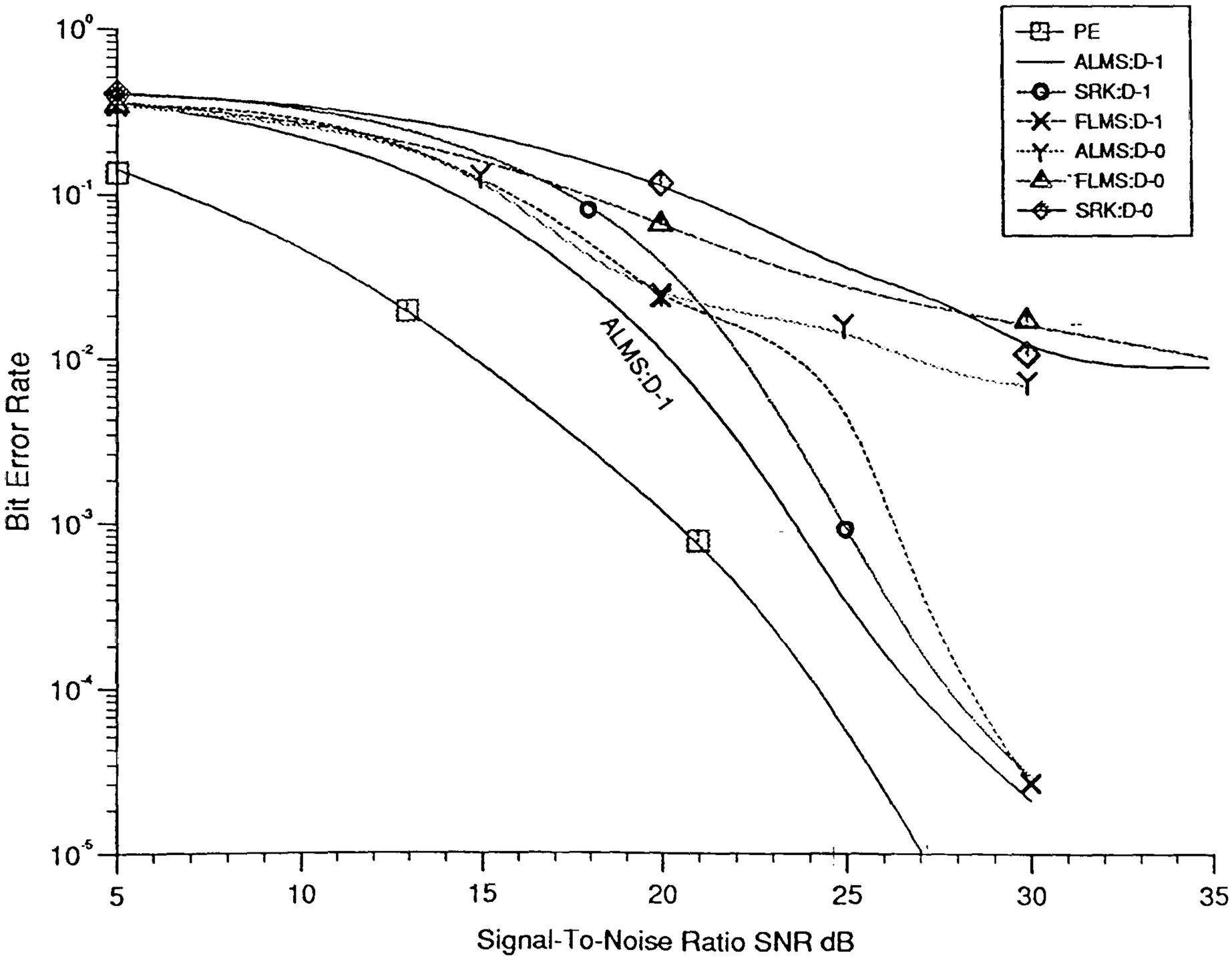


Fig.7.7.4 Performance of Combined Detector 3, Estimators and Adaptive Equaliser over Channel 2, with (d=1.05, iter. number=40 and MPA every T)

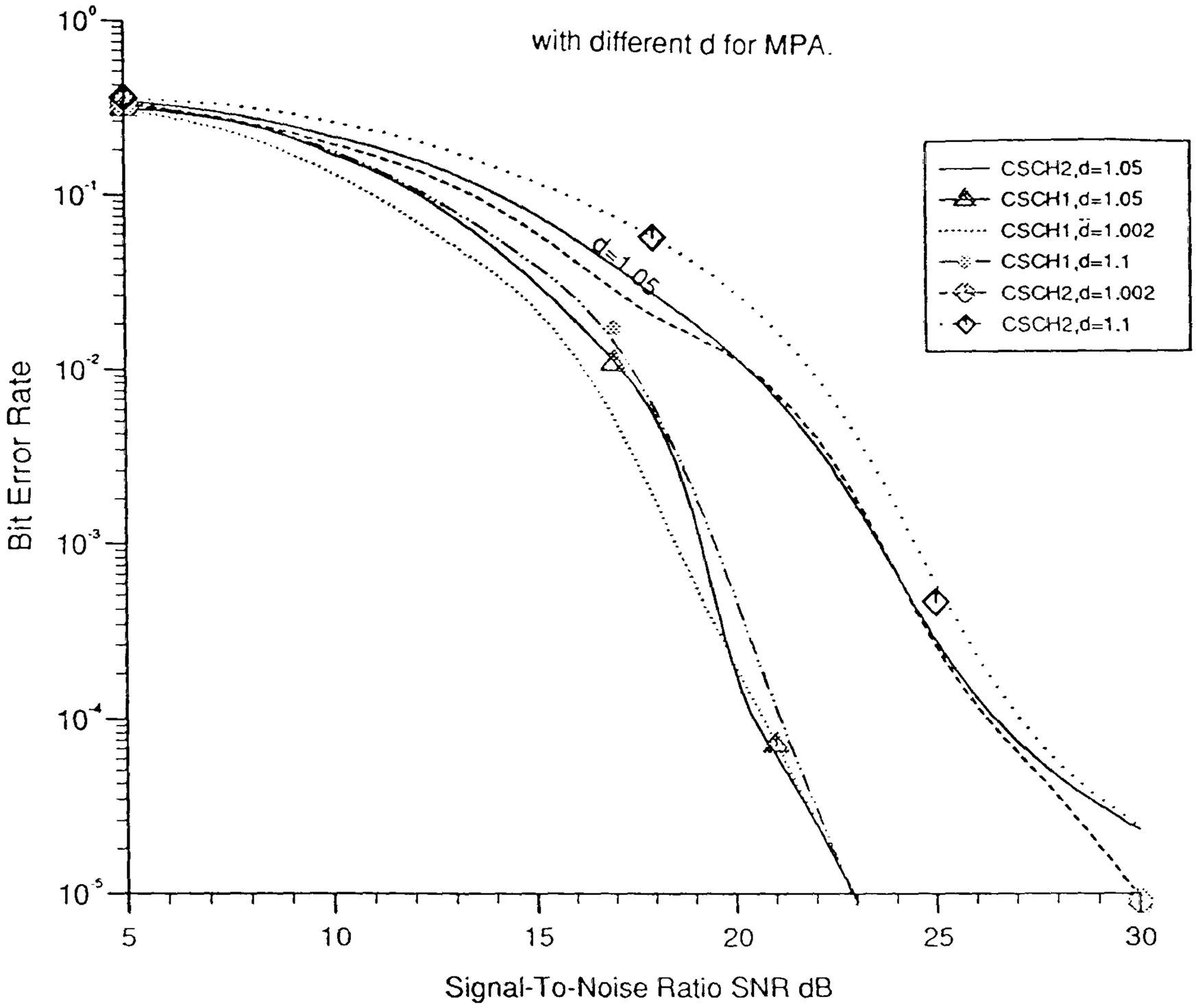


Fig.7.7.5 Performance of Combined Detector 3, Estimator 3 and Adaptive Equaliser over Channel1 and 2, with (iter. number=30 and MPA every T).

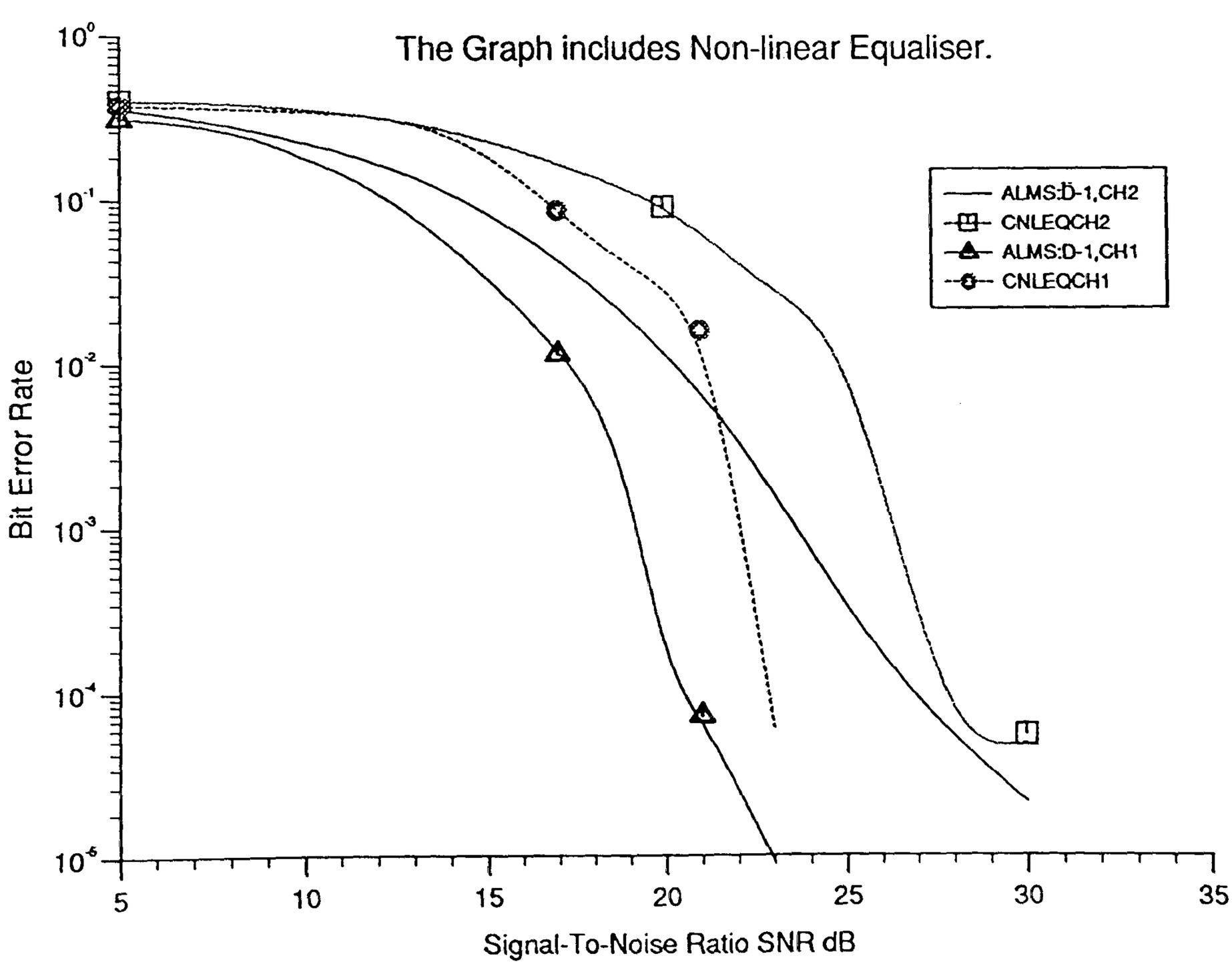


Fig.7.7.6 Performance Comparision of Combined Detector 3, Estimator 3 and Adaptive Equaliser with Combined Non-linear Equaliser system when operating over Channel 1 and 2, with (d=1.05, iter. number=40 and MPA every T)

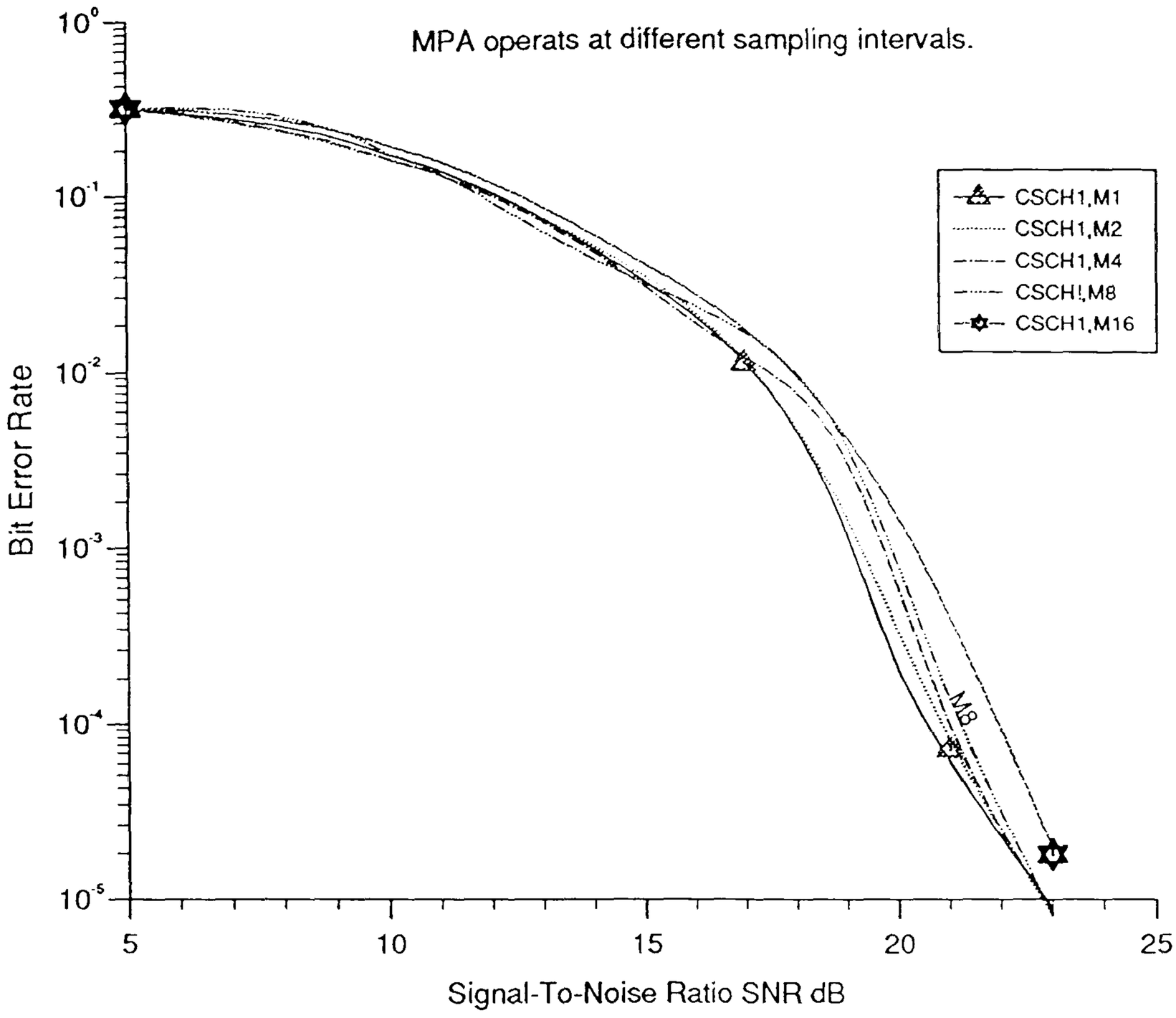


Fig.7.7.7 Performance of Combined Detector 3, Estimator 3 and Adaptive Equaliser over Channel 1, with (d=1.05, iter. number=40 and MPA once every MT).

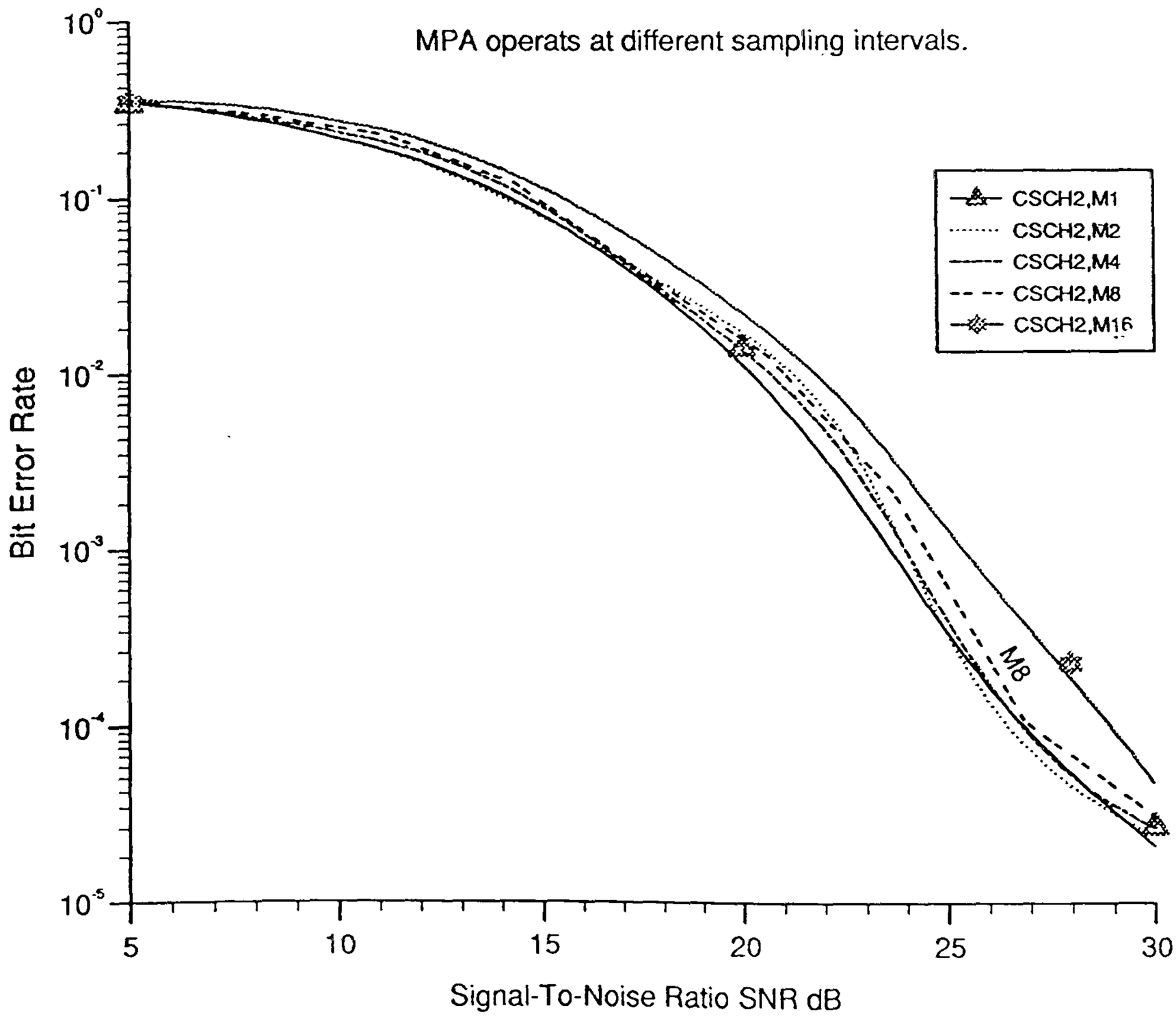


Fig.7.7.8 Performance of Combined Detector 3, Estimator 3 and Adaptive Equaliser over Channel 2, with (d=1.05, iter. number=40 and MPA once every MT).

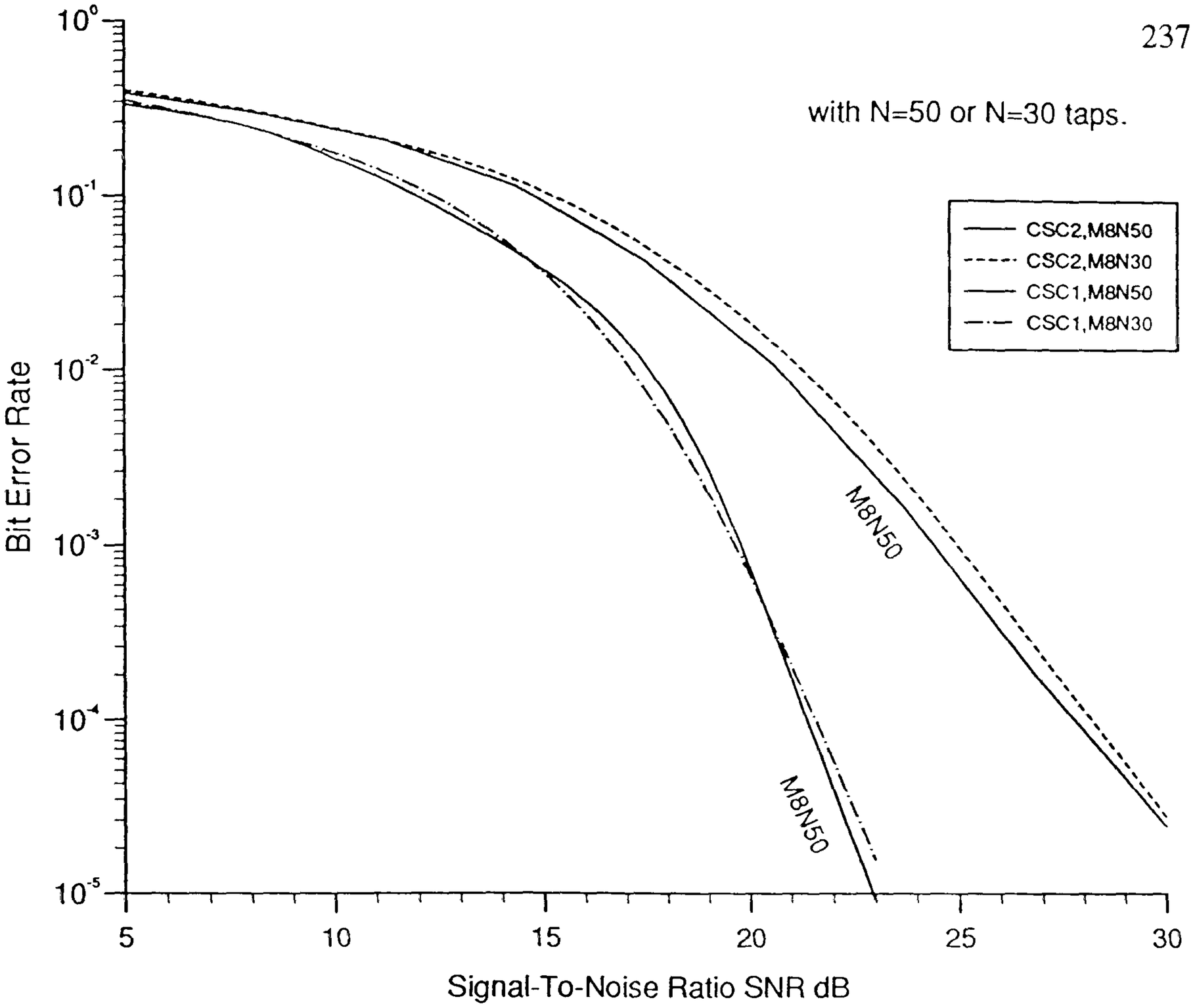


Fig.7.7.9 Performance of Combined Detector 3, Estimator 3 and Adaptive Equaliser over Channel 1 and 2, with (d=1.05, iter. number=40 and MPA once every 8T)

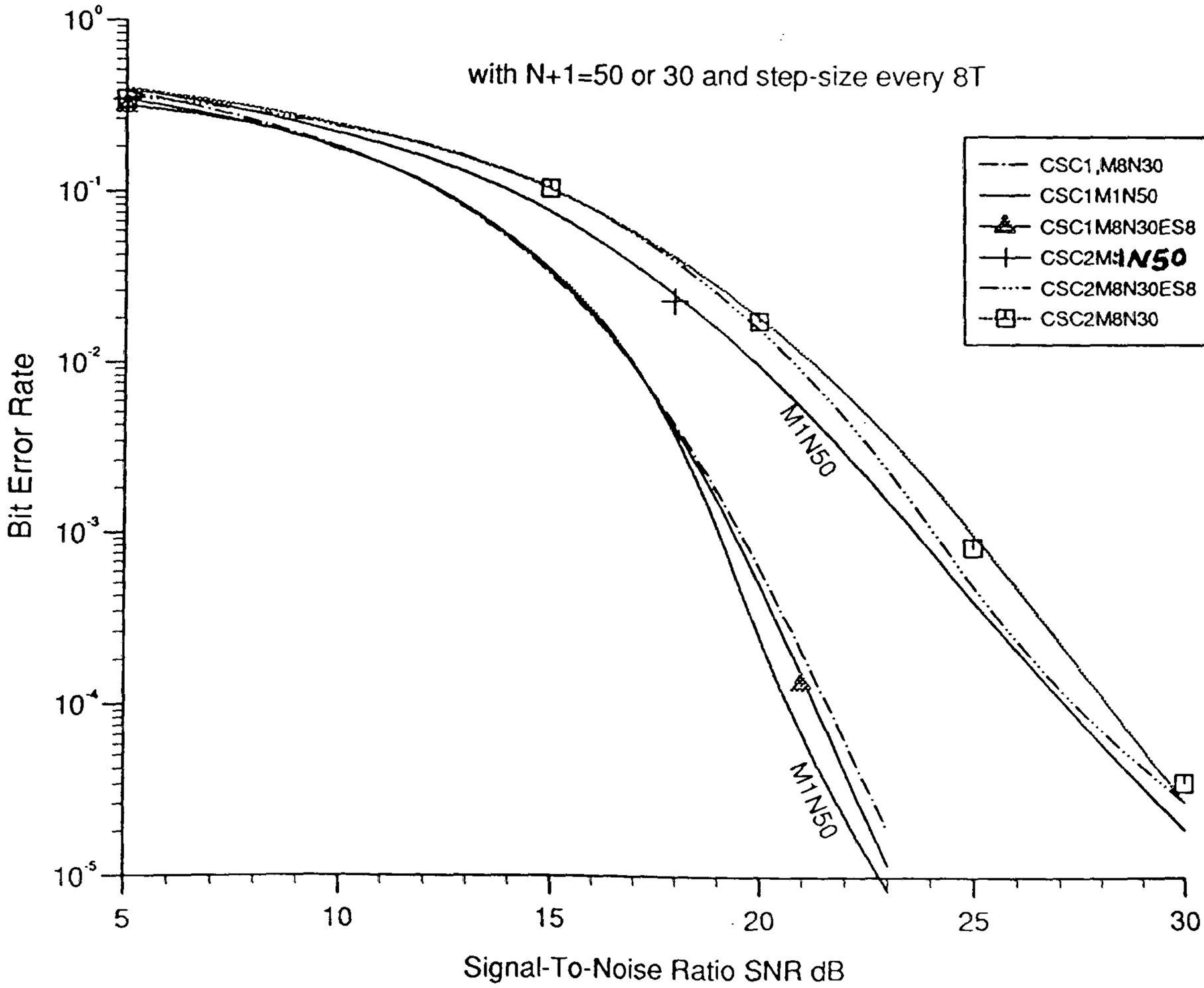


Fig.7.7.10 Performance of Combined Detector 3, Estimator 3 and Adaptive Equaliser over Channel 1 and 2, with (d=1.05, j=40 and 'MPA, step-size' once every 8T)

CHAPTER 8

CONCLUSIONS AND FURTHER WORK

8.1 CONCLUSIONS

The investigation has been concerned with the development of the detectors, channel estimators and adaptive equaliser, for a serial modem at a data rate of *4.8kBits/s* over HF channels.

The adaptive adjustment of the receiver used in this thesis is based on the algorithm found by Clark and Hue [39], for use over a telephone channel. A suitable modification of this algorithm has been made for operating over a time-varying HF channel. In this method a root-finding (minimum phase) algorithm (MPA) is employed to determine, in sequence, the roots (zeros) of the Z-transform of the sampled impulse response (SIR) of the channel, whose absolute values are greater than d (where d in the range 1 to 1.5). It then uses a knowledge of these roots to determine the tap-gains of the pre-detector filter and to form an estimate of the SIR of the channel and filter, that should now be near-minimum phased. The algorithm is also used to adjust the tap-gains of a decision feed-back equaliser (DFE). In Chapter 4 the performance of the MPA was observed assuming perfect estimation, over channels 1, 2, 3 and 4 with known SIR. It was shown that the MPA algorithm operates quite well over all channels and it is suitable for converting the non-minimum phase channel to minimum (or near-minimum) phase. Although the minimum phasing of the channel is not critical in a full Viterbi detector, it is desirable for the near-maximum likelihood (NML) detectors of the type considered here, (essential in the case of a conventional DFE).

In Chapter 5 three types of NML detectors have been tested assuming perfect estimation and the channel is made minimum (or near-minimum) phase using the MPA that described in Chapter 4. The performance of these detectors has been compared with that of the corresponding decision feedback equaliser, and the detector that gives the best compromise between performance and complexity was selected for further tests. Simulation results have shown that all the NML detectors tested here give better performance than DFE. At low error rates, the latter is between 1 and 4 dB's worse than the worst of the NML detectors. The result also indicates that detector 3 achieves the best compromise between performance and complexity.

In Chapter 6 the task of channel estimation was tested, assuming correct detection of all data symbols. The channel estimators are either based on the least-mean-square (LMS) algorithm or the modified recursive-least-square (RLS) algorithm (square-root kalman algorithm SRKA). Simulation results have shown that the simpler LMS algorithm

gives a comparable level of accuracy with the more complex SRK algorithm, though its start-up convergence is slower. The reason for non-optimum performance of the RLS algorithm has been attributed to the assumption by the Kalman algorithm that the channel is modelled as time-invariant, which is not the case in the HF channel considered here. The simulation results have shown that the best performance of the estimators is given by estimators 3 and 5 with degree one prediction. The poorest performance is given by estimator 1 (LMS) and 4 (SRK) without prediction, especially at high SNR. Thus, employing degree-1 prediction that takes into account the rate of change in the SIR of the channel, improves the performance of the estimator over the whole range of SNR. At low SNR, which is of the greatest practical importance, estimator 3 gives the best performance. The computational complexity required by estimator 3 is much less than that required by estimators 4 or 5. The channel estimators listed in order of complexity and starting with the simple estimator are estimator 1, 2, 3, 4 and 5. Therefore, as the most promising as the compromise in terms of performance and complexity, estimator 3 is chosen to be the preferred estimator for the given applications in this thesis.

Finally, various arrangements of the combined systems have been tested, when all the functions of detection, estimation and adaptive linear filtering are included. The combined system uses detector 3 chosen from previous tests, five estimators described in Chapter 6 and the minimum phase algorithm to adjust the tap-gains of the adaptive filter. It has been found that for correct operation of the combined receiver under severe conditions such as prolonged periods of severe flat fading (deep-fades), there should be regular retraining for the receiver to prevent a total collapse of the estimation process. It is also found that when retraining is performed at intervals of *960 symbols*, all the combined systems can recover from the total collapse. Simulation results have shown that the best performance is given by system 4 (ALMS:D-1) which uses detector 3, adaptive step size LMS estimator with degree-1 prediction, and MPA. It is observed that the degradation at error rate of 10^{-4} is about *1.5-2.5dB* and *3-4dB* over channel 1 and 2 respectively compared to the upper-bound case. Therefore, combined system 4 is selected as the preferred system in terms of performance and complexity for further tests. Extensive computer simulation tests have been carried out on system 4, and the results have shown that further reduction in complexity is achieved in combined system 4 without unduly affecting the performance of the system. This has led to a novel combined system that operates quite well over both channels but is not unduly complex, which makes the practical implementation of such a system feasible. The results also indicate that error free performance can be achieved over both channels given adequate transmission power. It is demonstrated that system 4 can operate without a training sequence at high SNR. The final conclusion is that the proposed system is a potentially useful design for the receiver of a *4800bit/sec* digital data transmission system over HF radio links, using the new technology of high-speed digital signal processing (DSP) such as TMS320C30 or C40.

8.2. SUGGESTIONS FOR FURTHER WORK

1. Hardware Implementation

The techniques considered in this thesis are algorithms or processes that operate on a sequence of sample values. Modern digital modems normally operate in this way and the techniques described are a direct application to practical modems. The combined system can be realised in block by block structured programming in software as presented in Appendix G. The digital system can be implemented in hardware using the new technology of high-speed digital signal processing (DSP) such as TMS320Cxx family. The class of operations to be performed by the receiver algorithms such as multiply, shift, add or subtract, etc., are DSP-type operations, therefore TMS320C30 or C40 can be of great use for the hardware realisation of the systems.

The digital signal processor TMS320C30 and 40 are high-performance CMOS-32bit floating point devices and they support many features [115-117] such as :

. The 50-nsec cycle time of the TMS320C30-40 allows it to execute operation at a performance rate up to *40 MFLOPS* or *20 MIPS*. The generation performance is further enhanced through its large on-chip memories, concurrent DMA controller, and two external interface ports. It can perform parallel multiply and ALU operations on integer or floating-point data in single cycle.

. The TMS320C3x advanced interface design can be used to implement a wide variety of system configurations. Its two external buses and DMA capability provide a parallel 32-bit interface to external devices, while interrupt interface, dual serial ports and general-purpose digital I/O provide communication with a multitude of peripherals. The expansion bus can be used to implement data transfers concurrently with and independent of operations on the primary bus.

. A high-Level-Language such as C can be used to implement the algorithms. The designer can optimise execution by creating very fast assembly-language routines that implement the time-critical sections, and call them from C as regular functions. The C compiler supports the insertion of assembly language code into C-source code. The C-functions may be called from assembly language. Variables defined in the C-source may be accessed in assembly language modules and vice versa. The result is a compiler that allows user to tailor the amount of high-level programming versus the amount of assembly language according to his applications. The C-compiler is supported on the TMS320C30.

These features and many others make the device very powerful for digital implementation of modern digital modems. The possible hardware configuration of the DSP (TMS320C30) is shown in Fig.8.1. Analogue-to-digital (ADC) and digital-to-analogue (DAC) are components that are commonly required in DSP systems. For modem applications using DSP, a complete ADC and DAC input/output system on a single chip

such as the TLC320C44 or 47 may be appropriate. The TLC320C44 or 47 analogue interface circuit (AIC) integrates a bandpass, switched-capacitor, anti-aliasing-input filter, 14bit resolution ADC and DAC converters, and a lowpass, switched capacitor, output-reconstruction filter, all on a single monolithic CMOS chip. It offers numerous combinations of master clock input frequencies and conversion/sampling rate (up to *19000* and *25000Hz*), which can be changed via DSP control.

The proposed combined receiver described in Chapter 7 is ready for practical implementation. The functional block diagram of the transmitter and receiver is shown in Fig.8.2.

The QPSK modulator and demodulator in Fig.8.2 are relatively simple devices and they can be implemented using DSP.

The total number of arithmetic operations per sampling interval was calculated in Chapter 7 for each part of the combined receiver. For example consider the sampling rate equal to *8000Hz*, hence the available processing time is T=*1/8000=0.125msec*. This is equivalent to *5MFLOPS or 2.5MIPS* for a 50-ns DSP. The SQRT and INVERSE operations can be performed in *138* and *76* machine cycle respectively with *10* decimal digits approximate accuracy using TMS320C30 [117]. Therefore, the total number of arithmetic operations required by estimator, detector and MPA is *1238, 4322, 3205* operations respectively. These operations do not include the storing and retrieving of values in memory.

From the above discussion, a single processor (DSP) will not be powerful enough (or fast enough) to perform all the functions of the combined receiver, and so separate processor operating in parallel would probably be necessary. Thus the combined receiver could require 4 to 5 DSP. The possible DSP structure of the system is shown in Fig.8.3.

When frame by frame transmission is performed, the use of FIFO (first in first out) buffers, each of which can hold in store at least one frame of received samples, gains some advantages. The receiver simultaneously processes the complete set of received samples from any one frame, while the received samples of the next frame gradually fill up the second buffer. At the end of the frame the receiver switches to operate on the second buffer, leaving the first buffer to be filled by the newly arrived samples from the following frame, and so forth. This feature can provide delays required by the various sub-systems of the receiver, such as adaptive filter ahead of the detector. This enables the channel estimator to perform its function by a delay of (N) samples relative to those fed to the filter. It also allows efficient use to be made of the available computing power of the receiver DSPs.

Since the MPA is independent of all other activities in the receiver, the algorithm may be carried out 'off-line'. The estimator and detector work in sequence (the estimator begins when detector finishes and vice-versa). This would largely simplify the implementation, since one DSP will be assigned to one task only. Finally the use of

concurrent programming may save a considerable number of operations in a practical system.

The HF channel simulator in Fig.8.2 whose details are given in Chapter 2 can be implemented using the same type of DSP, and the function block diagram of a practical simulator is illustrated in Fig.8.4. The sampling frequency for example can be selected to be *8000Hz*, for HF simulator functions. The only restriction is that, the signal $q_i(t)$ should be sampled at *50Hz* for the reason that the poles of a Bessel filter should be at a locations in the z-plane at an adequate distance from the unit circle to prevent instability. Then linear interpolation is adequate to raise the sampling rate of $q_i(t)$ to the required sampling rate (*8000Hz*), i.e. the step of interpolation is *160*. The feedforward transversal Hilbert filter (H-F) for voice band signals must have a pass band as flat as possible over *400-4600Hz*, and the number of its taps could be *27-33taps*, which are adequate for this application. Using FIFO input and output buffers allows efficient use to be made of the available computing power of the DSP. Thus the input samples from ADC are stored in the array input buffer and the output samples are fetched from the array output buffer. The length of each buffer could be chosen to be *160* words length.

Finally, the HF modem can be implemented using parallel DSP such as *TMS3230C40* for simple hardware implementation in one circuit, which is important in portable equipment. Various methods of parallel processing are available, and Fig.8.5 shows one of these method using *TMS320C40*. The parallel processor may be able to perform other functions such as AGC, carrier phase and timing tracking during the main functions of the modem. Therefore, it would be worthy to investigate the most cost effective hardware design of the proposed system using DSPs.

2. Topics for Further Investigation

. A simple ideal retraining process is used in the combined HF receiver considered in this thesis to prevent the total collapse of the channel estimator under severe conditions such as a prolonged high level burst of noise or a prolonged period of severe flat fading (deep fade). Further investigations into applying a suitable and practical retraining process would be very useful for the combined system.

. Only differential coding is considered in this thesis. Therefore the potentials for coding and interleaving gains are a topic for further work, since suitable coding and interleaving schemes would be very useful in improving the system error rate performance under severe conditions.

. Further work may be carried out on the problem of carrier phase recovery and symbol timing (receiver synchronisation), and it would be interesting to observe how much the performance of the combined system degrades when there is also a frequency offset (or Doppler shift).

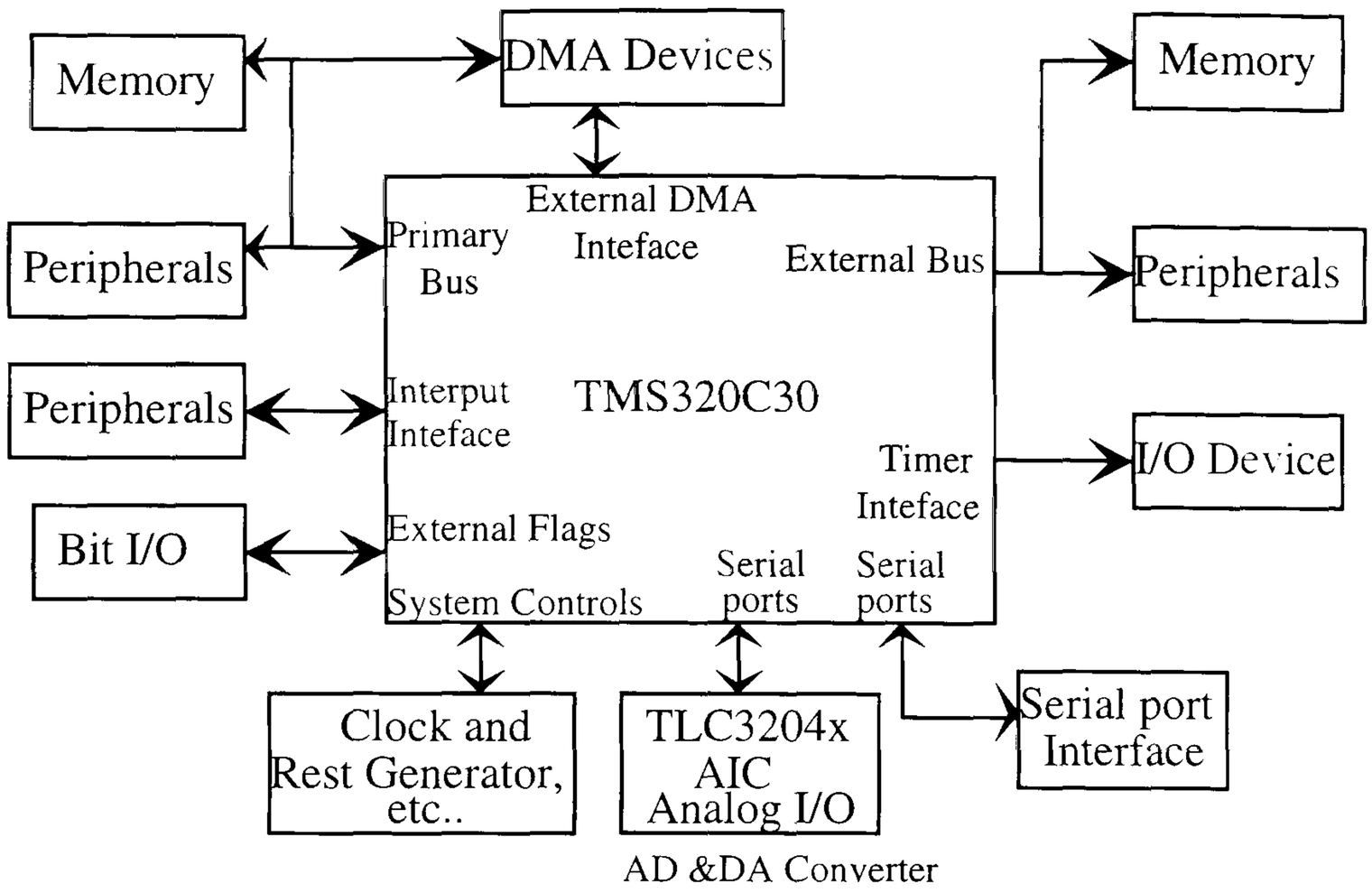
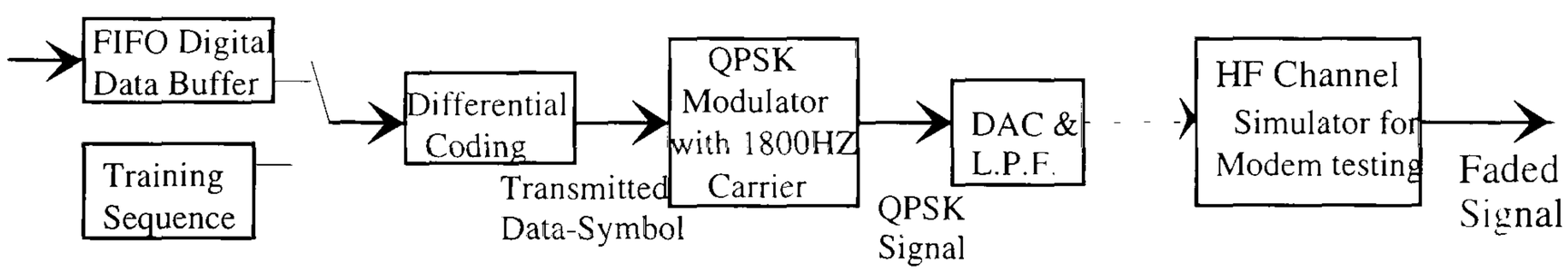
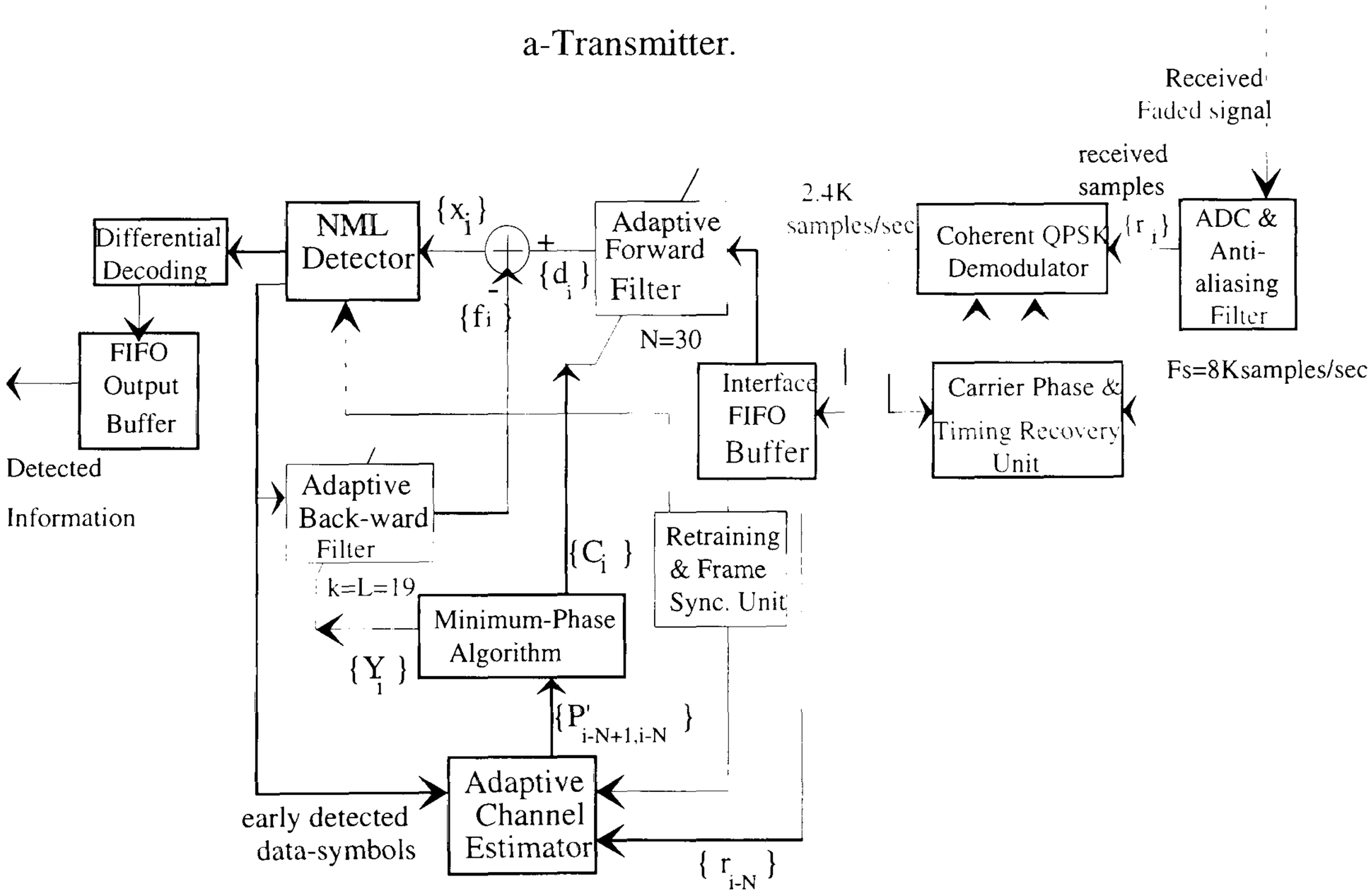


FIG. 8.1 Possible System Configuration.



a-Transmitter.



b-Receiver

FIG. 8.2 Functional Block Diagram of Transmitter and Receiver.

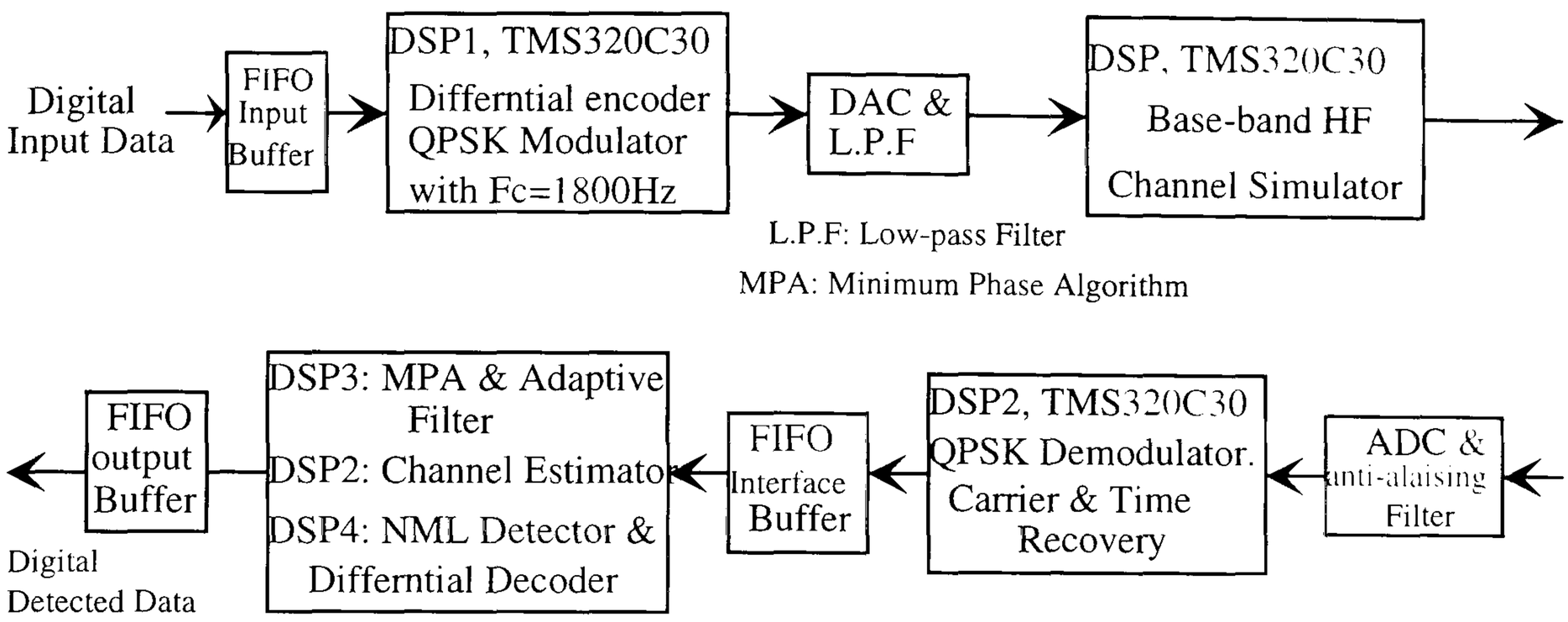


FIG. 8.3 DSPS Structure of The System.

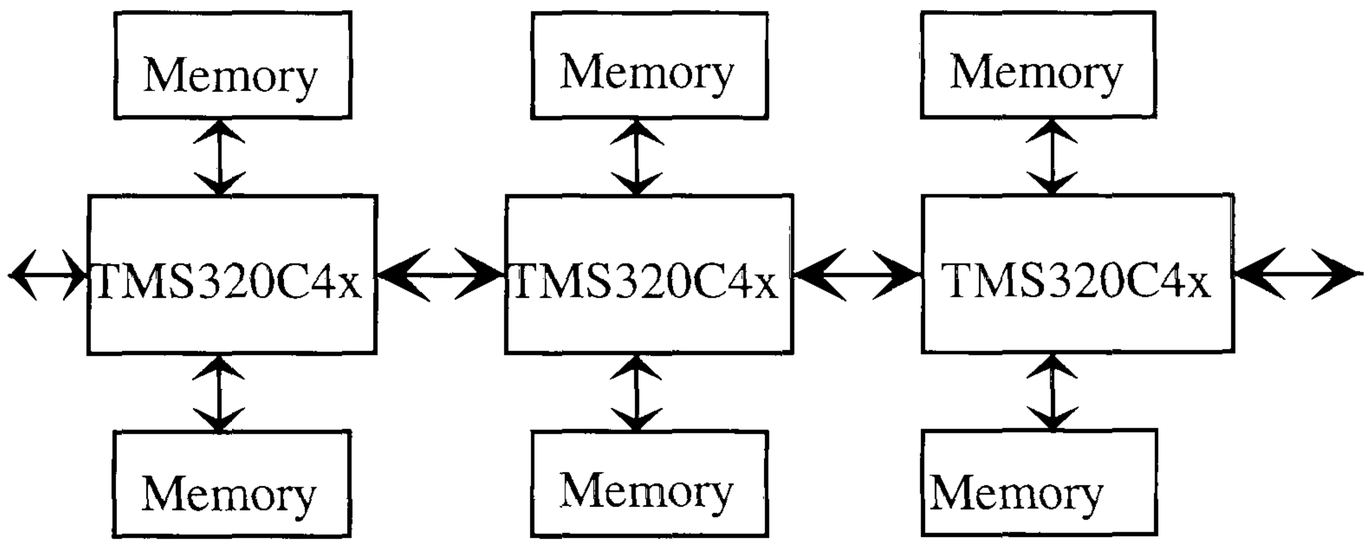
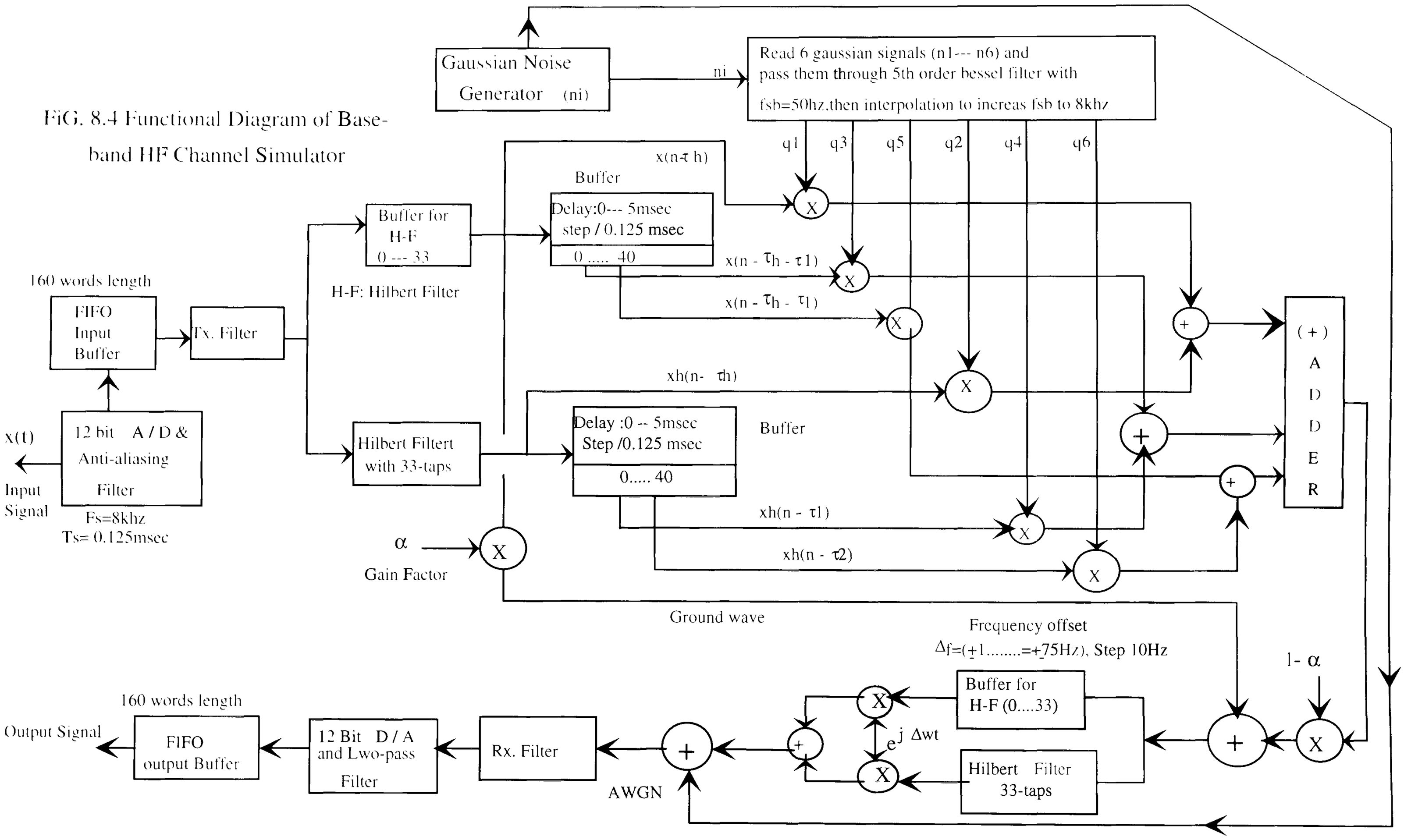


FIG. 8.5 Pipeline Linear Array Configuration of Parallel Processing.

FiG. 8.4 Functional Diagram of Base-band HF Channel Simulator



APPENDIX A

DERIVATION OF RAYLEIGH FADING FILTER 5-POLE BESSEL FILTER

As shown in chapter 2, the random process $q_1(t)$ is generated by filtering a zero mean white Gaussian noise $N_1(t)$. The filter used in Fig.2.8.5 has a Gaussian frequency response and is given by :

$$|Q_1(f)|^2 = |Q_2(f)|^2 = \exp[-\frac{F^2}{2F_{rms}^2}] \qquad \text{A.1}$$

Let the frequency response of each filter (Bessel filter) be W(f) then :

$$W(f) = \exp[-\frac{F^2}{4F_{rms}^2}] \qquad \text{A.2}$$

To calculate the cut-off frequency (F_c)

$$W(f) = 3dB \text{ or } W(f) = \frac{1}{\sqrt{2}} \text{ then } 0.707 = \exp(-\frac{F_c^2}{4F_{rms}^2}) \Rightarrow -\frac{F_c^2}{4F_{rms}^2} = \ln(0.707) = -0.346574 \Rightarrow$$

$$F_c = 1.17741\,F_{rms} \qquad \text{A.3}$$

But from the relation between the *rms* frequency and the frequency spread we have :

$$F_{sp} = 2\,F_{rms} \qquad \text{A.4}$$

from eqn.A.3 and A.4 then :

$$F_c = 1.17741\frac{F_{sp}}{2} = 0.588705\,F_{sp} \qquad \text{A.5}$$

The impulse response and the magnitude response tend toward Gaussian as the filter order is increased. Therefore a fith-order bessel filter has been used to obtain the Rayleigh fading filter [19]. The bessel filter have the transfer function in S-plane of the form [18,19] :

$$H(s) = \frac{A_0}{B_n(s)} \qquad \text{A.6}$$

where A_0 is a normalising constant and is chosen such that $H(0)=1$ and given in this formula :

$$A_0 = \frac{(2n)!}{2^n \cdot n!} \qquad \text{A.7}$$

where n is the order of the filter.

$B_n(s)$ is the $n^{th}-$order bessel polynomial defined by the formulas [18]:

$$B_0 = 1 \quad , \quad B_1 = s+1$$
$$B_n(s) = (2n-1)\cdot B_{n-1} + s^2 \cdot B_{n-2} \qquad \text{A.8}$$

or :

$$B_n(s) = \sum_{k=0}^{n} A_k \cdot s^k \qquad \text{A.9}$$

where : $A_k = \dfrac{(2n-k)!}{2^{n-k}\cdot k!(n-k)!}$ for k=0,1,2,.....,n. A.10

A 5^{th} – order bessel filter has been chosen as a practical choice and thus n=5, therefore the transfer function of this filter becomes :

$$H(s) = \frac{945}{s^5+15s^4+105s^3+420s^3+945s+945} \qquad \text{A.11}$$

at $k = 0 \Rightarrow$ from eqn.A.10 then : $A_0 = \dfrac{10!}{2^5 \cdot 5!} = 945$

Eqn.A.11 can be expressed as :

$$H(s) = \frac{945}{\prod_{i=1}^{5}(s-p_i)} \qquad \text{A.12}$$

where p_i are the poles of $H(s)$ and are given by [18] :

$$p_1 = -3.64674 + j0$$
$$p_2, p_3 = -3.35196 \pm j1.74266 \qquad \text{A.13}$$
$$p_4, p_5 = -2.32467 \pm j3.57102$$

To obtain the frequency response of the bessel filter we substitute $(s = j\Omega)$ in eqn.A.12 then :

$$H(j\Omega) = \frac{945}{\prod_{i=1}^{5}(j\Omega-p_i)} \qquad \text{A.14}$$

where Ω is the angular frequency and $j = \sqrt{-1}$. The $(3dB)$ cut-off angular frequency Ω_c is given by [17] : $\Omega_c = 2.4274 \ \ rad/\sec$ A.15

Ω_c is calculated from eqn.A.11 by taking the modules of $H(j\Omega_c)$ equal to the $1/\sqrt{2} = 0.707$ and solving the equation we obtain the value of Ω_c as in eqn.A.15 :

$$|H(j\Omega_c)| = \left|\frac{945}{(j\Omega_c)^5+15(j\Omega_c)^4+105(j\Omega_c)^3+420(j\Omega_c)^2+945(j\Omega_c)+945}\right| = 0.707$$

or at 3dB : $20\cdot\log_{10}(0.707) = 3dB$

One of the important parameter in the characterisation of a channel is the frequency spread F_{sp}. Thus, it is desirable to relate the cut-off frequency of the bessel filter to the frequency spread.

let : $W = a_0 \cdot \Omega$ A.16

But the constant a_0 is defined by

$$a_0 = \frac{W_c}{\Omega_c} = \frac{2\pi \cdot F_c}{\Omega_c} = \frac{2\pi \cdot F_c}{2.4274} = 2.58844 \cdot F_c \Rightarrow$$

$$a_0 = 2.58844 \cdot F_c \qquad \text{A.17}$$

Substituting Ω by W/a_0 in eqn.A.14 then :

$$H(j\frac{w}{a_0}) \equiv H(jw) = \frac{945}{\prod_{i=1}^{5}[j\frac{w}{a_0} - p_i]} = \frac{945}{(\frac{1}{a_0})^5 \bullet \prod_{i=1}^{5}[jw - a_0 \cdot p_i]}$$

$$H(jw) = \frac{945a^5}{\prod_{i=1}^{5}[jw - a_0 p_i]} = \frac{945a^5}{\prod_{i=1}^{5}[jw - R_i]} \qquad \text{A.18}$$

where $R_i = a_0 \cdot p_i \Rightarrow \quad [a_0]^5 = [2.58844]^5 \cdot F_c^5 \approx 116.2\, F_c^5$ A.19

From eqn.A.18 and A.19 we write :

$$H(jw) = \frac{109805\, F_c^5}{\prod_{i=1}^{5}[jw - R_i]} = \frac{g_0}{\prod_{i=1}^{5}(X - R_i)} \qquad \text{A.20}$$

where : $X = jW \Rightarrow$

$$R_i = a_0 \cdot p_i = 2.58844\, F_c \cdot p_i \quad , \quad for \ i = 1,2,3,.,.,.,5$$

$$g_0 = 109805 \cdot F_c^5 \qquad \text{A.21}$$

Table A.1 summarises all parameters of analogue bessel filter for frequency spread of $2Hz$, $1Hz$, $0.5Hz$. The transfer function of a 5^{th}-order bessel filter is given by eqn.A.20 to be implemented in digital for use in computer simulation through the method which is called the impulse-invariant transformation method [17,19]. The important feature of this transformation is that, the impulse response of the resulting digital filter is a sampled version of the impulse response of the analogue filter. In this technique the poles $\{R_i\}$ in the S-plane of eqn.A.21 are transformed to poles at $\{e^{RT}\}$ in the Z-plane [19]. where T is the sampling period, eqn.A.20 can be written as :

$$H(z) = \frac{g_0}{\prod_{i=1}^{5}(1 - e^{R_i T} Z^{-1})} = \frac{g_0}{\prod_{i=1}^{5}(1 - q_i Z^{-1})} \qquad \text{A.22}$$

where g_0 is the gain of the filter, $q_i^{'s}$ $(for\ i = 1,2,.,.,.,5)$ are the poles in Z-plane and equal to : $q_i = e^{R_i T}$ A.23

The poles in Z-plane obtained from eqn.A.23 and for example, when $F_{sp} = 2Hz$ and sampling rate $= 100Hz \Rightarrow T = 1/100 = 0.01\text{sec} = 10m\text{sec}$ then :

$q_1 = e^{R_1 T} = \exp(-11.114005/100) = 0.8948134$, and so on for $q_i^{'s}$.

Table A.2 shows the roots in Z-plane. The sampling rate is chosen 100 samples/sec for accurate representation of $q_1(t), q_2(t)$ which have Gaussian spectra.

Using :

$$e^{x+jy} = e^{x} \cdot e^{jy} = e^{x} \cdot (\cos y + j \sin y) \Rightarrow$$

$$e^{R_2 T} = \exp(-10.2156 + j5.311/100) = e^{-0.102} \cdot e^{j0.053} = 0.903(\cos(0.053) + j \sin(0.053))$$

$$q_2 = e^{R_2 T} = 0.90176 + j0.04784$$

The 5th order bessel filter is realised in digital form as shown in Fig.A-1, it consists of two (2-poles) section and a one-pole section. Where the poles in each 2-pole section are complex conjugates and the single pole section has a real pole. The transfer function of the filter in Fig.A-1 is, therefore:

$$H(z) = \frac{k}{(1-q_1 \cdot z^{-1})\{(1-q_2 \cdot z^{-1})(1-q_3 \cdot z^{-1})\}\{(1-q_4 \cdot z^{-1})(1-q_5 \cdot z^{-1})\}}$$

or

$$H(z) = \frac{k}{(1-q_1 \cdot z^{-1})\{1-(q_2+q_3) \cdot z^{-1}+q_2 q_3 z^{-2}\}\{1-(q_4+q_5) \cdot z^{-1}+q_4 q_5 z^{-2}\}} \quad \text{A.24}$$

where: $(1-q_2 \cdot z^{-1})(1-q_3 \cdot z^{-1}) = 1-q_2 \cdot z^{-1}-q_3 \cdot z^{-1}+q_2 q_3 z^{-2} = 1-(q_2+q_3) z^{-2}+q_2 q_3 z^{-2}$

Thus, by identification between eqn.A.24 and Fig.A-1 the filter coefficients $\{C_i\}$ are :

$$C_5 = -q_1$$

$$C_1 = -(q_2+q_3)$$

$$C_2 = q_2 q_3 \quad \text{A.25}$$

$$C_3 = -(q_4+q_5)$$

$$C_4 = -(q_4 q_5)$$

and the transfer function of the filter becomes:

$$H(z) = \frac{k}{(1+C_5 \cdot z^{-1})(1+C_1 z^{-1}+C_2 z^{-2})(1+C_3 z^{-1}+C_4 z^{-2})} \quad \text{A.26}$$

The numerical values of the tap gain coefficients of the filter are given in Table A-3.
The value of k, called the gain of the filter, in eqn.A.26 is chosen such that $\{q_i(t)\}$'s have a variance corresponding to $(1/2n^{'s})$, where n's represents the number of ionospheric reflection. This ensures that the mean length of the channel sampled impulse-response vector is equal to unity, theoretically the value of k can be obtained as follows :
The energy in the wave form *H(f),* using eqn.A.20 is given by :

$$E_W = \int_{-\infty}^{\infty} |H(f)|^2 \cdot df \quad \text{A.27}$$

is determined. E_W is normalised by a scalar, g_0 such that the energy in the wave form is equal to $(1/2n^{'s})$. But, a simpler method is to pass a sequence of digital data, whose first element is a 1 and the rest of the elements are zero, through the digital filter. For a sufficiently long sequence the sum of the sequares of the output of the digital filter, E_{sum}

is very close to E_w, particularly if the sampling frequency of the filter is considerably larger than the bandwidth of the filter. Then, E_{sum} is normalised by g_0 to obtain an energy equal to $(1/2n^{rs})$. In this way, the gains of the 5-poles digital filter for all the channels are given in Table A-4. The linear interpolation in Fig.A-1 is used to obtain the required sampling rate. Since $q_i(t)$ has been sampled at $(100Hz)$ as a compromise between the need to limit the degree of interpolation and the need to have the pole locations in the Z-plane at an adequate distance from the unit circle. In order to obtain the required sampling rate (2400 baud) linear interpolating between samples is employed to rise the sampling rate from 100 to 2400.

Frequency Spread Fsp (Hz)	2	1	0.5
Cut-off Frequency Fc (Hz)	1.17741	0.588705	0.2943525
Constant (Gain) g_0	248462.5	7764.5	242.64
Filter Poles in the S-Plane			
R1	-11.114005	-5.557	-2.7785
R2,R3	-10.2156±j5.31102	-5.1078±j2.6555	-2.5539±j1.32776
R4,R5	-7.0848±j10.883	-3.5434±j5.4416	-1.7712±j2.7208

TABLE A.1 : Summarise The Parameters of 5th-order Analogue Bessel-Filter.

Frequency Spread Fsp (Hz)	2	1	0.5
Sampling-Rate (s/sec)	100	100	100
Filter Poles in the Z-Plane			
q1	0.8948134	0.94595	0.972597
q2,q3	0.90161±j0.04793	0.94987±j0.02523	0.9747±j0.012943
q4,q5	0.92609±j0.1012	0.96377±j0.05249	0.9821±j0.02673

TABLE A.2 : Z-Plane Poles of 5th-order Bessel-Filter.

Frequency Spread Fsp (Hz)		2	1	0.5
Sampling-Rate (s/sec)		100	100	100
Filter Poles in Z-Plane.	C5	-0.894813073	-0.9459455973	-0.972597346
	C1	-1.8032297224	-1.8997381241	-1.9493964847
	C2	0.81520668026	0.902887966616	0.9502204171
	C3	-1.85218288326	-1.92753451578	-1.964160484
	C4	0.86788454672	0.931603213133	0.9651959455

TABLE A.3 : Tap-Gain Coefficients of 5th-order Bessel-Filter.

		Number of Sky-wave / Seed (Id)		
		1/57	2/57	3/75
	2	10452	15156	19501
Fsp(*Hz*)	1	204933	308538	415975
	0.5	4011459	6397988	9593500
		DC-Gain of Bessel Filter		

TABLE A.4 : The Gains of the Bessel-Filter.

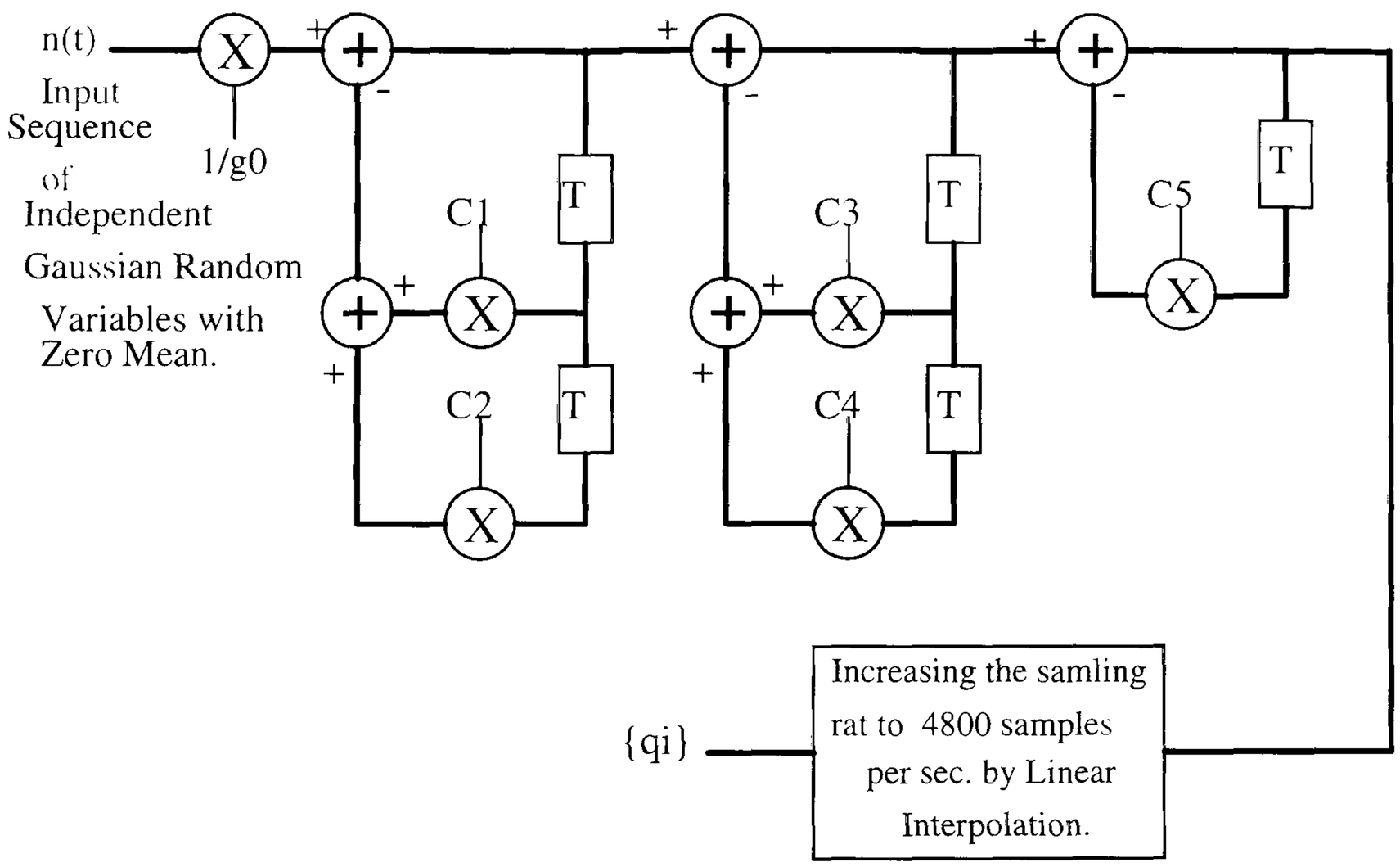


Fig. A.1 : Block Diagram of the 5th order Bessel Filter used for the Generation of Fading Samples.

APPENDIX B

THE MODEL OF QAM SYSTEM

The model of QAM system is illustrated in Fig.B-1. The input to the transmitter part $\{\alpha_i\}$ is first differentially coded into multi-level signals, $\{s_{1,i}\}$ and $\{s_{2,i}\}$, which are two streams of data symbols and statistically independent and equally likely to have any of m possible values, where

$$s_{1,i}\ ,\ s_{2,i} = 2l - m + 1 \quad ; \quad l = 0,1,.,.,.,.,.,m-1 \qquad \text{B-1}$$

The two data streams are shaped separately to the appropriate bandwidth by two lowpass transmitter filters, having the same impulse response $a(t)$, and transfer function $A(f)$, before modulating the "in-phase" and "quadrature" carriers represented by $\sqrt{2}\cos(2\pi f_c t)$ and $-\sqrt{2}\sin(2\pi f_c t)$ respectively. The absolute value of the transfer function of the low-pass transmitter filter is assumed to be band-limited from (*-1/2T to 1/2T*) *Hz*, thus ensuring that the signalling rate of *1/T Hz* is at the Nyquist rate [1,33,34] and is shown in Fig.B-2. The two outputs of the transmitter filters are next modulated by two carriers in phase quadrature with the same frequency f_c. The resulting QAM signal is then transmitted over the transmission path. The value of carrier frequency f_c is chosen such that $|D(f)|$, fits into the frequency characteristics of the available bandwidth of the channel, and the spectra of the baseband signals are shifted into the pass-band of the transmission path. The amplitude spectrum, $|D(f)|$ is shown in Fig. B-3. The white Gaussian noise in Fig.B-1 is real valued and has a two sided power spectral density of $N_0/2$ over all frequencies, and is a random process with zero mean.

At the receiver end a band pass filter (Filter C), whose real-valued impulse response $c(t)$, is used to suppress the out-of-band noise without excessively distorting the signal itself. The filter has a band-limited amplitude spectrum $|C(f)|$ such that :

$$|C(f)| = \begin{cases} 1 & f_c - \frac{1}{2T} \le |f| \le f_c + \frac{1}{2T} \\ 0 & \text{elsewhere} \end{cases} \qquad \text{B-2}$$

The amplitude spectrum of filter C, $|C(f)|$ is shown in Fig.B-4. The information to be transmitted over the transmission path in Fig.B-1-a is carried by the sequence of complex-valued data symbols $\{s_i\}$ where :

$$s_i = s_{1,i} + j\, s_{2,i} \qquad \text{B-3}$$

where $\{s_{1,i}\}$ and $\{s_{2,i}\}$ are real-valued, statistically independent and equally likely to have any of their possible values, and $j = \sqrt{-1}$.

The two-dimensional representation of a 4-point QAM signal is shown in Fig.B-5. Towards the end of this work a 4-point QAM (QPSK) is considered, hence the possible values of $s_{1,i}$ and $s_{2,i}$ are :

$$s_{1,i} = \pm 1 \quad , \quad s_{2,i} = \pm 1 \qquad \text{B-4}$$

The signals at the outputs of the two lowpass filters in the transmitter side of Fig.B-1 are given by :

$$\sum_i s_{1,i} \cdot \delta\,(t - iT) * a(t) \;\; and \;\; \sum_i s_{2,i} \cdot \delta\,(t - iT) * a(t) \qquad \text{B-5}$$

where $*$ denotes the convolution. Using the convolution method we can write eqn.B-5 as

$$\sum_i s_{1,i} \cdot a\,(t - iT) \;\; and \;\; \sum_i s_{2,i} \cdot a\,(t - iT) \qquad \text{B-6}$$

The modulated signal at the output of the adder *d(t)* is given by :

$$d(t) = \sqrt{2}\{\sum_i s_{1,i} \cdot a(t - iT)\cos 2\pi f_C t - \sum_i s_{2,i} \cdot a(t - iT)\sin 2\pi f_C t\} \qquad \text{B-7}$$

The factor $\sqrt{2}$ in eqn.B-7 gives each of $(\sqrt{2} \cdot \cos 2\pi f_c t)$ and $(\sqrt{2} \cdot \sin 2\pi f_c t)$ signals, a mean-square value (average power level) of unity, when transmitted over an infinite period. From eqn.B-7, the transmitted bandpass signal is :

$$d(t) = \sqrt{2} \cdot \sum_i s_{1,i} \cdot a(t - iT)[\frac{1}{2}(e^{j2\pi f_c t} + e^{-j2\pi f_c t})] -$$
$$-\sqrt{2} \cdot \sum_i s_{2,i} \cdot a(t - iT)[\frac{1}{2j}(e^{j2\pi f_c t} - e^{-j2\pi f_c t})] \qquad \text{B-8}$$

$$d(t) = \frac{1}{\sqrt{2}} \sum_i a(t - iT)[(s_{1,i} + j\, s_{2,i})e^{j2\pi f_c t} + (s_{1,i} - j\, s_{2,i})e^{-j2\pi f_c t}] \qquad \text{B-9}$$

or equivalently

$$d(t) = \frac{1}{\sqrt{2}} \sum_i s_i \cdot a(t - iT)e^{j2\pi f_c t} + \frac{1}{\sqrt{2}} \sum_i s_i^* \cdot a(t - iT)e^{-j2\pi f_c t} \qquad \text{B-10}$$

Then :

$$d(t) = \frac{1}{\sqrt{2}}[\sum_i (s_i \cdot e^{j2\pi f_c t} + s_i^* \cdot e^{-j2\pi f_c t}) \cdot a(t - iT)] \qquad \text{B-11}$$

$(s_i^* \cdot e^{-j2\pi f_c t})$ is the complex conjugate of $(s_i \cdot e^{j2\pi f_c t})$. From eqn.B-11 an alternative QAM transmitter system may be represented as shown in Fig.B-6.

Using the Fourier transform we can write :

$$s_i \cdot a(t-iT) \leftrightarrow s_i \cdot A(f) \cdot e^{-j2\pi f_c iT} \quad \text{and}$$
$$1/\sqrt{2}\ \ s_i \cdot a(t-iT)e^{j2\pi f_c t} \overset{F.T}{\leftrightarrow} 1/\sqrt{2}\ \ s_i \cdot A(f-f_c) \cdot e^{-j2\pi (f-f_c)iT}$$

$$\frac{1}{\sqrt{2}}\ s_i^* \cdot a(t-iT)e^{-j2\pi f_c t} \overset{F.T}{\leftrightarrow} \frac{1}{\sqrt{2}} s_i^* \cdot A(f+f_c) \cdot e^{-j2\pi (f+f_c)iT} \qquad \text{B-12}$$

It follows from this, that the Fourier transform *D(f)* of the QAM signal *d(t)* is :

$$D(f) = \frac{1}{\sqrt{2}}[\sum_i s_i \cdot A(f-f_c)e^{-j2\pi (f-f_c)iT} + \sum_i s_i^* \cdot A(f+f_c)e^{-j2\pi (f+f_c)iT}] \text{B-13}$$

It can be seen that the $\sum_i s_i \cdot a(t-iT)$ has no DC component, which means that the QAM signal d(t) is "suppressed-carrier". At the output of Bandpass filter (Filter C) the signal is coherently demodulated by nominal carrier frequency f_c but the demodulator has no knowledge of the carrier phase. The phase error ϕ in the receiver local oscillator may thus have any value between $-\pi$ *and* $+\pi$ radians. The outputs of the two demodulators are then passed via two lowpass filters "B" which suppress the high frequency components so that only the baseband signals are retained. Each of the lowpass filters (Filter B) has the impulse response *b(t)* and transfer function *B(f)*, which band limits the signals at their inputs from (*-1/2T to 1/2T Hz*). The amplitude response *|B(f)|* of filter B is shown in Fig.B-7. From Fig.B-1 the signal at the input to the coherent demodulator is given by :

$$z(t) = m(t) * c(t) \qquad \text{B-14}$$
$$m(t) = d(t) * h(t) + n(t) \qquad \text{B-15}$$

$$z(t) = [d(t) * h(t) + n(t)] * c(t) = \\ = d(t) * h(t) * c(t) + n(t) * c(t) \qquad \text{B-16}$$

The two received signals at the output of lowpass filter is given by :

$$r_1(t) = \sqrt{2}[z(t)\cos(2\pi f_c t + \phi)] * b(t) \qquad \text{B-17}$$

$$r_2(t) = -\sqrt{2}[z(t)\sin(2\pi f_c t + \phi)] * b(t) \qquad \text{B-18}$$

where ϕ is the phase error in the receiver reference carrier. Combining $r_1(t)$ and $r_2(t)$ together gives at the input of the sampler the complex form of received signal :

$$r(t) = r_1(t) + j\, r_2(j) \qquad \text{B-19}$$

From eqns.B-17, B-18 and B-19 we can write :

$$\begin{aligned} r(t) &= \sqrt{2}\{z(t)[\cos(2\pi f_c t + \phi) - j\sin(2\pi f_c t + \phi)]\} * b(t) \\ &= \sqrt{2}\{z(t)\exp(-j(2\pi f_c t + \phi))\} * b(t) \end{aligned} \qquad \text{B-20}$$

Substituting eqn.B-16 into eqn.B-20 gives :

$$\begin{aligned} r(t) = \sqrt{2}\{[d(t) * h(t) * c(t) + n(t) * c(t)]e^{-j(2\pi f_c t + \phi)}\} * b(t) = \\ \sqrt{2}\{[d(t) * h(t) * c(t)]e^{-j(2\pi f_c t + \phi)} + [n(t) * c(t)]e^{-j(2\pi f_c t + \phi)}\} * b(t) \end{aligned} \qquad \text{B-21}$$

Substituting the value of *d(t)* from eqn.B-11 into eqn.B-21 gives :

$$\begin{aligned} r(t) = \sqrt{2}\{ [(\sum_i (s_i e^{j2\pi f_c t} + s_i^* \cdot e^{-j2\pi f_c t}) a(t - iT)) * h(t) * c(t)]e^{-j(2\pi f_c t + \phi)} \\ + \sqrt{2}[n(t) * c(t)]e^{-j(2\pi f_c t + \phi)}\} * b(t) \end{aligned} \qquad \text{B-22}$$

by using the following relationship :

$$\{f_1(t) * f_2(t)\}e^{-j2\pi f_c t} = f_1(t)e^{-j2\pi f_c t} * f_2(t)e^{-j2\pi f_c t} \qquad \text{B-23}$$

which obtained by the direct application of the convolution integral. Now eqn.B-22 can be written as :

$$\begin{aligned} r(t) = \{ [(\sum_i (s_i e^{j2\pi f_c t} + s_i^* \cdot e^{-j2\pi f_c t}) a(t - iT))\ e^{-j2\pi f_c t} * (h(t) * c(t)) e^{-j2\pi f_c t}]e^{-j\phi} \\ + \sqrt{2}[n(t) * c(t)]e^{-j(2\pi f_c t + \phi)}\} * b(t) \end{aligned}$$

Or,

$$\begin{aligned} r(t) = \{(\sum_i (s_i + s_i^* \cdot e^{-j4\pi f_c t}) a(t - iT) * [h(t) * c(t)]e^{-j2\pi f_c t}\} e^{-j\phi} * b(t) \\ + \{\sqrt{2}[n(t) * c(t)]e^{-j(2\pi f_c t + \phi)}\} * b(t) \end{aligned}$$

Then,

$$\begin{aligned} r(t) = \{\sum_i s_i \cdot a(t - iT) * [h(t) * c(t)]e^{-j2\pi f_c t}\} e^{-j\phi} * b(t) \\ + \{\sum_i s_i^* \cdot e^{-j4\pi f_c t} \cdot a(t - iT) * [h(t) * c(t)]e^{-j2\pi f_c t}\} e^{-j\phi} * b(t) \\ + \{\sqrt{2}[n(t) * c(t)]e^{-j(2\pi f_c t + \phi)}\} * b(t) \end{aligned} \qquad \text{B-24}$$

Because the spectrum of $(e^{-j4\pi f_c t}\cdot a(t))$ lies outside the spectrum of the two lowpass filters (Filter B), then the second term in eqn.B-24 can be set to zero. Therefore eqn.B-24 can be reduced to :

$$r(t)=\{\sum_i s_i\cdot a(t-iT)*[h(t)*c(t)]e^{-j2\pi f_c t}\}e^{-j\phi}*b(t) \qquad \text{B-25}$$
$$+\{\sqrt{2}[n(t)*c(t)]e^{-j(2\pi f_c t+\phi)}\}*b(t)$$

Now let :

$$p(t)=\{a(t)*[h(t)*c(t)]e^{-j2\pi f_c t}\}e^{-j\phi}*b(t) \qquad \text{B-26}$$

$$v(t)=\{\sqrt{2}[n(t)*c(t)]e^{-j(2\pi f_c t+\phi)}\}*b(t) \qquad \text{B-27}$$

Where $p(t)$ is the overall impulse response and $v(t)$ is the resultant complex valued noise component in the received signal $r(t)$. Eqn.B-25 can be written as :

$$r(t)=\sum_i s_i p(t-iT)+v(t) \qquad \text{B-28}$$

From eqn.B-26 $p(t)$ can not be considered as a baseband wave-form due to the component $\exp(-j2\pi f_c t)$, but the impulse response $(h(t)*c(t))$ has a bandpass transfer function whose spectrum is centred at f_c Hz. Now let :

$$g(t)=h(t)*c(t) \qquad \text{B-29}$$

Applying Fourier transform we can write :

$$G(f)=H(f)\cdot C(f) \qquad \text{B-30}$$

$g(t)$ may be expressed in term of a lowpass complex-valued impulse response $q(t)$ and a carrier as [1,20,33] :

$$g(t)=2\Re[q(t)e^{j2\pi f_c t}] \qquad B-31-a$$
$$=q(t)e^{j2\pi f_c t}+q^*(t)e^{-j2\pi f_c t} \qquad \text{B-31-b}$$

Where $q^*(t)$ is the complex conjugate of $q(t)$. Using the inverse Fourier transform eqn.B-31-b can be written as :

$$G(f)=Q(f-f_c)+Q^*(-f-f_c) \qquad \text{B-32}$$

From eqn.B-32 the spectrum of $g(t)$ is shifted in frequency by $(-f_c\ Hz)$. Thus making $[g(t)\cdot\exp(-j2\pi f_c t)]$ a lowpass response. Substituting $g(t)=h(t)*c(t)$ in eqn.B-26 by its value in eqn.B-31-b results :

$$p(t) = \{\mathrm{a(t)} * [q(t)e^{j2\pi f_c t} + q^*(t)e^{-j2\pi f_c t}]e^{-j2\pi f_c t}\}e^{-j\phi} * b(t)$$
$$= \{(\mathrm{a(t)} * q(t)) + \mathrm{a(t)} * q^*(t)e^{-j4\pi f_c t}\}e^{-j\phi} * b(t) \qquad \text{B-33}$$
$$= [(\mathrm{a(t)} * q(t))e^{-j\phi}] * b(t) + [(\mathrm{a(t)} * q^*(t)e^{-j4\pi f_c t})e^{-j\phi}] * b(t)$$

The spectrum of $q^*(t)\exp(-j4\pi f_c t)$ is given by $Q^*(-f-2f_c)$ which is assumed to be zero over the passband of the filter A (see Fig.B-2), i.e. when :

$$Q^*(-f-2f_c) = 0 \quad for \quad -\frac{1}{2T} < f < \frac{1}{2T} \qquad \text{B-34}$$

then the term : $a(t) * q^*(t)\exp(-j4\pi f_c t)$ can be set to zero reducing the eqn.B-33 to :

$$p(t) = [(\mathrm{a(t)} * q(t))e^{-j\phi}] * b(t) \qquad \text{B-35}$$

Therefore *p(t)* is converted to a baseband with consideration that *A(f)* and *B(f)* are band limited. Furthermore, all the quantities (*a(t)*, *b(t)*, *h(t)*, *c(t)*, f_c and ϕ) in eqn.B-26 are either known or can easily be determined from the corresponding frequency characteristics. The representation of eqn.B-26 does not need any band limiting conditions to be imposed on *g(t)*. Therefore it is more convenient to use the channel model in eqn.B-26, particularly, when the system is to be computer simulated. Thus, the channel model in eqn.B-26 can be considered to represent the baseband system which is equivalent to the bandpass QAM system. An alternative representation bandpass QAM system is shown in Fig.B-8. In deriving eqn.B-28, it is assumed that the receiver is synchronised with transmitter, but in practice the receiver needs a carrier recovery system to track the signal carrier variation. Such a system employs the well known phase locked loop technique.[21,22].

The additive noise is now represented by *v(t)* which is a band-limited complex-valued Gaussian noise wave form. Using eqn.B-23 and B-27 the noise function can be written as

$$v(t) = \{\sqrt{2}[(n(t)e^{-j2\pi f_c t}) * (c(t)\cdot e^{-j2\pi f_c t})]e^{-j\phi}\} * b(t) \qquad \text{B-36}$$

The Fourier transform of $(c(t)\exp(-j2\pi f_c t))$ is $C(f+f_c)$ which is symmetrical about carrier frequency $f_c Hz$. Any sample of the real part of *v(t)* and any sample of the imaginary part of *v(t)* are statistically independent Gaussian random variables with zero mean and fixed variance [23]. The factor $\exp(-j2\pi f_c t)$ in $(n(t)\exp(-j2\pi f_c t))$ shifts the spectrum of n(t) by $(-f_c)$ without affecting the power spectral density of *n(t)*, hence the power spectral density of $(n(t)\exp(-j2\pi f_c t))$ will be the same as that of *n(t)* which

is $N_0/2$. Now from eqn.B-36 by using the Fourier transform the power spectral density of $v(t)$ is given by :

$$|v(t)|^2 = 2 \cdot \frac{N_0}{2} \cdot |C(f+f_c)|^2 \cdot |B(f)|^2 \qquad \text{B-37}$$

The auto correlation function of $v(t)$ is $R_v(\tau)$, which can be calculated by applying the inverse discrete Fourier transform of eqn.B-37 :

$$R_v(\tau) = N_0 \cdot \int_{-\infty}^{\infty} |C(f+f_c)|^2 \cdot |B(f)|^2 \cdot e^{j2\pi f_c \tau} \cdot df \qquad \text{B-38}$$

For $B(f)$ as given in this model (see Fig.B-7), eqn.B-38 becomes :

$$R_v(\tau) = N_0 \cdot \int_{-1/2T}^{1/2T} |C(f+f_c)|^2 \cdot |B(f)|^2 \cdot e^{j2\pi f_c \tau} \cdot df \qquad \text{B-39}$$

When the receiver bandpass filter (Filter C) is symmetric about f_c over the positive frequency, i.e. when $C(f+f_c)$ is symmetric about zero for $-1/2T < f < 1/2T$, eqn.B-2 implies that :

$$\begin{aligned} |C(f+f_c)| = 1 \qquad & \text{for} \quad -\frac{1}{2T} < f < \frac{1}{2T} \\ & \text{and} \quad -2f_c - \frac{1}{2T} < f < -2f_c + \frac{1}{2T} \\ 0 \qquad & \text{elsewhere} \end{aligned} \qquad \text{B-40}$$

From eqn.B-40, eqn.B-39 becomes :

$$R_v(\tau) = N_0 \cdot \int_{-1/2T}^{1/2T} e^{j2\pi f_c \tau} \cdot df = \frac{N_0}{T} \cdot \frac{\sin(\pi\tau/T)}{\pi\tau/T} \qquad \text{B-41}$$

The variance of $v(t)$ is given by the auto correlation function at zero lag [1,33], thus :

$$\sigma_v^2 = R_v(0) = N_0 \cdot \int_{-1/2T}^{1/2T} |C(f+f_c)|^2 \cdot df = \frac{N_0}{T} \qquad \text{B-42}$$

From eqn.B-41 the auto correlation function is real-valued, hence the real and imaginary parts of $v(t)$ are uncorrelated [1,23]. It could be shown that the auto correlation function of each of the real and imaginary parts of $v(t)$ is given by half the real part of $R_v(\tau)$, [1,17], whereas the cross-correlation function of those parts is an odd function of (τ) and is given

by half the imaginary part of $R_v(\tau)$. For this model the variance σ_v^2. for each of the real and imaginary parts of $v(t)$, is :

$$\sigma_v^2 = N_0/T \qquad \text{B-43}$$

The average transmitted energy per signal element is given by :

$$E_s = \varepsilon\,[\,|s_i|^2 \cdot \int_{-\infty}^{\infty} a^2(t)\cdot dt] \qquad \text{B-44}$$

Where $\varepsilon[.]$ refers to the expected value, assuming $\{s_i\}$ are statistically independent and have zero mean, eqn.B-44 becomes :

$$E_s = \overline{s_i}^2 \cdot \int_{-\infty}^{\infty} a^2(t)\cdot dt \qquad \text{B-45}$$

Where $\overline{s_i}^2$ is the mean-square value of s_i. Applying paraseval's theorem then eqn.B-45 becomes :

$$E_s = \overline{s_i}^2 \cdot \int_{-1/2T}^{1/2T} |A(f)|^2 \cdot df \qquad \text{B-46}$$

Since the data symbol s_i in eqn.B-46 may have (m) possible values, it carries $(\log_2 m)$ bits of information, then the average transmitted energy per bit (E_B) is given by :

$$E_B = \frac{\overline{s_i}^2}{\log_2 m} \cdot \int_{-1/2T}^{1/2T} |A(f)|^2 \cdot df \qquad \text{B-47}$$

Where m is the level of the QAM signal. For the QPSK system $(m= 4)$, then :

$$E_B = \frac{\overline{s_i}^2}{2} \cdot \int_{-1/2T}^{1/2T} |A(f)|^2 \cdot df \qquad \text{B-48}$$

From eqn.B-28, the average energy per signal element in received signal $r(t)$ is :

$$E_R = [|s_i|^2 \cdot \int_{-\infty}^{\infty} |p(t)|^2 \cdot dt] = \overline{s_i}^2 \cdot \int_{-\infty}^{\infty} |p(t)|^2 \cdot dt = \overline{s_i}^2 \cdot \int_{-\infty}^{\infty} |P(f)|^2 \cdot df \qquad \text{B-49}$$

Where $P(f)$ is the transfer function of $p(t)$ in eqn.B-26, where :

$$p(t)=\{a(t)*[h(t)*c(t)]e^{-j2\pi f_c t}\}e^{-j\phi}*b(t)$$

Using the Parseval's theorem, then :

$$E_R=\overline{s_i}^2\cdot\int_{-1/2T}^{1/2T}|A(f)|^2\cdot|H(f+f_c)|^2\cdot|C(f+f_c)|^2\cdot|B(f)|^2\,df \qquad \text{B-50}$$

Assuming the filter C and B used in this model, such that $|B(f)|=1$ over the integration range and from eqn.B-40 results :

$$E_R=\overline{s_i}^2\cdot\int_{-1/2T}^{1/2T}|A(f)|^2\cdot|H(f+f_c)|^2\cdot df \qquad \text{B-51}$$

For the case considered here ($m=4$), the average received signal energy per bit is given by

$$\begin{aligned}E_R&=\frac{\overline{s_i}^2}{\log_2 m}\cdot\int_{-1/2T}^{1/2T}|A(f)|^2\cdot|H(f+f_c)|^2\cdot df=\\&=\frac{\overline{s_i}^2}{2}\cdot\int_{-1/2T}^{1/2T}|A(f)|^2\cdot|H(f+f_c)|^2\cdot df\end{aligned} \qquad \text{B-52}$$

NOTE : Consider the convolution in eqn.B-23,

$$u_1(t)=(f_1(t)e^{j2\pi f_c t})*(f_2(t)e^{j2\pi f_c t})$$

Using the convolution theorem [33] :

$$f_1(t)*f_2(t)=\int_{-\infty}^{\infty}f_1(\tau)\cdot f_2(t-\tau)\cdot d\tau$$

$$\Rightarrow u_1(t)=\int_{-\infty}^{\infty}f_1(\tau)e^{j2\pi f_c\tau}\cdot f_2(t-\tau)e^{j2\pi f_c(t-\tau)}d\tau$$

$$=\int_{-\infty}^{\infty}f_1(\tau)\cdot f_2(t-\tau)e^{j2\pi f_c t}d\tau$$

$$=e^{j2\pi f_c t}\cdot\int_{-\infty}^{\infty}f_1(\tau)\cdot f_2(t-\tau)d\tau=e^{j2\pi f_c t}\cdot[f_1(t)*f_2(t)]$$

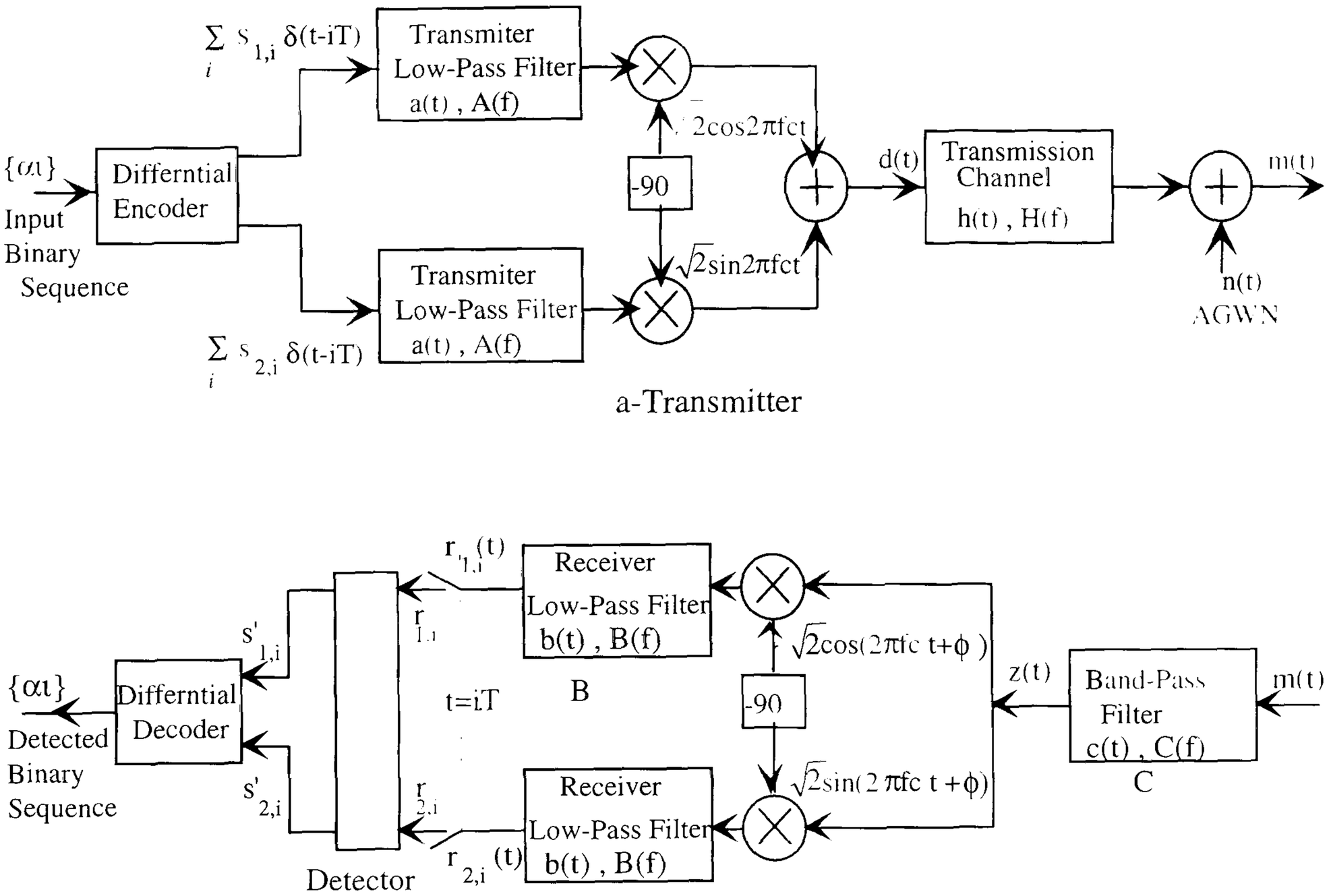


Fig.B-1: Model of QAM Data Transmission System.

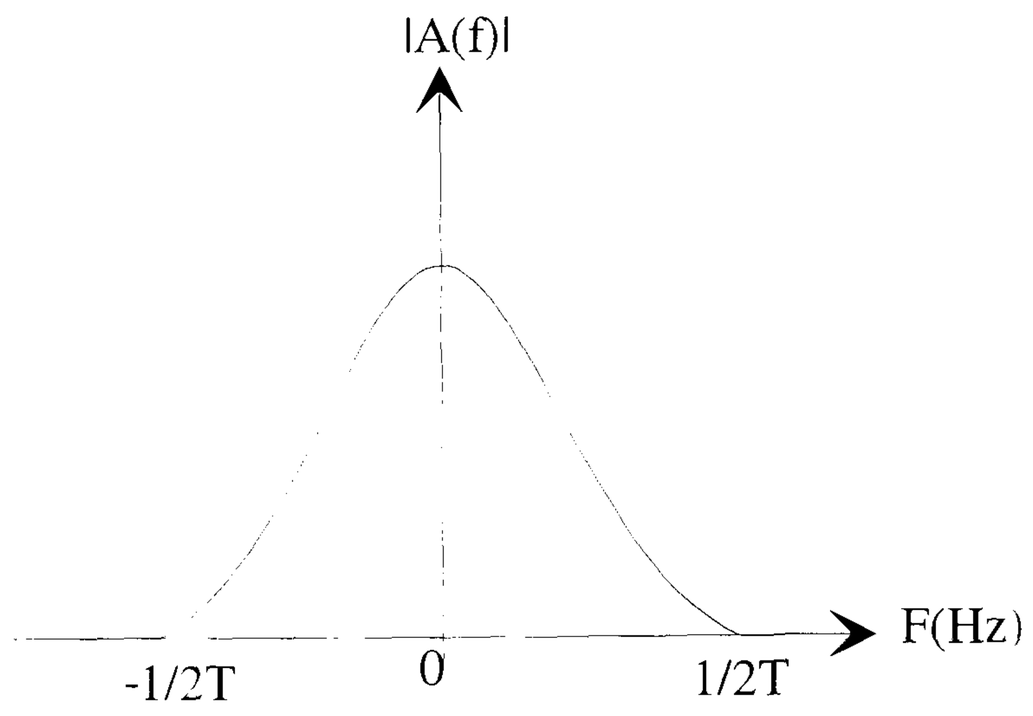


Fig. B-2 : Amplitude Spectrum of Low-Pass Filter.

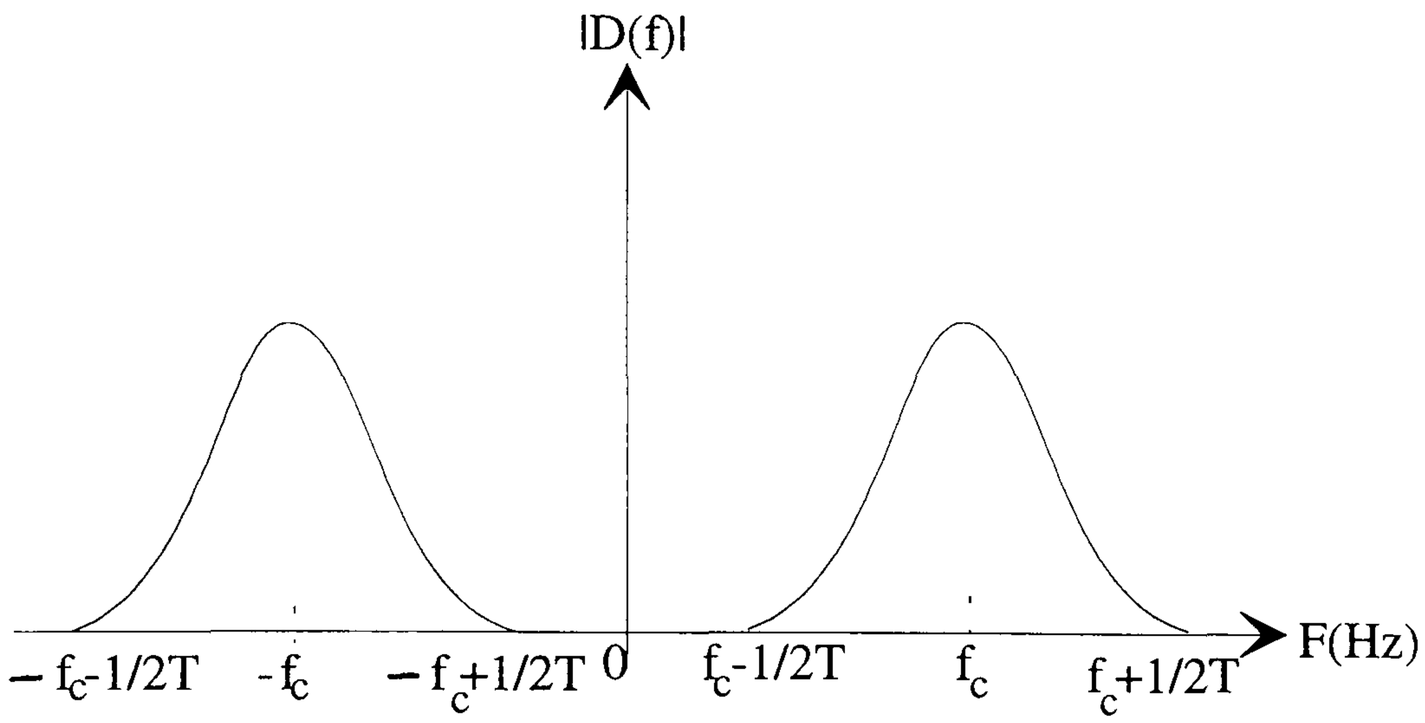


Fig. B-3 : Amplitude Spectrum of QAM Signal (*D(f)*) at the Input of Transmission Path.

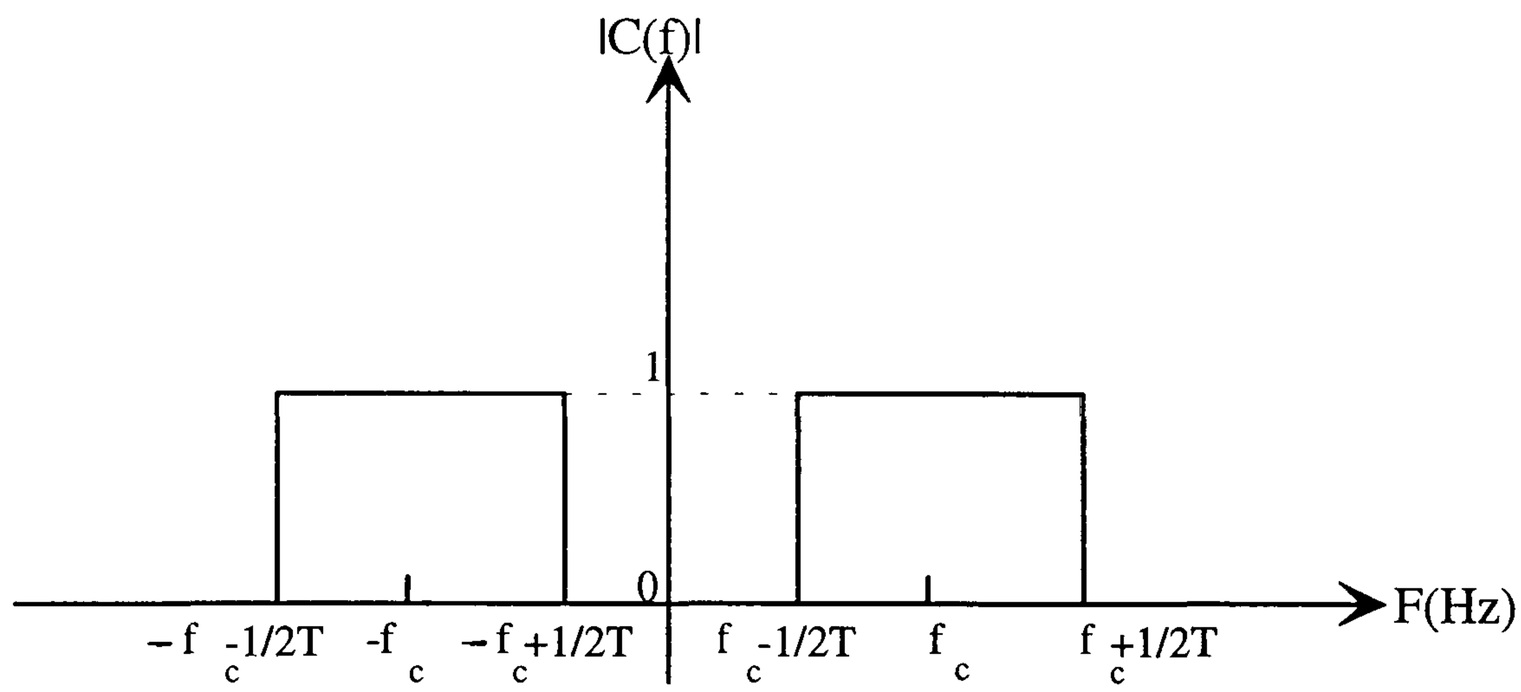


Fig. B-4 : Amplitude Spectrum of Band-Pass Filter at the Receiver End (Filter C).

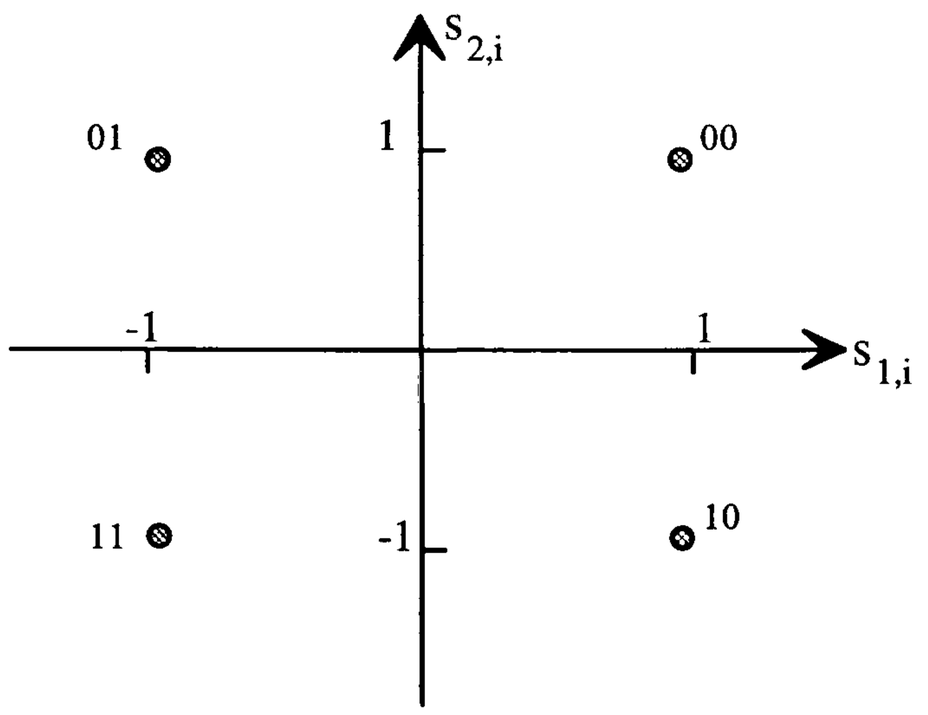


Fig. B-5 : 4-Point QAM Signal Constellation.

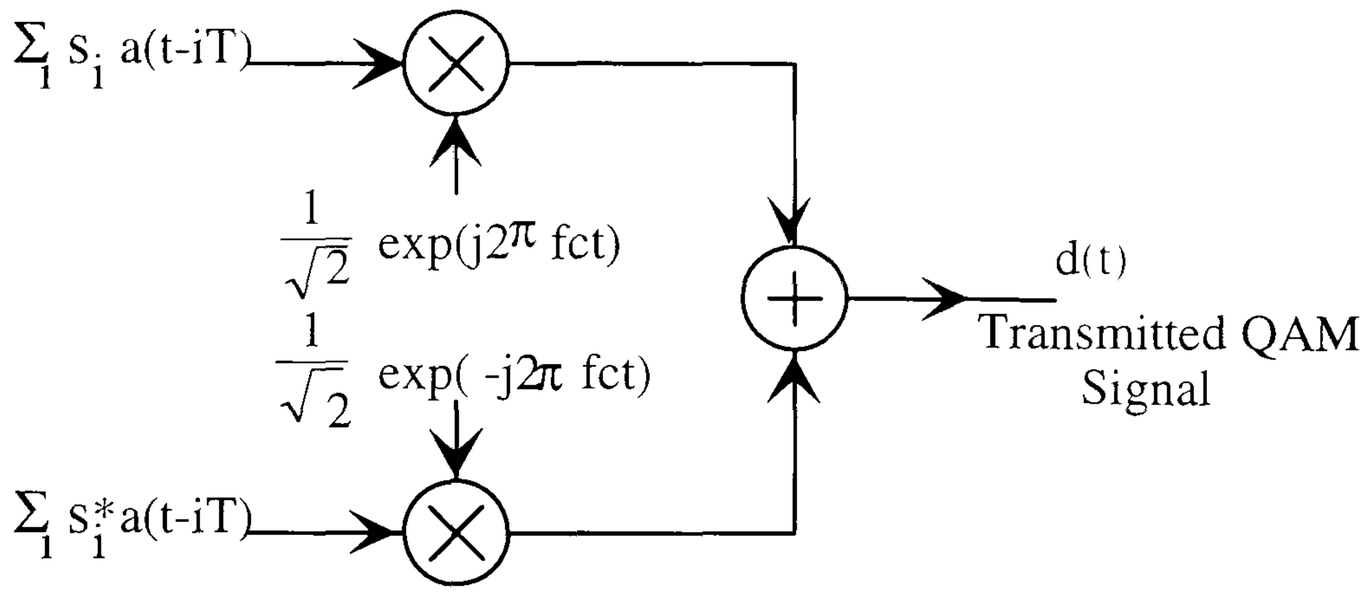


Fig. B-6 : An Alternative Representation of QAM Transmission System.

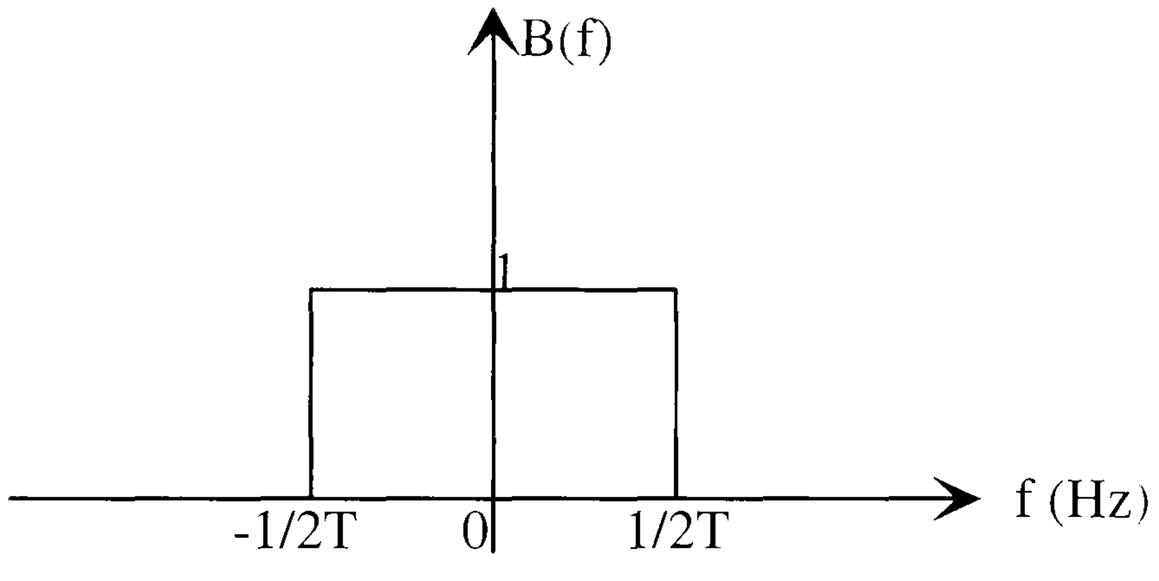


Fig. B-7 : Transfer Function of Receiver Low-Pass Filter (Filter B).

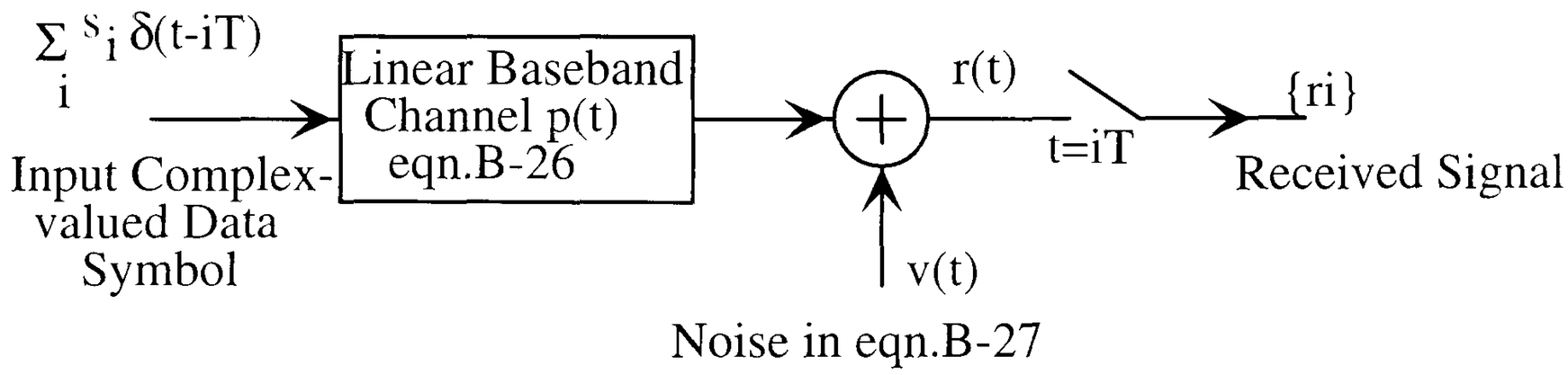


Fig. B-8 : Base-Band Model of QAM System.

APPENDIX B1

THE SIGNAL TO NOISE RATIO

The noise signal *v(t)* is given by eqn.3.2.31 as:

$$v(t) = \sqrt{2}\{[n(t) * c(t)] \cdot e^{-j2\pi f_c t}\} * b'(t) \qquad \text{B-1-1}$$

v(t) is the noise signal in the received signal (see eqn.3.2.35). Let $\{v_i\}$ which represent *v(t)* sampled at 2400 *samples/sec*, are the noise component in the received signal samples $\{r_i\}$. The auto-correlation function of *v(t)* is given by eqn.3.2.46 as :

$$R_v(t) = N_0 \cdot \int_{-f_c}^{f_c} |C(f+f_c)|^2 \cdot |B'(f)|^2 \cdot e^{j2\pi f \tau}\, df \qquad \text{B-1-2}$$

where $N_0/2$ is the power spectral density of *n(t)*, *C(f)* and *B'(f)* are the transfer function of filter C and B', respectively (Fig.3.2.1). Now from eqn.3.2.28, the transfer function of filter B (Fig.3.2.5) is given by :

$$B(f) = G(f + f_c) \cdot C(f + f_c) \cdot B'(f) \qquad \text{B-1-3}$$

where *G(f)* is the transfer function of the radio filter G and $|G(f+f_c)|^2$ can be approximated to *1* over the passband of filters C and B'. Thus eqn.B-1-2 becomes :

$$R_v(t) = N_0 \cdot \int_{-f_c}^{f_c} |B(f)|^2 \cdot e^{j2\pi f \tau}\, df \qquad \text{B-1-4}$$

The variance of *v(t)* is given by $R_v(0)$, which is :

$$\sigma_v^2 = N_0 \cdot \int_{-f_c}^{f_c} |B(f)|^2 \cdot df \equiv N_0 \cdot \int_{-\infty}^{\infty} |b(t)|^2 \cdot dt \qquad \text{B-1-5}$$

where the Parseval's theorem is used. It is well known that if *Z(f)* is a band limited spectrum whose inverse F-T is the continuos wave form *z(t)*, and if *z(t)* is sampled at a rate satisfy Nyquist rate, to give the samples $\{z_h\}$ [33], then :

$$\int_{-\infty}^{\infty} |z(t)|^2 \cdot dt = T \cdot \sum_{-\infty}^{\infty} |z_h|^2 \qquad \text{B-1-6}$$

using eqn.B-1-6, then eqn.B-1-5 becomes :

$$\int_{-\infty}^{\infty}|b(t)|^2 \cdot dt = \frac{T}{2} \cdot \sum_{-\infty}^{\infty}|b_k|^2 \qquad \text{B-1-7}$$

where $(2/T)$ is the sampling rate of the $\{b_k\}$, (T= *1/2400 sec*), thus eqn.B-1-5 becomes :

$$\sigma_v^2 = \frac{N_0 \cdot T}{2} \cdot \sum_{-\infty}^{\infty}|b_k|^2 \Rightarrow \frac{N_0}{2} = \frac{\sigma_v^2}{T \cdot \sum_{-\infty}^{\infty}|b_k|^2} \qquad \text{B-1-8}$$

Let the signal-to-noise ratio be defined as :

$$\psi = 10 \cdot \log_{10} \cdot \frac{E_B}{N_0/2} \qquad \text{B-1-9}$$

where E_B is the average transmitted energy per bit at the input of the HF channel, which is given by eqn.3.2.41 as :

$$E_B = \frac{\overline{s_i}^2}{2} \cdot \int_{-\infty}^{\infty}|A(f)|^2 \cdot df = \frac{\overline{s_i}^2}{2} \cdot \int_{-\infty}^{\infty}|a(t)|^2 \cdot dt \qquad \text{B-1-10}$$

by using the Parseval's theorem in eqn.B-1-10. Where $\overline{s_i}^2$ is the expected value of data symbol s_i which carries 4 bits of information. Using the result in eqn.B-1-6, then :

$$E_B = \frac{\overline{s_i}^2}{2} \cdot \frac{T}{2} \cdot \sum_{k=-\infty}^{\infty}|a_{1,k}|^2 = \frac{\overline{s_i}^2}{4} \cdot T \sum_{k=-\infty}^{\infty}|a_{1,k}|^2 \qquad \text{B-1-11}$$

Where $\{a_{1,k}\}$ are the sampled value of the filter A, obtained at *4800 samples/second*.
From eqns.B-1-9, B-1-11 and B-1-8, then :

$$\psi = 10 \cdot \log_{10} \cdot \frac{(\overline{s_i}^2/4) \cdot T \cdot \sum_k |a_{1,k}|^2}{\frac{\sigma_v^2}{T \cdot \sum_k |b_k|^2}} = 10 \cdot \log_{10}\left(\frac{\overline{s_i}^2 \cdot T^2}{4 \cdot \sigma_v^2} \cdot \sum_k |a_{1,k}|^2 \cdot \sum_k |b_k|^2\right) \qquad \text{B-1-12}$$

Now, in order to calculate the signal/noise ratio in easier mode, it is necessary to eliminate the factor (T^2) in eqn.B-1-12. This is done by scaling the sampled impulse

response $\{a_{1,k}\}$ & $\{b_k\}$ of the filter A and B given by Tables 3.3.1 and 3.3.2 such that the sum of the squares of their sample values become equal to their sampling rate, that is

$$\sum_{k=-\infty}^{\infty} |a_{1,k}|^2 = \sum_{k=-\infty}^{\infty} |b_k|^2 = \frac{2}{T} \qquad \text{B-1-13}$$

Eqn.B-1-12 becomes :

$$\psi = 10 \cdot \log_{10} \cdot \frac{\bar{s}_i^2}{\sigma_v^2} \qquad \text{B-1-14}$$

It is necessary to derive a relationship between σ_v^2 & $(2\sigma^2)$, where the latter is the total variance of the real and imaginary parts of discrete noise samples generated from the software model (G05DDF) obtained from (NAG) library. For the particular discrete time noise model assumed, it can be shown [17] that these two variances are related by :

$$\sigma_v^2 = 2\sigma^2 \cdot \sum_{k=-\infty}^{\infty} |b_k|^2 \qquad \text{B-1-15}$$

If now $\sum |b_k|^2 = 1$, the two variances would be numerically equal. However since this is not the case the discrete time noise samples obtained via the software model G05DDF, are passed through a scaled version of the receiver filter B, whose samples are given by $\{b'_k\}$, and where the scaling is carried out such that :

$$\sum |b'_k|^2 = 1 \qquad \text{B-1-16}$$

This would introduce the correct correlation to the noise samples, while maintaining the relationship :

$$\sigma_v^2 = 2\sigma^2 \qquad \text{B-1-17}$$

Then from eqn.B-1-14 and B-1-17, then :

$$\psi = 10 \cdot \log_{10} \cdot \frac{\bar{s}_i^2}{2\sigma^2} \qquad \text{B-1-18}$$

APPENDIX B2

DIFFERENTIAL CODING-DECODING

Differential encoding is an operation carried out on a digital signal before modulation, in such a way that the information is not transmitted by phase, but by phase shifts in the carrier. This method solves the problem of phase ambiguity, at the cost of additional coding, and a slight increase in the error rate.

The operation of differential encoding is to convert the transmitted binary stream $\{\alpha_i\}$ (Fig.3.5.1) into the data symbol $\{s_i\}$. First the encoder divides the binary stream $\{\alpha_i\}$ into two groups and each group has two digits as shown in Fig.B-2-1. For simplicity $\alpha_{1,i}$ & $\alpha_{2,i}$ represent the two binary digits in the i^{th} group as shown in Table B-2-1. The corresponding two digits $B_{1,i}$; $B_{2,i}$ are next formed by the encoder using $\alpha_{1,i}$; $\alpha_{2,i}$ according to Table B-2-1 and with the aid of the delayed two digits ($B_{1,i-1}$; $B_{2,i-1}$). $B_{1,i}$ *and* $B_{2,i}$ are converted to the complex-valued signal s_i, according to Table B-2-3 and Fig. B-2-3, where s_i is given by:

$$s_i = s_{1,i} + j\, s_{2,i} \qquad \text{B-2-1}$$

The two digits $B_{1,i}$ & $B_{2,i}$ can be given by the logical expression as follows [47]:

$$\begin{aligned} B_{1,i} &= \alpha_{1,i} \cdot \overline{B}_{1,i-1} \cdot \overline{B}_{2,i-1} + \overline{\alpha}_{2,i} \cdot B_{1,i-1} \cdot \overline{B}_{2,i-1} + \\ &\quad + \overline{\alpha}_{1,i} \cdot B_{1,i-1} \cdot B_{2,i-1} + \alpha_{2,i} \cdot \overline{B}_{1,i-1} \cdot B_{2,i-1} \end{aligned}$$

$$\begin{aligned} B_{2,i} &= \alpha_{2,i} \cdot \overline{B}_{1,i-1} \cdot \overline{B}_{2,i-1} + \alpha_{1,i} \cdot B_{1,i-1} \cdot \overline{B}_{2,i-1} + \\ &\quad + \overline{\alpha}_{2,i} \cdot B_{1,i-1} \cdot B_{2,i-1} + \overline{\alpha}_{1,i} \cdot \overline{B}_{1,i-1} \cdot B_{2,i-1} \end{aligned} \qquad \text{B-2-2}$$

where ($\overline{x}$) is the invert of (x). At the receiver side the decoding process is carried out, at the output of the detector (Fig.3.5.2), in order to obtain the detected values of $\{\alpha_i\}$. The decoding process is shown in Fig.B-2-2, where the detected binary digits $B'_{1,i}$ & $B'_{2,i}$ are first determined from the detected data symbol values $\{s'_i\}$ according to Table B-2-3 and Fig.B-2-3 . Now the encoder uses $B'_{1,i}$ & $B'_{2,i}$ (with the aid of $B'_{1,i-1}$ & $B'_{2,i-1}$), to form the detected values $\alpha'_{1,i}$ & $\alpha'_{2,i}$ according to Table B-2-2 and the decoding equation [47] :

$$\begin{aligned} \alpha'_{1,i} &= B'_{1,i} \cdot \overline{B}'_{1,i-1} \cdot \overline{B}'_{2,i-1} + B'_{2,i} \cdot B'_{1,i-1} \cdot \overline{B}'_{2,i-1} + \\ &\quad + \overline{B}'_{1,i} \cdot B'_{1,i-1} \cdot B'_{2,i-1} + \overline{B}'_{2,i} \cdot \overline{B}'_{1,i-1} \cdot B'_{2,i-1} \end{aligned}$$

$$\alpha'_{2,i} = B'_{2,i} \cdot \overline{B}'_{1,i-1} \cdot \overline{B}'_{2,i-1} + \overline{B}'_{1,i} \cdot B'_{1,i-1} \cdot \overline{B}'_{2,i-1} + \\ + \overline{B}'_{2,i} \cdot B'_{1,i-1} \cdot B'_{2,i-1} + \overline{B}'_{1,i} \cdot \overline{B}'_{1,i-1} \cdot B'_{2,i-1} \qquad \text{B-2-3}$$

It can be seen from Table B-2-1 and Fig.B-2-3, that the signal corresponding to the group of digits $\alpha_{1,i} \& \alpha_{2,i}$ represents the difference in phase between the two signals corresponding to the two groups of digits $B_{1,i}$; $B_{2,i}$ and $B_{1,i-1}$; $B_{2,i-1}$. As such, the phase of the signal corresponding to $B_{1,i}$; $B_{2,i}$ is equal to the sum of the two phases of the signals corresponding to $\alpha_{1,i}$; $\alpha_{2,i}$ and $B_{1,i-1}$; $B_{2,i-1}$ as:

$$\Delta\theta_i = \theta_i - \theta_{i-1} \qquad \text{B-2-4}$$

Thus, the group of digits $B_{1,i}$; $B_{2,i}$ could readily be obtained from Fig.B-2-3 by knowing the two groups of digits $\alpha_{1,i}$; $\alpha_{2,i}$ and $B_{1,i-1}$; $B_{2,i-1}$. Moreover, phase-shifts of multiples of (90°) in the received signal could no longer lead to prolonged error bursts in the detected binary digits $\{\alpha_i\}$, since the transmitted binary digits now represent phase differences. Gray-coding has been used in addition to differential coding (Fig.B-2-3) such that adjacent values of s_i differ only in one digit. This of course, reduces further, the probability of error in the detected data signals.

Natural code	Gray-code
00	10
10	11
01	01
11	00

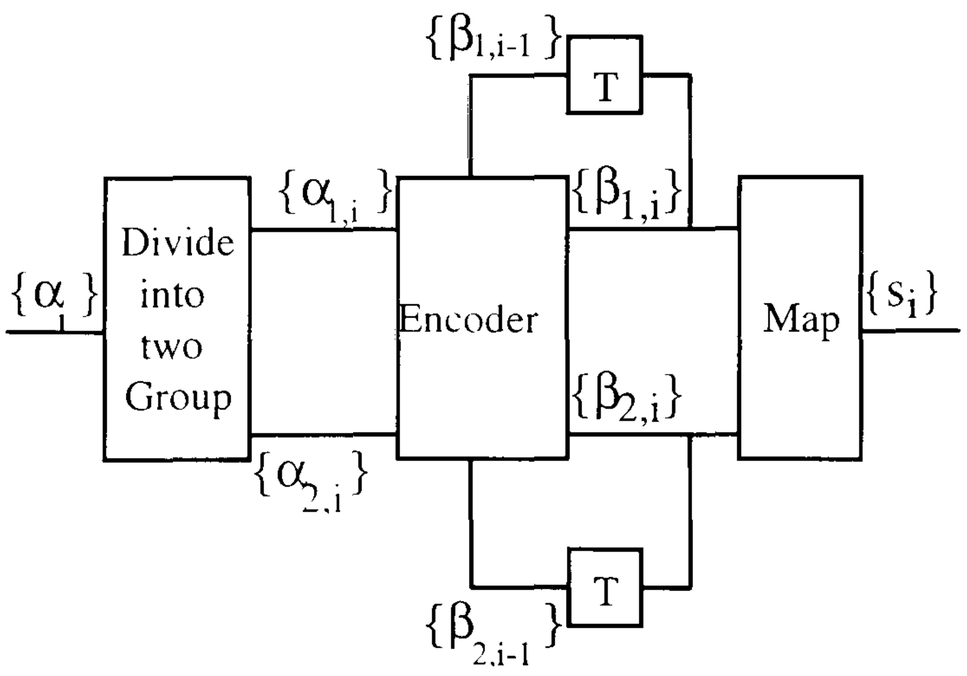


Fig.B-2-1 : QPSK Differential Coding Process.

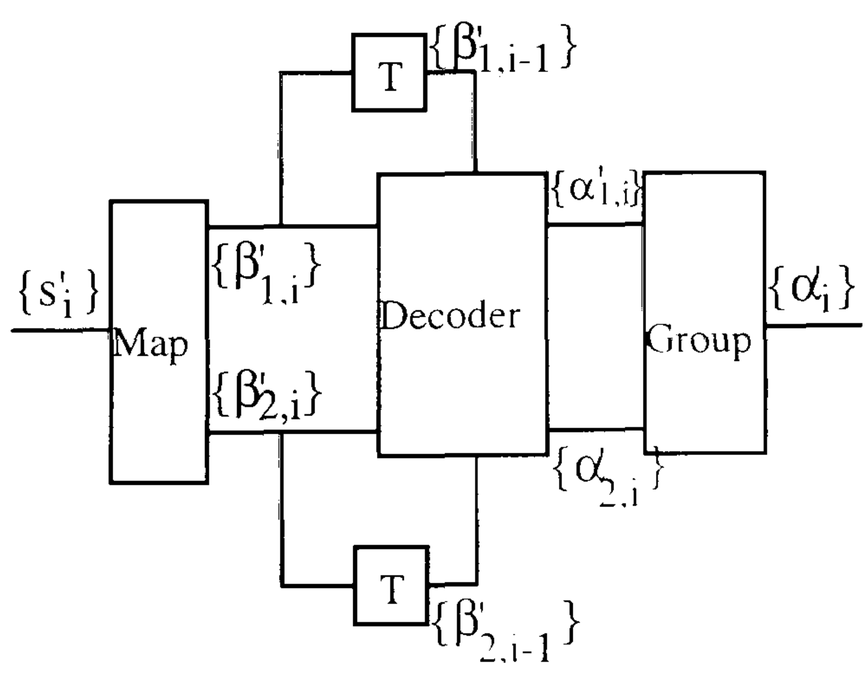
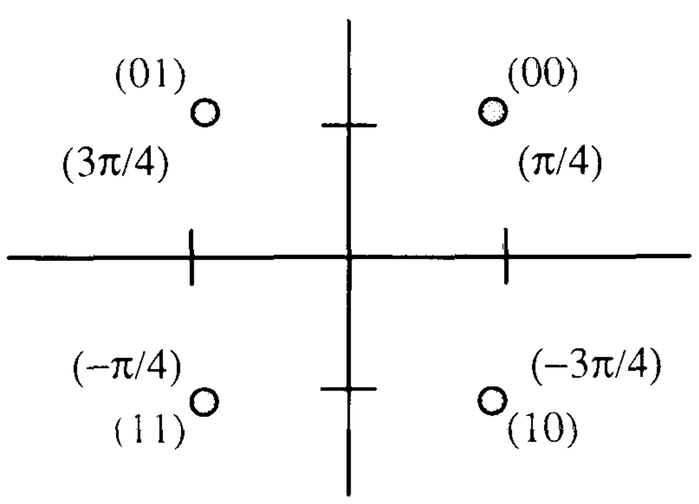


Fig.B-2-2: QPSK Differential Decoding process.

β_{i-1}		α_i		β_i		β'_{i-1}		β'_i		α'_i	
$\beta_{1,i-1}$	$\beta_{2,i-1}$	$\alpha_{1,i}$	$\alpha_{2,i}$	$\beta_{1,i}$	$\beta_{2,i}$	$\beta'_{1,i-1}$	$\beta'_{2,i-1}$	$\beta'_{1,i}$	$\beta'_{2,i}$	$\alpha'_{1,i}$	$\alpha'_{2,i}$
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	1	0	0	0	1	0	1
0	0	1	0	1	0	0	0	1	0	1	0
0	0	1	1	1	1	0	0	1	1	1	1
0	1	0	0	0	1	0	1	0	0	1	0
0	1	0	1	1	1	0	1	0	1	0	0
0	1	1	0	0	0	0	1	1	0	1	1
0	1	1	1	1	0	0	1	1	1	0	1
1	0	0	0	1	0	1	0	0	0	0	1
1	0	0	1	0	0	1	0	0	1	1	1
1	0	1	0	1	1	1	0	1	0	0	0
1	0	1	1	0	1	1	0	1	1	1	0
1	1	0	0	1	1	1	1	0	0	1	1
1	1	0	1	1	0	1	1	0	1	1	0
1	1	1	0	0	1	1	1	1	0	0	1
1	1	1	1	0	0	1	1	1	1	0	0

Table B-2-1 : Differential Encoding of Binary Digits.

Table B-2-2 : Differential Decoding of Binary Sequence.



$\beta_{1,i}$	$\beta_{2,i}$	s_i
0	0	+1+j1
0	1	-1+j1
1	0	+1-j1
1	1	-1-j1

Fig.B-2-3 The Constellation Diagram of s_i According to Table B-2-3.

Table B-2-3 Encoding Form of s_i

APPENDIX C

MINIMUM PHASE ALGORITHM

This algorithm is based on the method of adaptive receiver adjustment found by Clark and Hue [39] for time invariant channel. The operation of this algorithm can be described as follows :

The algorithm first forms a filter with Z-transform given by :

$$A_j(z) = (1+\beta_j z)^{-1} \qquad \text{C-1}$$

for $j=0,1,2,.....k$, in turn, using an iterative process to adjust β_j, so that as (j) is increased β_j tends toward γ_1 (eqn.4.4.5). γ_1 is the first root of $P(z)$ to be processed by the algorithm. Since the filter with Z-transform $Aj(z)$ does not operate on the received signal in real time (as will be illustrated later), its Z-transform is not restricted to having only negative and zero powers of z [23,39]. When $j=k$, then :

$$A_k(z) = (1+\beta_k z)^{-1} \approx (1+\gamma_1 z)^{-1} \qquad \text{C-2}$$

The algorithm next forms a filter with Z-transform $E1(z)$ such that :

$$E_1(z) = (1+\beta_k z)^{-1}\cdot(1+\beta_k^* z^{-1}) \approx (1+\gamma_1 z)^{-1}\cdot(1+\gamma_1^* z^{-1}) \qquad \text{C-3}$$

This process is carried out for each γ_i, ($i=1,2,3,.....m$), to give a total of m filters, with Z-transform $\{Ei(z)\}$, which are connected in cascade as shown in Fig.C-1, and a delay of (N-m) sampling intervals is added. In explaining the operation of the algorithm, the sequence $\{p\}$, (eqn.4.4.1) is for convenience treated as occurring in real time, i.e. its first component p_0 at time $t=0$ and its last component p_L at time $t=LT$. The method of carrying this process is :

Factoring eqn.4.4.2 , yields :

$$P(z) = (1+\gamma_1 z)[a_0 z^{-1} + a_1 z^{-2} + \cdots\cdots + a_{L-1} z^{-L}] \qquad \text{C-4}$$

where $(-1/\gamma_1)$ is a root of $P(z)$ that laying outside the unit circle in Z-plane. Multiplying eq.C-1 by C-4 then :

$$P(z)\cdot A_j(z) = (1+\beta_j z)^{-1}\cdot(1+\gamma_1 z)[a_0 z^{-1} + a_1 z^{-2} + \cdots\cdots + a_{L-1} z^{-L}] =$$

$$= (1+\gamma_1 z)(1-\beta_j z+\beta_j^2 z^2 - \beta_j^3 z^3 + \cdots\cdots)[a_0 z^{-1} + a_1 z^{-2} + \cdots\cdots + a_{L-1} z^{-L}] =$$

$$= (1-\beta_j z+\beta_j^2 z^2 - \beta_j^3 z^3 + \cdots + \gamma_1 z - \gamma_1\beta_j z^2 + \gamma_1\beta_j^2 z^3 - \gamma_1\beta_j^3 z^4 + \cdots)[a_0 z^{-1} + a_1 z^{-2} + \cdots\cdots + a_{L-1} z^{-L}]$$

$$= [1 + (\gamma_1 - \beta_j)z - \beta_j(\gamma_1 - \beta_j)z^2 + \beta_j^2(\gamma_1 - \beta_j)z^3 - \cdots\cdots][a_0 z^{-1} + a_1 z^{-2} + \cdots\cdots + a_{L-1} z^{-L}] =$$

$$= [1 + \Gamma z - \beta_j \Gamma z^2 + \beta_j^2 \Gamma z^3 - \beta_j^3 \Gamma z^4 + \cdots\cdots][a_0 z^{-1} + a_1 z^{-2} + \cdots\cdots + a_{L-1} z^{-L}] \qquad \text{C-5}$$

$\Rightarrow$

$$P(z) \cdot A_j(z) = [a_0 z^{-1} + a_1 z^{-2} + \cdots\cdots + a_{L-1} z^{-L}] + (\Gamma a_0 + \Gamma a_1 z^{-1} + \cdots + \Gamma a_{L-1} z^{-L+1}) -$$

$$- \beta_j \Gamma a_0 z - \beta_j \Gamma a_1 - \beta_j \Gamma a_2 z^{-1} - \cdots - \beta_j \Gamma a_{L-1} z^{-L+2} + \beta_j^2 \Gamma a_0 z^2 + \beta_j^2 \Gamma a_1 z +$$

$$+ \beta_j^2 \Gamma a_2 + \beta_j^2 \Gamma a_3 z^{-1} + \cdots + \beta_j^2 \Gamma a_{L-1} z^{-L-3} - \beta_j^3 \Gamma a_0 z^3 - \beta_j^3 \Gamma a_1 z^2 - \beta_j^3 \Gamma a_2 z - \beta_j^3 \Gamma a_3 -$$

$$- \beta_j^3 \Gamma a_4 z^{-1} - \cdots\cdots - \beta_j^3 \Gamma a_{L-1} z^{-L+4} + \cdots\cdots\cdots$$

where : $\Gamma = (\gamma_1 - \beta_j)$

$\Rightarrow$

$$P(z) \cdot A_j(z) = \Gamma a_0 - \beta_j \Gamma a_1 + \beta_j^2 \Gamma a_2 - \beta_j^3 \Gamma a_3 + \beta_j^4 \Gamma a_4 - \cdots\cdots +$$

$$+ (a_0 + \Gamma a_1 - \beta_j \Gamma a_2 + \beta_j^2 \Gamma a_3 - \beta_j^3 \Gamma a_4 + \cdots\cdots\cdots) z^{-1} +$$

$$+ (a_1 + \Gamma a_2 - \beta_j \Gamma a_3 + \beta_j^2 \Gamma a_4 - \beta_j^3 \Gamma a_5 + \cdots\cdots\cdots) z^{-2} + \qquad \text{C-6}$$

$$+ (\cdots\cdots) z^{-3} + \cdots\cdots + (\cdots\cdots) z^{-L} + + z(-\beta_j \Gamma a_0 + \beta_j^2 \Gamma a_1 - \beta_j^3 \Gamma a_2 + \cdots) + z^2(\cdots\cdots) + \cdots\cdots$$

Then : $P(Z) \cdot A_j(z) = \cdots + e_{j,-2} + e_{j,-1} \cdot z + e_{j,0} + e_{j,1} \cdot z^{-1} + e_{j,2} \cdot z^{-2} + \cdots\cdots + e_{j,L} \cdot z^{-L}$ C-7

where now the $\{e_{j,h}\}$, for $(-\infty < h \le L)$, form the sequence at the output of the filter with Z-transform $A_j(z)$, when the sequence P is fed into it. Since this operation is not carried out in real time, but on the stored sequence P (eqn.4.4.1), there is nothing to prevent the components ($e_{j,-1}$, $e_{j,-2}$, $e_{j,-3}$, ...) being non zero [23,39]. Therefore the iterative process operates entirely on the sequence ($L+1$)-components, ($e_{j,0}$, $e_{j,1}$, $e_{j,2}$, ..., $e_{j,L}$) only. From eqns.C-6 and C-7, we can write :

$$e_{j,0} = a_0(\gamma_1 - \beta_j) - a_1\beta_j(\gamma_1 - \beta_j) + \beta_j^2 a_2(\gamma_1 - \beta_j) - \cdots\cdots + a_{L-1}(-\beta_j)^{L-1}(\gamma_1 - \beta_j) =$$

$$= (\gamma_1 - \beta_j)[a_0 - a_1\beta_j + a_2\beta_j^2 - \cdots\cdots\cdots + a_{L-1}(-\beta_j)^{L-1}] \qquad \text{C-8}$$

$$e_{j,1} = a_0 + a_1(\gamma_1 - \beta_j) - a_2\beta_j(\gamma_1 - \beta_j) + \beta_j^2 a_3(\gamma_1 - \beta_j) - \cdots\cdots + a_{L-1}(-\beta_j)^{L-2}(\gamma_1 - \beta_j) =$$

$$= a_0 + (\gamma_1 - \beta_j)[a_1 - a_2\beta_j + a_3\beta_j^2 - \cdots\cdots\cdots + a_{L-1}(-\beta_j)^{L-2}] \qquad \text{C-9}$$

$$e_{j,2} = a_1 + a_2(\gamma_1 - \beta_j) - a_3\beta_j(\gamma_1 - \beta_j) + \beta_j^2 a_4(\gamma_1 - \beta_j) - \cdots\cdots + a_{L-1}(-\beta_j)^{L-3}(\gamma_1 - \beta_j) =$$

$$= a_1 + (\gamma_1 - \beta_j)[a_2 - a_3\beta_j + a_4\beta_j^2 - \cdots\cdots\cdots + a_{L-1}(-\beta_j)^{L-3}] \qquad \text{C-10}$$

$$e_{j,3} = a_2 + (\gamma_1 - \beta_j)[a_3 - a_4\beta_j + a_5\beta_j^2 - \cdots\cdots\cdots + a_{L-1}(-\beta_j)^{L-4}] \qquad \text{C-11}$$

and so on.

When $\beta_j \to \gamma_1$, then from eqn.C-8 to eqn.C-11, yields :

$$e_{j,h} \approx a_{h-1} \; ; \text{ for } h=1, 2, 3, \ldots\ldots\ldots, L \qquad \text{C-12}$$

but from eqn.C-8, then :

$$e_{j,0} \approx (\gamma_1 - \beta_j)\,\varepsilon_j \approx 0 \qquad \text{C-13}$$

where : $$\varepsilon_j = e_{j,1} - e_{j,2}\cdot\beta_j + e_{j,3}\cdot\beta_j^2 - \cdots\cdots + e_{j,L}\cdot(-\beta_j)^{L-1} \qquad \text{C-14}$$

so that : $$\gamma_1 - \beta_j \approx e_{j,0}/\varepsilon_j \quad \Rightarrow \quad \gamma_1 \approx \beta_j + e_{j,0}/\varepsilon_j \qquad \text{C-15}$$

This suggests that a recursive formula can be used to estimate the root γ_1 as :

$$\beta_{j+1} = \beta_j + d\cdot(e_{j,0}/\varepsilon_j) \qquad \text{C-16}$$

where d is a positive real constant normally in the range $(0\text{-}1)$, and β_{j+1} is an estimate of γ_1 which may be a better estimate of γ_1 than is β_j .

Implementation of eqn.C-16 can be carried out in an iterative manner :

1. The sequence P eqn.4.4.1 is stored, with an estimate β_j of the quantity γ_1. At the start of the operation, the algorithm sets the first estimate of γ_1 equal to zero, i.e. $\beta_0=0$. Next an appropriate one tap feed-back transversal filter Fig.C-2 is adjusted. This filter is stable so long as $|\beta_j|<1$ [39], which is always arranged to be so, and its Z-transform is given by :

$$\frac{1}{1+\beta_j\cdot z^{-1}} = (1+\beta_j\cdot z^{-1})^{-1} = (1-\beta_j\cdot z^{-1}+\beta_j^2\cdot z^{-1}-\beta_j^2\cdot z^{-1}+\cdots\cdots) \qquad \text{C-17}$$

The sequence $\{p_h\}$, $(h=0,1,2,\ldots\ldots, L)$ is fed through the filter after being reversed in order so that, the first component is now p_L. The sequence P, passing via the filter in reverse order, is taking to be moving back-wards in time, starting with the component p_L at time $t=LT$ and ending with the component p_0 at time $t=0$. The delay of one sampling interval T in the filter in Fig.C-2 becomes an advance of T, with Z-transform z. Therefore the Z-transform of the feed-back filter becomes $A_j(z)$ eqn.C-1. The output of this filter is the sequence $\{e_{j,h}\}$ given by eqn.C-7. Only the $(L+1)$-components $(e_{j,0}, e_{j,1},\ldots\ldots\ldots,e_{j,L})$ of this sequence are in fact generated.

2. The algorithm now calculates (ε_j) given by eqn.C-14, and by using eqn.C-16 an improved estimate of γ_1 is obtained and β_j is replaced by β_{j+1}. The effective Z-transform of the new one-tap feed-back filter, when operating on the reverse order sequence is :

$$A_{j+1}(z) = (1+\beta_{j+1}\cdot z)^{-1} \qquad \text{C-18}$$

where the coefficient of z^{-h} in $P(z).A_{j+1}(z)$ is $e_{j+1,h}$ and the constant (d) in eqn.C-16 is usually set to unity or may be reduced to some value depending on the channel to be treated, to enable the algorithm to locate most of the roots of $P(z)$ that lie outside the unit

circle [39]. The iterative process continues in the same manner, and terminated when one of the following conditions have been reached :

$$\text{I.)} \; |\,(\, e_{j,0} / \varepsilon_j \,)\,| < b. \quad ; \quad \text{II.)} \; j = 40. \quad ; \quad \text{III.)} \; |\,\beta_j\,| > 1 \quad ; \quad \text{IV.)} \; \varepsilon_j \to 0. \qquad \text{C-19}$$

where (b) is an appropriate small positive real constant. In case (IV) an overflow occurs in ($e_{j,0} / \varepsilon_j$). When the condition (I) in eqn.C-19 is achieved the iteration process is taken to have converged and if the value of (j) at convergence is k, then :

$$\beta_k \approx \gamma_1 \qquad \text{C-20}$$

The output of the filter in Fig.C-2 has the Z-transform given by :

$$P(z) \cdot A_k(z) = e_{k,0} + e_{k,1} z^{-1} + \cdots\cdots + e_{k,L} z^{-L} \approx a_0 + a_1 z^{-1} + \cdots\cdots + a_{L-1} z^{-L} \qquad \text{C-21}$$

where $\{a_h\}$ are given by eqn. C-4, and $(e_{k,0} \approx 0)$.

Having found the negative reciprocal of the first root, the SIR of the channel P, held in store is now modified as though it were connected to a transversal filter, which removes the root that has been found. Moreover, the last output sequence obtained from the filter in Fig.C-2 is the response of the channel when the first root has been removed. The effect of adding the complex conjugate of the reciprocal of the root is obtained by using a two-tap feed-forward filter, which is shown in Fig.C-3, whose Z-transform is

$$B_k(z) = 1 + \beta_k^* \cdot z^{-1} \qquad \text{C-22}$$

The sequence $\{e_{k,h}\}$, for $h=0,1,2,\ldots\ldots,L$ is now fed via this filter after being reversed in order as shown in Fig.C-3, which is taking to occur at time $t=0$. The output of this filter is given by :

$$Y'_1(z) = y'_{1,-1} + y'_{1,0} \cdot z^{-1} + y'_{1,1} \cdot z^{-2} + \cdots\cdots\cdots y'_{1,L} \cdot z^{-L-1} \qquad \text{C-23}$$

which is approximately equal to $[P(z).A_k(z).B_k(z)]$, and where $y'_{1,-1} \approx 0$. The filters A, B can considered as one filter, whose Z-transform is :

$$E_1(z) = A_k(z) \cdot B_k(z) \approx (1 + \gamma_1 \cdot z)^{-1} \cdot (1 + \gamma_1^* \cdot z^{-1}) \qquad \text{C-24}$$

as in eqn.C-3. This filter is an all-pass network.

Finally, the output sequence of the $\{y'_{1,h}\}$ is advanced by one place (sampling interval), deleting the first component ($y'_{1,-1}$) which is approximately zero to give the final sequence Y'_1 whose Z-transform is obtained by multiplying eqn.C-23 by (z), such that

$$Y'_1(z) = y'_{1,0} \cdot + y'_{1,1} \cdot z^{-1} + y'_{1,2} \cdot z^{-2} + \cdots\cdots\cdots y'_{1,L} \cdot z^{-L} \qquad \text{C-25}$$

which is the SIR of the channel plus a filter, removing the first root and substituting the complex conjugate of its reciprocal. For practical purposes, the linear factor $(1+\gamma_1.z)$ of $P(z)$ in eqn.4.4.5 is replaced in $Y'_1(z)$ by the linear factor $(1+\gamma^*_1 . z^{-1})$. Thus the root $(1/\gamma_1)$ of $P(z)$ is replaced by the root $(-\gamma^*_1)$ which lies inside the unit circle. The sequence Y'_1 is now reversed in order and used in place of the sequence P as the starting point for allocating the next root. When $\beta_j \neq 0$, a necessary and sufficient condition for $\beta_j = \gamma_1$ is that $(e_{j,0} = 0)$ [39]. At every step in the iterative process $|\beta_j|$ is checked, whenever $|\beta_j| > 1$ or when $(j=40)$, the iterative process is taken to have diverged and so is terminated. The process is restarted with the next initial value of β_0. An appropriate five starting points are involved in this process [39].

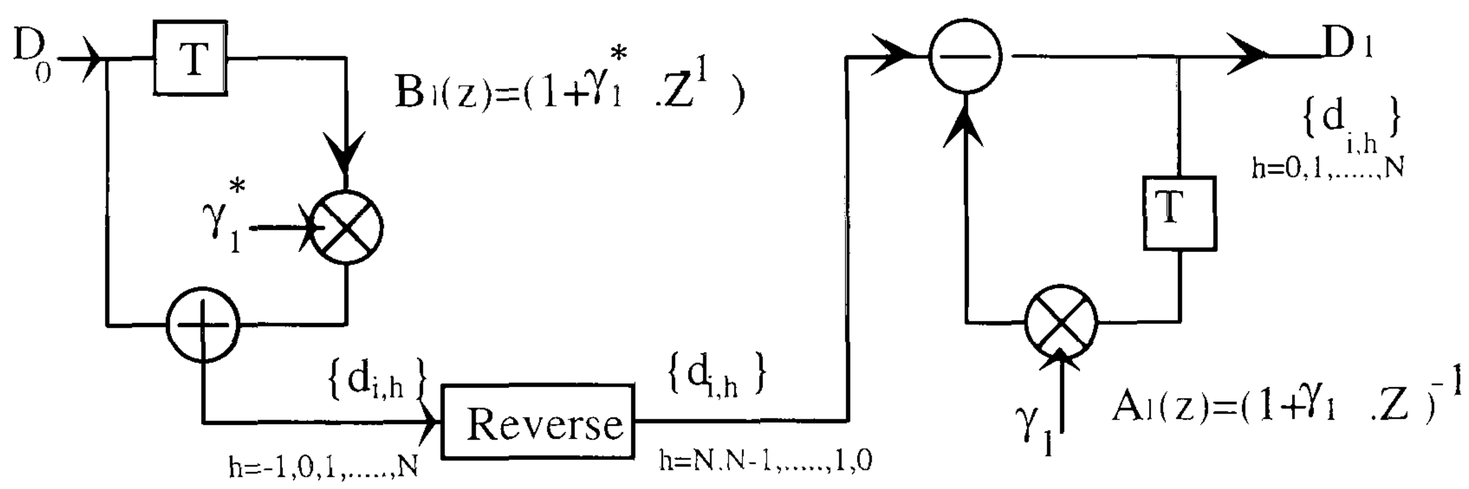


Fig.C-1 : Block Diagram of Filter E1.

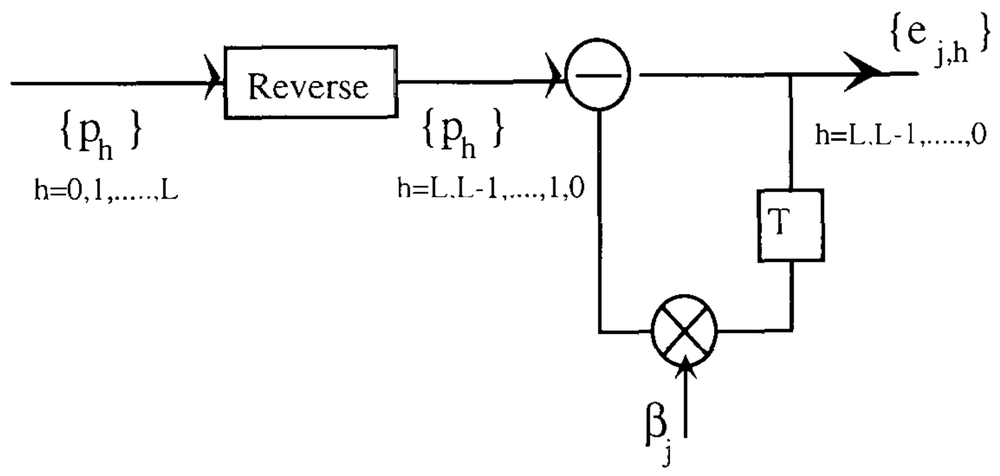


Fig.C-2 : One-tap Feedback Filter.

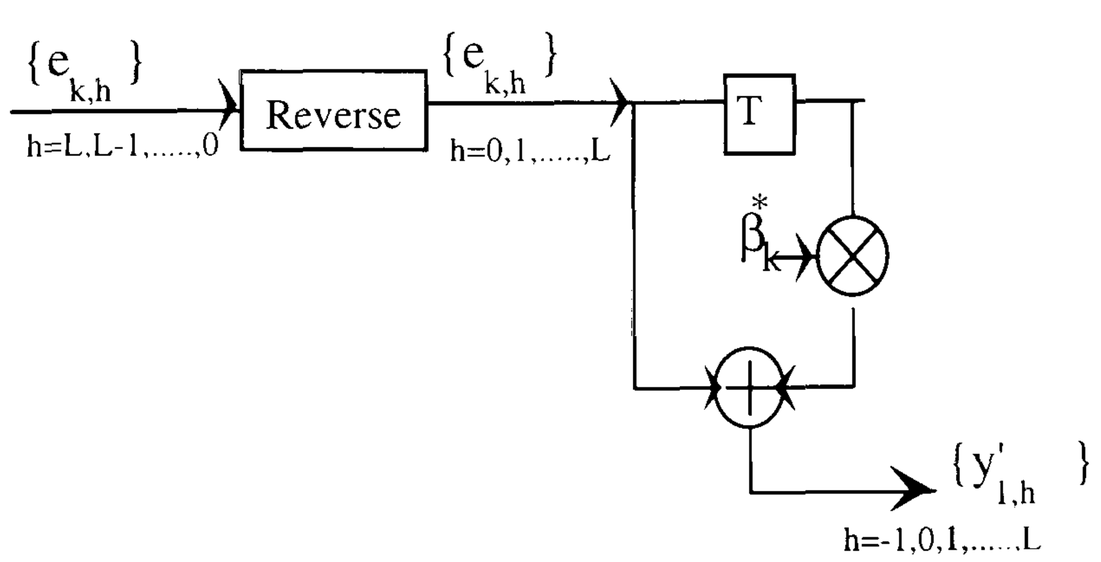


Fig.C-3 : Two-tap Feed-Forward Filter.

APPENDIX D

DERIVATION OF KALMAN ALGORITHM

The cost function of the RLS Kalman algorithm J_i is given by eqn.6.6.7 as :

$$J_i = \sum_{h=0}^{i} w^{i-h} \cdot (r_h - P'_i S_h^T)(r_h - P'_i S_h^T)^{\mathrm{H}} \qquad \text{D-1}$$

$$= \sum_{h=0}^{i} w^{i-h} (r_h \; r_h^{\mathrm{H}} - r_h \; S_h^* P_i'^{\mathrm{H}} - r_h^{\mathrm{H}} P_i' S_h^T + P_i' S_h^T S_h^* P_i'^{\mathrm{H}})$$

where r_h^{H} is the complex conjugate of r_h and so r_h^*, the vector S_h^* is the complex conjugate of the vector S_h and the vector $P_i'^{\mathrm{H}}$ is the conjugate transpose of the vector P'_i, that is $(P'_i{}^*)^T$, where : $(S^T)^{\mathrm{H}} = (S^T)^{T*} = S^*$

The cost function J_i in eqn.D-1 is real and positive and with all parameters (except P'_i) remaining constant. J_i is a convex function of the channel estimation vector P'_i with a single global minimum at a particular value of J [1,23,63]. At this point the gradient of J_i with respect to P'_i is zero, the gradient being non zero at all other points.

The gradient of J_i with respect to P'_i is given by :

$$\nabla J_i = [\frac{\partial J_i}{\partial P'_{i,0}}, \frac{\partial J_i}{\partial P'_{i,1}}, .., .., ... \frac{\partial J_i}{\partial P'_{i,L}}] \qquad \text{D-2}$$

The optimum coefficient $P'_{i,h}$ for $h=0,1,..,..,....L$ can be obtained when $\nabla J_i \equiv 0$.

First the quantity P'_i is a complex value and can be written in the form :

$$P'_i = [(p'_{i,0,1} + j\, p'_{i,0,2}), (p'_{i,1,1} + j\, p'_{i,1,2}), .., .., .., (p'_{i,L,1} + j\, p'_{i,L,2})] \qquad \text{D-3}$$

For ∇J_i to be zero, each component of the vector ∇J_i must also be zero. Differentiating eqn.D-2 with respect to P'_i, is given by [23,63]. And by the definition of differentiation with respect to a vector $\frac{\partial J_i}{\partial P'_i}$, it can be written [23,63] :

$$\nabla J_i = \frac{\partial J_i}{\partial P'_i} = \begin{bmatrix} \partial J_i / \partial p'_{i,0,1} + j\, \partial J_i / \partial p'_{i,0,2} \\ \partial J_i / \partial p'_{i,1,1} + j\, \partial J_i / \partial p'_{i,1,2} \\ . \\ . \\ \partial J_i / \partial p'_{i,L,1} + j\, \partial J_i / \partial p'_{i,L,2} \end{bmatrix}^T \quad , (P_i'^{\mathrm{H}}) = (P_i'^*)^T \qquad \text{D-4}$$

For simplicity let for example the vector p'_i be one component as :

$$P'_i = p'_{i,0,1} + j\,p'_{i,0,2} \Rightarrow P'^*_i = p'_{i,0,1} - j\,p'_{i,0,2}$$

$$\frac{\partial J_i}{\partial P'_i} = \frac{\partial J_i}{\partial p'_{i,0,1}} + j\frac{\partial J_i}{\partial p'_{i,0,2}} = 1 + j(j) = 1 - 1 = 0 \&$$

$$\frac{\partial J_i}{\partial P'^*_i} = \frac{\partial J_i}{\partial p'_{i,0,1}} + j\frac{\partial J_i}{\partial p'_{i,0,2}} = 1 + j(-j) = 1 + 1 = 2 \qquad \text{D-5}$$

From eqns.D-1 and D-4 then :

$$\nabla J_i = \frac{\partial J_i}{\partial P'_i} = \sum_{h=0}^{i} w^{i-h}\Big[\ \frac{\partial\,(r_h\ r_h^{\mathrm{H}})}{\partial p'_i} + \frac{\partial\,(-r_h\ S_h^* P'^{\mathrm{H}}_i)}{\partial p'_i} + \frac{\partial\,(-r_h^{\mathrm{H}} P'_i S_h^T)}{\partial p'_i} + \frac{\partial\,(P'_i S_h^T S_h^* P'^{\mathrm{H}}_i)}{\partial p'_i}\ \Big] \qquad \text{D-6}$$

Consider the term $J_{i,1} = -r_h \cdot S_h^* \cdot P'^{\mathrm{H}}_i$

And let: $D = -r_h \cdot S_h^* \Rightarrow J_{i,1} = D \cdot P'^*_i = \sum_{k=0}^{L} d_k (p'_{i,k,1} - j\,p'_{i,k,2})$ D-7

Where : d_k are the elements of the vector D for $k=0,1,.,.,...,L$, hence :

$$\frac{\partial J_{i,1}}{\partial p'_{i,k,1}} = d_k \quad , \quad for \quad k = 0,1,.,.,.....,L \qquad \text{D-8}$$

And

$$\frac{\partial J_{i,1}}{\partial p'_{i,k,2}} = -jd_k \quad , \quad for \quad k = 0,1,.,.,.....,L \qquad \text{D-9}$$

Substituting eqns.D-8 and D-9 into eqn. D-4 and simplifying :

$$\frac{\partial J_{i,1}}{\partial P'_i} = D_k + j(-j\,D_k) = 2\,D_k = -2\,r_h\,S_h^* \qquad \text{D-10}$$

Similarly consider the term :

$$J_{i,2} = -r_h^{\mathrm{H}} \cdot P'_i \cdot S_h^T = -r_h^* \cdot P'_i \cdot S_h^T \qquad \text{D-11}$$

$$D = -r_h \cdot S_h^* \Rightarrow D^{\mathrm{H}} = -(r_h \cdot S_h^*)^{\mathrm{H}} = r_h^{\mathrm{H}} \cdot S_h^T \Rightarrow J_{i,2} = P'_i \cdot D^{\mathrm{H}} \qquad \text{D-12}$$

In the way as above $J_{i,2}$ can be written :

$$J_{i,2} = \sum_{k=0}^{L} d_k^* (p'_{i,k,1} + j p'_{i,k,2}) \qquad \text{D-13}$$

Hence: $\dfrac{\partial J_{i,2}}{\partial p'_{i,k,1}} = d_k^* \quad , \quad for \quad k = 0,1,.,.,.... ,L$

$$\frac{\partial J_{i,2}}{\partial p'_{i,k,2}} = j d_k^* \quad , \quad for \quad k = 0,1,.,.,.... L \qquad \text{D-14}$$

Substituting D-14 in eqn.D-4 Then :

$$\frac{\partial J_{i,2}}{\partial P'_i} = d_k^* + j(j d_k^*) = 0 \qquad \text{D-15}$$

Now consider : $J_{i,3} = P'_i \cdot S_h^T \cdot S_h^* \cdot P'^*_i = P'_i \cdot \Lambda \cdot P'^*_i = J_{i,3,1} \cdot J_{i,3,2}$ D-16

Where : $\Lambda = S_h^T \cdot S_h^* \quad ; \quad J_{i,3,1} = P'_i \cdot \Lambda \ \& \ J_{i,3,2} = P'^*_i$ D-17

Then : $\dfrac{\partial J_{i,3}}{\partial P'_i} = \dfrac{\partial J_{i,3,1}}{\partial P'_i} \cdot J_{i,3,2} + J_{i,3,1} \cdot \dfrac{\partial J_{i,3,2}}{\partial P'_i}$

From the above discussion and the example given in eqn D-5 then :

$$\frac{\partial J_{i,3}}{\partial P'_i} = 0 + 2 \ J_{i,3,1} = 2 P'_i \Lambda \Rightarrow$$

$$\frac{\partial J_{i,3}}{\partial P'_i} = 2 P'_i \cdot S_h^T \cdot S_h^* \qquad \text{D-18}$$

At the end of this Appendix the proof of eqn.D-18 for two-components vector is illustrated.

and $\dfrac{\partial (r_h r_h^*)}{\partial P'_i} = 0$ D-19

From eqns.D-6, D-10, D-15, D-18, D-19 the differentiation of the cost function J_i can be written as :

$$\frac{\partial J_i}{\partial P'_i} = \sum_{h=0}^{i} w^{i-h} (-2 r_h S_h^* + 2 P'_i \cdot S_h^T \cdot S_h^*) \qquad \text{D-20}$$

Set the quantity : $\frac{\partial J_i}{\partial P'_i} = 0 \Rightarrow$

$$\sum_{h=0}^{i} w^{i-h}(2P'_i \cdot S_h^T \cdot S_h^*) = \sum_{h=0}^{i} w^{i-h}(2 r_h S_h^*) \qquad \text{D-21}$$

Now let : $Q_i = \sum_{h=0}^{i} w^{i-h}(r_h S_h^*)$ D-22

$$R_i = \sum_{h=0}^{i} w^{i-h}(S_h^T \cdot S_h^*) \qquad \text{D-23}$$

$$\Rightarrow \quad P'_i R_i = Q_i \quad \Rightarrow \quad P'_i = Q_i \cdot R_i^{-1} \qquad \text{D-24}$$

Eqn.D-23 can be written as :

$$R_i = w \cdot \sum_{h=0}^{i-1} w^{i-1-h}(S_h^T \cdot S_h^*) + S_i^T \cdot S_i^* = w R_{i-1} + S_i^T \cdot S_i^* \qquad \text{D-25}$$

Similarly eqn.D-22 can be written as :

$$Q_i = w \cdot \sum_{h=0}^{i-1} w^{i-1-h}(r_h S_h^*) + r_i S_i^* = w \cdot Q_{i-1} + r_i S_i^* \qquad \text{D-26}$$

And from eqn. D-24 it can be seen that :

$$P'_{i-1} R_{i-1} = Q_{i-1} \qquad \text{D-27}$$

Substitute eqns. D-25, D-26 and D-27 into eqn.D-24 then :

$$P'_i R_i = w \cdot P'_{i-1} \cdot R_{i-1} + r_i \cdot S_i^* \qquad \text{D-28}$$

From eqn.D-25 $\Rightarrow w R_{i-1} = R_i - S_i^T \cdot S_i^*$ D-29

Substitute eqn.D-29 into eqn.D-28 then :

$$P'_i R_i = P'_{i-1} \cdot R_i - P'_{i-1} S_i^T S_i^* + r_i \cdot S_i^* \Rightarrow$$

$$P'_i = P'_{i-1} - P'_{i-1} S_i^T S_i^* \cdot R_i^{-1} + r_i \cdot S_i^* \cdot R_i^{-1} \qquad \text{D-30}$$

$$= P'_{i-1} + (r_i - P'_{i-1} S_i^T) S_i^* \cdot R_i^{-1}$$

Let : $\Phi_i = R_i^{-1} \quad \Rightarrow$

$$P'_i = P'_{i-1} + (r_i - P'_{i-1} S_i^T) S_i^* \cdot \Phi_i \qquad \text{D-31}$$

From eqn.D-25 $\Rightarrow \Phi_i^{-1} = w \Phi_{i-1}^{-1} + S_i^T \cdot S_i^*$ D-32

Applying matrix inverse identity (Inverse-Lemma) to eqn.D-25 as :

First R_i is assumed to be non-singular, so that pre multiplying eqn.D-25 by R_i^{-1} then :

$$R_i^{-1} \cdot R_i = I = w\, R_i^{-1} \cdot R_{i-1} + R_i^{-1} \cdot S_i^T \cdot S_i^* \qquad \text{D-33}$$

Post multiplying eqn.D-33 by R_{i-1}^{-1}, then eqn.D-33 becomes :

$$R_{i-1}^{-1} = w\, R_i^{-1} + R_i^{-1} \cdot S_i^T \cdot S_i^* \cdot R_{i-1}^{-1} \qquad \text{D-34}$$

$$\Rightarrow R_{i-1}^{-1} - wR_i^{-1} = R_i^{-1} \cdot S_i^T \cdot S_i^* \cdot R_{i-1}^{-1} \qquad \text{D-35}$$

Post multiplying eqn.D-34 by $S_i^T \Rightarrow$

$$\begin{aligned} R_{i-1}^{-1} \cdot S_i^T &= w\, R_i^{-1} \cdot S_i^T + R_i^{-1} \cdot S_i^T \cdot S_i^* \cdot R_{i-1}^{-1} \cdot S_i^T = \\ &= w\, R_i^{-1} \cdot S_i^T (I + w^{-1} S_i^* \cdot R_{i-1}^{-1} \cdot S_i^T) \end{aligned} \qquad \text{D-36}$$

The matrix $(I + w^{-1} S_i^* \cdot R_{i-1}^{-1} \cdot S_i^T)$ is also assumed non-singular, then :

$$wR_i^{-1} \cdot S_i^T = R_{i-1}^{-1} \cdot S_i^T (I + w^{-1} S_i^* \cdot R_{i-1}^{-1} \cdot S_i^T)^{-1} \qquad \text{D-37}$$

Post multiplying eqn.D-37 by $(S_i^* \cdot w^{-1} \cdot R_{i-1}^{-1}) \Rightarrow$

$$\begin{aligned} wR_i^{-1} \cdot S_i^T \cdot S_i^* \cdot w^{-1} \cdot R_{i-1}^{-1} &= \\ = R_{i-1}^{-1} \cdot S_i^T (I + w^{-1} S_i^* \cdot R_{i-1}^{-1} \cdot S_i^T)^{-1} \cdot S_i^* \cdot w^{-1} \cdot R_{i-1}^{-1} \end{aligned} \qquad \text{D-38}$$

Combining eqn.D-35 and eqn.D-38, then :

$$\begin{aligned} R_{i-1}^{-1} - wR_i^{-1} &= R_{i-1}^{-1} \cdot S_i^T (I + w^{-1} S_i^* \cdot R_{i-1}^{-1} \cdot S_i^T)^{-1} \cdot S_i^* \cdot w^{-1} \cdot R_{i-1}^{-1} \Rightarrow \\ R_i^{-1} &= w^{-1} R_{i-1}^{-1} - w^{-1} R_{i-1}^{-1} \cdot S_i^T (I + w^{-1} S_i^* \cdot R_{i-1}^{-1} \cdot S_i^T)^{-1} \cdot S_i^* \cdot w^{-1} \cdot R_{i-1}^{-1} = \\ &= \frac{1}{\mathrm{w}} [R_{i-1}^{-1} - R_{i-1}^{-1} \cdot S_i^T (w + S_i^* \cdot R_{i-1}^{-1} \cdot S_i^T)^{-1} \cdot S_i^* \cdot R_{i-1}^{-1}] \Rightarrow \end{aligned}$$

$$R_i^{-1} = \frac{1}{\mathrm{w}} [R_{i-1}^{-1} - \frac{R_{i-1}^{-1} \cdot S_i^T \cdot S_i^* \cdot R_{i-1}^{-1}}{w + S_i^* \cdot R_{i-1}^{-1} \cdot S_i^T}] \qquad \text{D-39}$$

Then : $\Phi_i = \frac{1}{\mathrm{w}} [\Phi_{i-1} - \frac{\Phi_{i-1} \cdot S_i^T \cdot S_i^* \cdot \Phi_{i-1}}{w + S_i^* \cdot \Phi_{i-1} \cdot S_i^T}]$ D-40

Let : $K_i = (w + S_i^* \cdot \Phi_{i-1} \cdot S_i^T)^{-1} \cdot S_i^* \cdot \Phi_{i-1}$ D-41

Then eqn.D-40 becomes :

$$\Phi_i = \frac{1}{\mathrm{w}} \cdot \Phi_{i-1}[I - \frac{S_i^T \cdot S_i^* \cdot \Phi_{i-1}}{w + S_i^* \cdot \Phi_{i-1} \cdot S_i^T}] = w^{-1} \cdot \Phi_{i-1}[I - S_i^T \cdot K_i] \quad \text{D-42}$$

Multiply eqn.D-42 by S_i^* then :

$$S_i^* \cdot \Phi_i = w^{-1}[S_i^* \cdot \Phi_{i-1} - S_i^* \cdot \Phi_{i-1} \cdot S_i^T \cdot K_i] \quad \text{D-43}$$

From eqn.D-41 $\Rightarrow S_i^* \cdot \Phi_{i-1} = (w + S_i^* \cdot \Phi_{i-1} \cdot S_i^T) \cdot K_i$ D-44

From eqn.D-43 and D-44 then :

$$S_i^* \cdot \Phi_i = w^{-1}[(w + S_i^* \cdot \Phi_{i-1} \cdot S_i^T) \cdot K_i - S_i^* \cdot \Phi_{i-1} \cdot S_i^T \cdot K_i] = K_i \quad \text{D-45}$$

Substitute eqn.D-45 into D-31 then :

$$P'_i = P'_{i-1} + (r_i - P'_{i-1} S_i^T) \cdot K_i = P'_{i-1} + e_i \cdot K_i \quad \text{D-46}$$

Where : $e_i = (r_i - P'_{i-1} S_i^T)$ D-47

Finally the RLS Kalman algorithm is given by eqns. D-41, D-42, and D-46, D-47.

Derivation of eqn.D-16 for two-components vector

Eqn. D-16 is given by : $J_{i,3} = P'_i \cdot S_h^T \cdot S_h^* \cdot P'^*_i = P'_i \cdot \Lambda \cdot P'^*_i$ 1

Where : $\Lambda = S_h^T \cdot S_h^*$ is (L)x(L) matrix. 2

assume P'_i is a two component row vector given by :

$$P'_i = [(p'_{i,0,1} + j\, p'_{i,0,2}), (p'_{i,1,1} + j\, p'_{i,1,2})] \quad 3$$

and Λ is 2x2 matrix given by :

$$\Lambda = \begin{bmatrix} \Lambda_{11} & \Lambda_{12} \\ \Lambda_{21} & \Lambda_{22} \end{bmatrix} \quad 4$$

Then from eqn.D-4, we can write :

$$\frac{\partial J_{i,3}}{\partial P'_i} = \left[\left(\frac{\partial J_{i,3}}{\partial p'_{i,0,1}} + \frac{\partial J_{i,3}}{\partial p'_{i,0,2}} \right) \quad \left(\frac{\partial J_{i,3}}{\partial p'_{i,1,1}} + \frac{\partial J_{i,3}}{\partial p'_{i,1,2}} \right) \right] \qquad 5$$

Where:

$$J_{i,3} = P'_i \cdot \Lambda \cdot P'^{*}_i = [(p'_{i,0,1} + j\,p'_{i,0,2}), (p'_{i,1,1} + j\,p'_{i,1,2})] \cdot \begin{bmatrix} \Lambda_{11} & \Lambda_{12} \\ \Lambda_{21} & \Lambda_{22} \end{bmatrix} \cdot \begin{bmatrix} (p'_{i,0,1} + j\,p'_{i,0,2}) \\ (p'_{i,1,1} + j\,p'_{i,1,2}) \end{bmatrix}^{*} =$$

$$= \begin{matrix} [(p'_{i,0,1} + j\,p'_{i,0,2})\,\Lambda_{11} + (p'_{i,1,1} + j\,p'_{i,1,2}) \cdot \Lambda_{21}, \\ ,(p'_{i,0,1} + j\,p'_{i,0,2})\,\Lambda_{12} + (p'_{i,1,1} + j\,p'_{i,1,2}) \cdot \Lambda_{22}] \end{matrix} \cdot \begin{bmatrix} (p'_{i,0,1} - j\,p'_{i,0,2}) \\ (p'_{i,1,1} - j\,p'_{i,1,2}) \end{bmatrix} =$$

$$= [(p'^{2}_{i,0,1} + p'^{2}_{i,0,2})\,\Lambda_{11} + (p'_{i,1,1} + j\,p'_{i,1,2}) \cdot \Lambda_{21} \cdot (p'_{i,0,1} - j\,p'_{i,0,2}) +$$
$$+ (p'_{i,0,1} + j\,p'_{i,0,2})\,\Lambda_{12} \cdot (p'_{i,1,1} - j\,p'_{i,1,2}) + (p'^{2}_{i,1,1} + p'^{2}_{i,1,2})\,\Lambda_{22}]$$

Therefore:

$$\frac{\partial J_{i,3}}{\partial p'_{i,0,1}} = 2\,p'_{i,0,1} \cdot \Lambda_{11} + (p'_{i,1,1} + j\,p'_{i,1,2}) \cdot \Lambda_{21} + \Lambda_{12} \cdot (p'_{i,1,1} - j\,p'_{i,1,2})$$

$$\frac{\partial J_{i,3}}{\partial p'_{i,0,2}} = 2\,p'_{i,0,2} \cdot \Lambda_{11} - j(p'_{i,1,1} + j\,p'_{i,1,2}) \cdot \Lambda_{21} + j\,\Lambda_{12} \cdot (p'_{i,1,1} - j\,p'_{i,1,2} \qquad 6$$

$$\frac{\partial J_{i,3}}{\partial p'_{i,1,1}} = (p'_{i,0,1} - j\,p'_{i,0,2}) \cdot \Lambda_{21} + \Lambda_{12} \cdot (p'_{i,0,1} + j\,p'_{i,0,2}) + 2\,\Lambda_{22} \cdot p'_{i,1,1}$$

$$\frac{\partial J_{i,3}}{\partial p'_{i,1,2}} = j(p'_{i,0,1} - j\,p'_{i,0,2}) \cdot \Lambda_{21} - j\,\Lambda_{12} \cdot (p'_{i,0,1} - j\,p'_{i,0,2}) + 2\,\Lambda_{22} \cdot p'_{i,1,2} \qquad 7$$

From eqn.6 then:

$$\left(\frac{\partial J_{i,3}}{\partial p'_{i,0,1}} + j \frac{\partial J_{i,3}}{\partial p'_{i,0,2}} \right) = 2\,\Lambda_{11}(p'_{i,0,1} + j\,p'_{i,0,2}) + 2\,\Lambda_{21} \cdot (p'_{i,1,1} + j\,p'_{i,1,2}) \qquad 8$$

From eqn.7 then :

$$\left(\frac{\partial J_{i,3}}{\partial p'_{i,1,1}} + j \frac{\partial J_{i,3}}{\partial p'_{i,1,2}} \right) = 2\,\Lambda_{12}(p'_{i,0,1} + j\,p'_{i,0,2}) + 2\,\Lambda_{22} \cdot (p'_{i,1,1} + j\,p'_{i,1,2}) \qquad 9$$

From eqns.5, 8, 9 then:

$$\frac{\partial J_{i,3}}{\partial P'_i} = [2(p'_{i,0,1} + j\,p'_{i,0,2})\,\Lambda_{11} + 2(p'_{i,1,1} + j\,p'_{i,1,2})\,\Lambda_{21}; 2(p'_{i,0,1} + j\,p'_{i,0,2})\,\Lambda_{12} + 2(p'_{i,1,1} + j\,p'_{i,1,2})\,\Lambda_{22}]$$

$$= 2\left[(p'_{i,0,1} + j\,p'_{i,0,2}); p'_{i,1,1} + j\,p'_{i,1,2})\right] \cdot \begin{bmatrix} \Lambda_{11}\,\Lambda_{12} \\ \Lambda_{21}\,\Lambda_{22} \end{bmatrix} = 2\,P'_{i2} \cdot \Lambda_2 \qquad 10$$

From eqns.1, 4 and 10 then:

$$\frac{\partial J_{i,3}}{\partial P'_i} = 2P'_i \cdot \Lambda = 2P'_i \cdot S_h^T \cdot S_h^* \qquad 11$$

APPENDIX E

U-D FACTORS OF SQUARE ROOT ALGORITHM

The error covariance matrix Φ_i is given by eqn.6.6.19 as

$$\Phi_i = U_i^* \cdot D_i \cdot U_i^T = (1+q)[U_{i-1} \cdot \overline{U}_{i-1}]^* \cdot \overline{D}_{i-1}[U_{i-1} \cdot \overline{U}_{i-1}]^T \qquad \text{E-1}$$

Where:

$$\overline{U}_{i-1}^* \cdot \overline{D}_{i-1} \cdot \overline{U}_{i-1}^T = D_{i-1} - \alpha^{-1} \cdot B_{i-1} \cdot B_{i-1}^{\mathrm{H}} = \overline{\Phi}_i \qquad \text{E-2}$$

$$\alpha = [S_i^T \cdot \Phi_{i-1} \cdot S_i^* + \eta] \;\; ; \;\; \eta = w \qquad \text{E-3}$$

$$B_{i-1} = D_{i-1} \cdot U_{i-1}^T \cdot S_i^* = D_{i-1} \cdot F_{i-1}$$

$$B_{i-1}^{\mathrm{H}} = (D_{i-1} \cdot U_{i-1}^T \cdot S_i^*)^{*T} = S_i^T \cdot U_{i-1}^* \cdot D_{i-1} \qquad \text{E-4}$$

$$F_{i-1} = U_{i-1}^T \cdot S_i^* \quad ; \quad (1+q) = w^{-1}$$

where U is an upper triangular matrix with unit diagonal element and non-equal off-diagonal elements : $(\mu_{ij} : i = 1,2,3,\ldots..,L-1; j = i+1, i+2,\ldots\ldots..,L)$, and D is a diagonal matrix with real-valued diagonal element $(d_1 d_2 \cdots\cdots d_L)$.

From eqn.E-1 we can write :

$$U_i = U_{i-1} \cdot \overline{U}_{i-1} \quad ; \quad D_i = (1+q) \cdot \overline{D}_{i-1} \qquad \text{E-5}$$

Let the vectors S, F and B be denoted as :

$$S_i^T = [s_1 \; s_2 \cdots\cdots\cdots s_L]_{1xL}$$
$$F_i^T = [f_1 \; f_2 \cdots\cdots f_L]_{1xL} \qquad \text{E-6}$$
$$B_i^T = [b_1 \; b_2 \cdots\cdots\cdots b_L]_{1xL}$$

The U-D factors of eqn.E-2 may be found by forming the quadratic form $(S^T \cdot \overline{\Phi} \cdot S^*)$ [48,71], as :

$$S^T \cdot \overline{\Phi} \cdot S^* = S^T \cdot \overline{U}^* \cdot \overline{D} \cdot \overline{U}^T \cdot S^* = \overline{F}^{*T} \cdot \overline{D} \cdot \overline{F} = \sum_{j=1}^{L} \overline{d}_j \cdot \left| f_j \right|^2 \qquad \text{E-7}$$

Where:

$$\overline{F} = \overline{U}^T \cdot S^* \Rightarrow \overline{F}^{\mathrm{H}} = (\overline{F}^*)^T = (\overline{U}^T \cdot S^*)^{\mathrm{H}} = S^T \cdot \overline{U}^* \qquad \text{E-8}$$

And from the matrix multiplication :

$$[F]_{m*k} = [U^T]_{m*n} \cdot [S]_{n*k} \Rightarrow [F]_{L*1} = [U^T]_{L*L} \cdot [S]_{L*1}$$

The elements of the vector $\bar{F}$ is given by :

$$\bar{f}_1 = s_1^*$$

$$\bar{f}_j = \sum_{k=1}^{j-1} \mu_{kj} \cdot s_k^* + s_j^* \quad ; \; j = 2,3,.,.,.....L \qquad \text{E-9}$$

Now let : $\rho_L = -\alpha^{-1} = -[S_i^T \cdot \Phi_{i-1} \cdot S_i^* + \eta]^{-1}$, then eqn.E-7 (or E-2) becomes :

$$S^T \cdot \overline{\Phi} \cdot S^* = S^T \cdot D \cdot S^* + \rho_L \cdot S^T \cdot B \cdot B^H \cdot S^* = \sum_{j=1}^{L} d_j \cdot |s_j|^2 + \rho_L \cdot |\sum_{j=1}^{L} b_j \cdot s_j|^2$$

$$= \sum_{j=1}^{L-1} d_j \cdot |s_j|^2 + d_L \cdot |s_L|^2 + \rho_L \cdot |\sum_{j=1}^{L} b_j \cdot s_j|^2 \qquad \text{E-10}$$

$$|\sum_{j=1}^{L} b_j \cdot s_j|^2 = (\sum_{j=1}^{L} b_j \cdot s_j)(\sum_{j=1}^{L} b_j \cdot s_j)^* = [\sum_{j=1}^{L-1} b_j \cdot s_j + b_L \cdot s_L][(\sum_{j=1}^{L-1} b_j \cdot s_j)^* + (b_L \cdot s_L)^*]$$

$$= |\sum_{j=1}^{L-1} b_j \cdot s_j|^2 + |b_L|^2 \cdot |s_L|^2 + b_L s_L \cdot (\sum_{j=1}^{L-1} b_j \cdot s_j)^* + (b_L \cdot s_L)^* \cdot \sum_{j=1}^{L-1} b_j \cdot s_j \qquad \text{E-11}$$

Substituting eqn.E-11 into eqn.E-10 then :

$$S^T \cdot \overline{\Phi} \cdot S^* = \sum_{j=1}^{L-1} d_j \cdot |s_j|^2 + (d_L + \rho_L |b_L|^2) \cdot |s_L|^2 + \rho_L \cdot |\sum_{j=1}^{L-1} b_j \cdot s_j|^2 +$$

$$+ \rho_L \cdot b_L \cdot s_L (\sum_{j=1}^{L-1} b_j^* \cdot s_j^*) + \rho_L \cdot (b_L \cdot s_L)^* \cdot \sum_{j=1}^{L-1} b_j \cdot s_j \qquad \text{E-12}$$

Let :

$$\bar{d}_L = d_L + \rho_L \cdot |b_L|^2 \quad , \quad \rho_{L-1} = \frac{\rho_L \cdot d_L}{\bar{d}_L} \qquad \text{E-13}$$

$$\bar{\mu}_{jL} = \frac{\rho_L \cdot b_L}{\bar{d}_L} \cdot b_j^* \quad ; \quad for \; j = 1,2,.,.,.....,L-1$$

Eqn.E-12 can be written as:

$$S^T \cdot \overline{\Phi} \cdot S^* = \sum_{j=1}^{L-1} d_j \cdot |s_j|^2 + \bar{d}_L \cdot |s_L|^2 + \rho_L \cdot \frac{(d_L + \rho_L |b_L|^2)}{\bar{d}_L} \cdot |\sum_{j=1}^{L-1} b_j \cdot s_j|^2 +$$

$$+ \rho_L \cdot b_L \cdot s_L (\sum_{j=1}^{L-1} b_j^* \cdot s_j^*) + \rho_L \cdot b_L^* \cdot s_L^* \cdot \sum_{j=1}^{L-1} b_j \cdot s_j \quad \Rightarrow$$

$$S^T \cdot \overline{\Phi} \cdot S^* = \sum_{j=1}^{L-1} d_j \cdot |s_j|^2 + \rho_L \cdot | \sum_{j=1}^{L-1} b_j \cdot s_j |^2 + \bar{d}_L \cdot | \sum_{j=1}^{L-1} \mu_{jL} \cdot s_j^* + s_L^* |^2$$

$$= \sum_{j=1}^{L-1} d_j \cdot |s_j|^2 + \rho_L | \sum_{j=1}^{L-1} b_j \cdot s_j |^2 + \bar{d}_L \cdot | \bar{f}_L |^2 \qquad \text{E-14}$$

Where:

$$\bar{f}_L = \sum_{j}^{L-1} \overline{\mu}_{jL} \cdot s_j^* + s_L^* \ \& | \bar{f}_L |^2 = \bar{f}_L \cdot f_L^* = | \sum_{j}^{L-1} \overline{\mu}_{jL} \cdot s_j^* + s_L^* |^2 \qquad \text{E-15}$$

Note : { " Proof of eqn.E-14

$$S^T \cdot \overline{\Phi} \cdot S^* = \cap = \sum_{j=1}^{L-1} d_j \cdot |s_j|^2 + \bar{d}_L \cdot s_L \cdot s_L^* + (\rho_L \cdot \frac{d_L}{\bar{d}_L} + \frac{\rho_L \rho_L}{\bar{d}_L} \cdot |b_L|^2) \cdot | \sum_{j=1}^{L-1} b_j \cdot s_j |^2 + \mathrm{E} + \mathrm{C}$$

$$\mathrm{E} = \rho_L \cdot b_L \cdot s_L (\sum_{j=1}^{L-1} b_j^* \cdot s_j^*) \quad ; \quad C = \rho_L \cdot b^* L \cdot s^* L \cdot \sum_{j=1}^{L-1} b_j \cdot s_j \Rightarrow$$

$$\cap = \sum_{j=1}^{L-1} d_j \cdot |s_j|^2 + \bar{d}_L \cdot s_L \cdot s_L^* + \rho_{L-1} \cdot | \sum_{j=1}^{L-1} b_j \cdot s_j |^2 + \frac{\rho_L \rho_L}{\bar{d}_L} \cdot b_L \cdot b_L^* \cdot | \sum_{j=1}^{L-1} b_j \cdot s_j |^2 + E + C = I \ + I_1 + I_2 + I_3 + E + C$$

$$I_1 + I_3 + E + C = \bar{d}_L \cdot s_L \cdot s_L^* + \bar{d}_L \cdot | \sum_{j=1}^{L-1} \frac{\rho_L}{\bar{d}_L} \cdot b_L b_j \cdot s_j |^2 + \rho_L \cdot b_L \cdot s_L (\sum_{j=1}^{L-1} b_j^* \cdot s_j^*) + \rho_L \cdot b_L^* \cdot s_L^* \cdot \sum_{j=1}^{L-1} b_j \cdot s_j$$

$$= \bar{d}_L \cdot | \sum_{j=1}^{L-1} \frac{\rho_L}{\bar{d}_L} \cdot b_L \cdot b_j^* \cdot s_j^* + s_L^* |^2 = \bar{d}_L \cdot (\sum_{j=1}^{L-1} \frac{\rho_L}{\bar{d}_L} \cdot b_L \cdot b_j^* \cdot s_j^* + s_L^*) (\sum_{j=1}^{L-1} \frac{\rho_L}{\bar{d}_L} \cdot b_L \cdot b_j^* \cdot s_j^* + s_L^*)^*$$

$$\Rightarrow S^T \cdot \overline{\Phi} \cdot S^* = \sum_{j=1}^{L-1} d_j \cdot |s_j|^2 + \rho_{L-1} \cdot | \sum_{j=1}^{L-1} b_j \cdot s_j |^2 + \bar{d}_L \cdot | \sum_{j=1}^{L-1} \frac{\rho_L}{\bar{d}_L} \cdot b_L \cdot b_j^* \cdot s_j^* + s_L^* |^2$$

$$= \sum_{j=1}^{L-1} d_j \cdot |s_j|^2 + \rho_{L-1} \cdot | \sum_{j=1}^{L-1} b_j \cdot s_j |^2 + \bar{d}_L \cdot | \sum_{j=1}^{L-1} \mu_{jL} \cdot s_j^* + s_L^* |^2$$

}.

Similarly, the inductive reduction follows recursively, and after (L-1) steps the quadratic form has been reduced to [48] :

$$S^T \cdot \overline{\Phi} \cdot S^* = \sum_{j=1}^{L} \bar{d}_j \cdot |\bar{f}_j|^2 \qquad \text{E-16}$$

Provided that :

$$\bar{d}_j = d_j + \rho_j \cdot |b_j|^2 \quad , j = 1, 2, \ldots\ldots \mathrm{L} \qquad \text{E-17}$$

$$\rho_{j-1} = \frac{\rho_j \cdot d_j}{\bar{d}_j} \quad ; \ j = 1, 2, \ldots\ldots\ldots, L \qquad \text{E-18}$$

$$\overline{\mu}_{kj} = \frac{\rho_j \cdot b_j}{\bar{d}_j} \cdot b_k^* \quad ; \ for \ k = 1, 2, \ldots\ldots, j-1 \quad ; \ j = 1, 2, \ldots\ldots, L \qquad \text{E-19}$$

Multiplying eqn.E-17 by $(1/(d_j \cdot \rho_j))$, then

$$\frac{\bar{d}_j}{d_j \rho_j} = \frac{d_j}{d_j \rho_j} + \rho_j \cdot \frac{1}{d_j \rho_j} \cdot |b_j|^2 = \frac{1}{\rho_j} + \frac{|b_j|^2}{d_j} \quad , \text{ from eqn. E - 18} \Rightarrow$$

$$\frac{1}{\rho_{j-1}} = \frac{\bar{d}_j}{d_j \rho_j} \Rightarrow \frac{1}{\rho_{j-1}} = \frac{1}{\rho_j} + \frac{1}{d_j} \cdot |b_j|^2 \quad ; \text{ for } j = 1,2,\ldots\ldots, L \qquad \text{E - 20}$$

from eqn.E-4 the element of the vector B is given by :

$$b_j = d_j \cdot f_j \quad , \quad j = 1,2,\ldots\ldots, L \Rightarrow$$

$$|b_j|^2 = d_j^2 \cdot |f_j|^2 \quad , \quad j = 1,2,\ldots\ldots, L \qquad \text{E-21}$$

Substitute eqn.E-21 into E-20

$$\frac{1}{\rho_{j-1}} = \frac{1}{\rho_j} + \frac{d_j^2}{d_j} \cdot |f_j|^2 = \frac{1}{\rho_j} + d_j \cdot |f_j|^2 \quad ; \text{ for } j = 1,2,\ldots\ldots, L$$

$$= \frac{1}{\rho_L} + \sum_{k=j}^{L} d_k \cdot |f_k|^2 = -\alpha_{j-1} \quad ; \text{ for } j = 1,2,\ldots\ldots, L \qquad \text{E-22}$$

Provided that : $\alpha = -\rho_L^{-1} = \sum_{k=1}^{L} d_k \cdot |f_k|^2 + \eta \qquad \text{E-23}$

$$\alpha_j = -\rho_j^{-1} = \sum_{k=1}^{j} d_k \cdot |f_k|^2 + \eta \quad ; \text{ for } j = 1,2,\ldots\ldots, L$$

$$\alpha_0 = \eta \qquad \text{E-24}$$

Illustration of the above equations :

$$\frac{1}{\rho_j} = -\alpha_j = -[\sum_{k=1}^{j} d_k \cdot |f_k|^2 + \eta] = -[\sum_{k=1}^{L} d_k \cdot |f_k|^2 + \sum_{k=1}^{j} d_k \cdot |f_k|^2 - \sum_{k=1}^{L} d_k \cdot |f_k|^2 + \eta]$$

$$\Rightarrow \frac{1}{\rho_j} = -[\alpha_L + \sum_{k=1}^{j} d_k \cdot |f_k|^2 - \sum_{k=1}^{L} d_k \cdot |f_k|^2] = \frac{1}{\rho_L} - \sum_{k=1}^{j} d_k \cdot |f_k|^2 + \sum_{k=1}^{L} d_k \cdot |f_k|^2 \quad , \quad \text{E} - 25$$

Substitute eqn.E-25 into eqn.E-22 then :

$$\frac{1}{\rho_{j-1}} = \frac{1}{\rho_L} - \sum_{k=1}^{j} d_k \cdot |f_k|^2 + \sum_{k=1}^{L} d_k \cdot |f_k|^2 + d_j \cdot |f_j|^2 \quad ; \quad j = 1,2.....,L$$

And using the fact : $- \sum_{k=1}^{j} d_k \cdot |f_k|^2 + d_j \cdot |f_j|^2 = - \sum_{k=1}^{j-1} d_k \cdot |f_k|^2 \Rightarrow$

$$- \sum_{k=1}^{j-1} d_k \cdot |f_k|^2 + \sum_{k=1}^{L} d_k \cdot |f_k|^2 = \sum_{k=j}^{L} d_k \cdot |f_k|^2 \; ; j = 1,2,.....,L$$

then, $\frac{1}{\rho_{j-1}} = \frac{1}{\rho_L} - \sum_{k=1}^{j-1} d_k \cdot |f_k|^2 + \sum_{k=1}^{L} d_k^2 \cdot |f_k|^2 = \frac{1}{\rho_L} + \sum_{k=j}^{L} d_k \cdot |f_k|^2 = -\alpha_{j-1}$ E-26

Note : $\alpha_j = \sum_{k=1}^{j-1} d_k \cdot |f_k|^2 + \eta + d_j \cdot |f_j|^2 = \alpha_{j-1} + d_j \cdot |f_j|^2 \;\; ; \; for \; j = 1,2.....,L \Rightarrow$

& using eqn.E - 21 $\Rightarrow \alpha_j = \alpha_{j-1} + b_j \cdot f_j^* \;\; ; \; for \; j = 1,2.....,L$

for eg. $j = 1 \Rightarrow \alpha_1 = \alpha_0 + d_1 \cdot |f_1|^2 = \eta + b_1 \cdot f_1^*$ E-27

Where $\alpha_{j-1} = \sum_{k=1}^{j-1} d_k \cdot |f_k|^2 + \eta$

Multiply eqn.E-27 by (-) and simplify :

$$-\alpha_{j-1} = - \sum_{k=1}^{j-1} d_k \cdot |f_k|^2 - \eta = - \sum_{k=1}^{L} d_k \cdot |f_k|^2 + \sum_{k=j}^{L} d_k \cdot |f_k|^2 - \eta \; =$$

$$-\alpha + \sum_{k=j}^{L} d_k \cdot |f_k|^2 = \frac{1}{\rho_{j-1}}$$

$\overline{d}_j$ can be calculated as follows :{ from eqn.E-18, then :

$$\rho_{j-1} = \frac{\rho_j \cdot d_j}{\overline{d}_j} \Rightarrow \rho_j \cdot d_j = \rho_{j-1} \cdot \overline{d}_j \Rightarrow \rho_j = \rho_{j-1} \cdot \frac{\overline{d}_j}{d_j} \quad \text{but:} \quad \overline{d}_j = d_j + \rho_j \cdot |b_j|^2 = d_j + \rho_j \cdot d_j^2 \cdot |f_j|^2 =$$

$$= d_j + \rho_{j-1} \cdot \frac{\overline{d}_j}{d_j} \cdot d_j^2 \cdot |f_j|^2 = d_j - \frac{1}{\alpha_{j-1}} \overline{d}_j \cdot d_j \cdot |f_j|^2 \Rightarrow d_j = \overline{d}_j [1 + \frac{1}{\alpha_{j-1}} d_j \cdot |f_j|^2]$$

$$d_j = \overline{d}_j [\frac{\alpha_{j-1} + d_j \cdot |f_j|^2}{\alpha_{j-1}}] \Rightarrow d_j \cdot \alpha_{j-1} = \overline{d}_j [\alpha_{j-1} + d_j \cdot |f_j|^2] \;, \; but \;\; \alpha_j = \alpha_{j-1} + d_j \cdot |f_j|^2 \Rightarrow$$

$$\Rightarrow d_j \cdot \alpha_{j-1} = \overline{d}_j \cdot \alpha_j \Rightarrow \overline{d}_j = d_j \cdot \frac{\alpha_{j-1}}{\alpha_j}$$

Or $\overline{d}_j(i) = d_j(i-1) \cdot \alpha_{j-1} / \alpha_j \;\; ; \quad \text{for} \;\; j = 1,2.....,L \Rightarrow$

$$\bar{d}_1(i) = d_1(i-1)\cdot\frac{\alpha_0}{\alpha_1} = \frac{d_1(i-1)}{\alpha_1}\eta$$

$$\bar{d}_j(i) = d_j(i-1)\cdot\frac{\alpha_{j-1}}{\alpha_j} \quad ; \quad for \quad j = 1,2,\ldots..,L \qquad \text{E-28}$$

From eqn.E-19 and substitute $\rho_j = -1/\alpha_j$

$$\bar{\mu}_{kj} = -\frac{b_j}{\alpha_j\cdot\bar{d}_j}\cdot b_k^* = -\frac{b_j}{\alpha_j\cdot d_j(i-1)\cdot\alpha_{j-1}/\alpha_j}\cdot b_k^* = -\frac{d_j(i-1)\cdot f_j}{d_j(i-1)\cdot\alpha_{j-1}}\cdot b_k^* =$$

$$= -\frac{f_j}{\alpha_{j-1}}\cdot b_k^* = b_k^*\cdot\lambda_j = g_k^*\cdot\lambda_j \quad where \quad \lambda_j = -\frac{f_j}{\alpha_{j-1}} \,\&\, g_k = b_k \quad ; \quad \text{E}-29$$

The U-D factors are given by eqn. E-9, E-21, E-27, E-28 and E-29 as:

$$\bar{f}_1 = s_1^*$$

$$\bar{f}_j = \sum_k^{j-1}\mu_{kj}\cdot s_k^* + s_j^* \quad ; \quad \text{j} = 2,3,.,.,\ldots..,\text{L} \qquad \text{E-30}$$

from $\text{E}-5 \Rightarrow d_j(i) = (1+q)\bar{d}_j(i) = h_q\cdot\bar{d}_j(i)$ where:

$$\bar{d}_1(i) = d_1(i-1)\cdot\frac{\alpha_0}{\alpha_1} = \frac{d_1(i-1)}{\alpha_1}\eta \Rightarrow d_1(i) = \frac{d_1(i-1)}{\alpha_1}\cdot h_q\cdot\eta \qquad \text{E-31}$$

$$\bar{d}_j(i) = d_j(i-1)\cdot\frac{\alpha_{j-1}}{\alpha_j} \Rightarrow d_j(i) = d_j(i-1)\frac{\alpha_{j-1}}{\alpha_j}\cdot h_q \quad ; \quad \text{for} \quad \text{j} = 1,2,\ldots..,\text{L}$$

$$\bar{\mu}_{kj} = -\frac{b_j}{\alpha_j\cdot\bar{d}_j}\cdot b_k^* = -\frac{f_j}{\alpha_{j-1}}\cdot b_k^* = b_k^*\cdot\lambda_j = g_k^*\cdot\lambda_j \text{ where } \lambda_j = -\frac{f_j}{\alpha_{j-1}} \,\&\, g_k = b_k \; ; \quad \text{E-32}$$

$$\alpha_j = \alpha_{j-1} + b_j\cdot f_j^* \; ; \text{ for } j = 2,\ldots..,L \;\; \& \;\; \alpha_1 = \alpha_0 + d_1\cdot|f_1|^2 = \eta + b_1\cdot f_1^* \qquad \text{E-33}$$

The intermediate Kalman gain G and the error covariance factors U and D can be obtained from eqn.E-5, as follows :

The element of the upper triangular matrix U which is given by eqn.E-5 can be calculated using :

$$[U_i]_{k,j} = [U_{i-1}]_{k,j}\cdot[\bar{U}_{i-1}]_{j,m} \Rightarrow$$

for $j = 2,3,.....,L \quad ; \quad k = 1,2,.....,j-1 \Rightarrow$

$$\mu_{kj}(i) = \mu_{kj}(i-1) + \overline{\mu}_{kj}(i-1) + \sum_{m=k+1}^{j-1} \mu_{mj}(i-1) \cdot \overline{\mu}_{mj}(i-1) \quad ; \quad \text{using eqn. E - 32} \Rightarrow$$

$$\mu_{kj}(i) = \mu_{kj}(i-1) + b_k^* \cdot \lambda_j + \sum_{m=k+1}^{j-1} \mu_{mj}(i-1) \cdot b_m^* \cdot \lambda_j =$$

$$= \mu_{kj}(i-1) + \lambda_j [b_k^* + \sum_{m=k+1}^{j-1} \mu_{mj}(i-1) \cdot b_m^*] \qquad \text{E-34}$$

The kalman gain is given by :

$$G_i = \alpha^{-1} \cdot \Phi_{i-1} \cdot S_i^* = \alpha_L^{-1} \cdot U_{i-1}^* \cdot D_{i-1} \cdot U_{i-1}^T \cdot S_i^* = \alpha_L^{-1} \cdot U_{i-1}^* \cdot D_{i-1} \cdot F_{i-1}$$
$$= \alpha_L^{-1} \cdot U_{i-1}^* \cdot B_{i-1} \quad ; \quad \text{Now let} \quad G'_i = U_{i-1}^* \cdot B_{i-1} \Rightarrow$$

The element of the gain vector G'_i can be calculated using :

$$g_L = b_L \quad \& \quad g_k = \sum_{j=k+1}^{L} b_j \cdot \mu_{kj}^* + b_k \quad \text{for} \quad k = 1,2,.....,L-1 \quad \text{or in another form :}$$

$$g_L = b_L \quad \& \quad g_k = b_k + b_j \cdot \mu_{kj}^* \quad \text{for} \quad j = 2,3,.....,L \; ; \; k = 1,2,.....,j-1 \qquad \text{E-35}$$

Note: $g_k^* = b_k^* + b_k^* \cdot \mu_{kj}$

From eqn.E-34 and E-35 we can write :

Let $g_j = b_j \quad ; \quad \text{for} \quad j = 1,2,.....,L$; then:

for $j = 2,.....,L$

$$\beta_1 = \mu_{kj}(i-1) \; ; \; k = 1,2,.....,j-1 \qquad \text{E-36}$$

$$\mu_{kj}(i) = \beta_1 + g_k^* \cdot \lambda_j$$

$$g_k(i) = g_k + g_j \cdot \beta_1^*$$

From the above equations, the final algorithm can be given by :

$$e_i = r_i - r'_i \qquad (1)$$
$$f_1 = s_1^* \qquad (2)$$
$$g_1 = d_1 \cdot f_1 \qquad (3)$$
$$\alpha_1 = \eta + g_1 \cdot f_1 \qquad (4)$$

For j = 2, L

$$f_j = \sum_{k=1}^{j-1} \mu_{k\,j} \cdot s_k^* + s_j^* \qquad (5)$$

$$g_j = d_j \cdot f_j \qquad (6)$$

$$\alpha_j = \alpha_{j-1} + g_j \cdot f_j \qquad (7)$$

Next j

$$\gamma = 1/\alpha_1 \qquad (8)$$

$$d_1 = d_1 \cdot h_q \cdot \eta \cdot \gamma \qquad (9)$$

where $h_q = 1/w$

For j = 2, L

$$\beta = \alpha_{j-1} \qquad (10)$$

$$\lambda_j = -f_j \cdot \gamma \qquad (11)$$

$$\gamma = 1/\alpha_j \qquad (12)$$

$$d_j = d_j \cdot h_q \cdot \beta \cdot \gamma \qquad (13)$$

For k = 1, j-1

$$\beta_1 = \mu_{kj} \qquad (14)$$

$$\mu_{kj} = \beta_1 + g_k^* \cdot \lambda_j \qquad (15)$$

$$g_k = g_k + g_j \cdot \beta_1^* \qquad (16)$$

Next k

Next j

$$\varepsilon = e_i \cdot \gamma \qquad (17)$$

for j=1,2,. . . . . ,L

$$p'_j = p'_{j-1} + g_j \cdot \varepsilon \qquad (18)$$

where $\{ p'_j \}$ are the elements of the estimated SIR vector (P'_j).
All quantities are complex accept w, μ , h_q , γ, β, $\{dj\}$ ($1 <= j <= L$) and $\{\alpha j\}$ ($1 <= j <= L$).
At the start of the process (at $t=0$)

$$d_j = 1.0 \quad \text{for } j = 1,2,. . . ..,L \qquad (19)$$

$\mu_{kj} = 0.0+J0.0$ for $k=1,2,.....,(L-1)$.
and j= $k+1, k+2,. . . . .,L$

Equations (1) to (19) represent the full SRK algorithm [48] .

APPENDIX F

GENERATION OF NON-MINIMUM AND MINIMUM PHASED SIR OF THE OVERALL LINEAR BASE-BAND CHANNEL USED IN THE TESTS.

```
      PROGRAM GSIR

c     Finale program for  generation the overall sampled impulse response 'SIR' of the linear base
c     band channel , where the transmission path is a model of HF radio links, with 2 or 3 sky
c     waves. This program gives  the actual SIR of the channel, as described in chapter 2, and
c     then apply the minimum phase algorithm to form the minimum phased SIR of the channel
c     as described in chapter 4.

      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      PARAMETER (nk=6,k=30,kp=20,ntap=50)
      DOUBLE PRECISION  Q(nk,30600),QQ(nk),eq(nk),vq(nk),G05DDF
      DOUBLE PRECISION  fmean(nk),fvar(nk),varbf3dcg(nk),con(nk)
      DOUBLE PRECISION  pr(k),pi(k),qr(k,k),qi(k,K)
      DOUBLE PRECISION  txfr(16),txfi(16),txfdr(16),txfdi(16)
      DOUBLE PRECISION  rxfr(k),rxfi(k)
      DOUBLE PRECISION  txfddr(16),txfddi(16)
C     DOUBLE PRECISION txfdddr(16),txfdddi(16)

      COMMON Q,QQ
      COMMON/adf/ ymr(kp),ymi(kp),syr(kp),syi(kp),yfr(ntap+1),yfi(ntap+1)
      COMMON/adfa/ ggy(kp),ggsy(kp),ggsyf(ntap)
      COMMON/padf/ nproot
      COMMON/stdf/ splr(kp),spli(kp)

C     TRANSMITTER FILTER (0.0msec. delay).
C     ------------------------------------------------------------
      DATA txfr /-0.1795896,-3.0773455,-9.9409021,-11.7869473,
     1          -3.4618271,4.4438154,3.0642536,-1.3596576,
     1          -1.4973528,0.2925598,0.5180829,-0.1842786,
     1          -0.3167778,0.0021899,-0.0443806,0.0515533/
      DATA txfi /2.3539405,20.7590237,45.5584592,41.4909978,
     1          8.7045826,-11.7869820,-5.5819054,3.1582131,
     1          1.7365460,-0.7776891,-0.1292556,0.2880296,
     1          -0.2324818,-0.2107548,0.0392056,0.0098505/

C     TRANSMITTER FILTER (1.1msec delay).
C     ------------------------------------------------------------
      DATA txfdr /-1.6694374,-7.8492148,-12.3887079,-6.6023157,
     1          2.9408554,4.3005084,-0.3368383,-1.9014342,
     1          -0.1433592,0.6242601,0.0278577,-0.3820071,
     1          -0.0416905,-0.0439705,0.0749333,-0.0594132/
      DATA txfdi /13.2372707,39.6493461,46.9272219,19.2346609,
     1          -8.8804125,-9.0256163,1.6284281,2.8139013,
     1          -0.4311352,-0.4537174,0.3081762,-0.0772327,
     1          -0.3043271,0.0085057,0.0093809,0.0094992/

C     TRANSMITTER FILTER (3.0msec delay).
C     ------------------------------------------------------------
      DATA txfddr /-1.3136537,-7.1104051,-12.3469721,-7.5848703,
     1          2.2353854,4.5938614,0.0931639,-1.9704176,
     1          -0.3233694,0.6313238,0.1035718,-0.3865939,
     1          -0.0734526,-0.0386471,0.0608046,-0.0713496/
```

```
      DATA txfddi /11.0688962,37.2136597,47.9575159,22.8262482,
     1            -7.2498590,-10.0026703,0.8695437,3.1072800,
     1            -0.2261096,-0.5552906,0.2882096,-0.0156703,
     1            -0.3215770,-0.0107706,0.0140909,0.0135711/

C    TRANSMITTER FILTER (2.0msec delay). For channel 3  simulation only.
C    ------------------------------------------------------------------------------------------
C        DATA txfdddr /-0.7630,-5.6487,-11.9216,-9.3589,
C    1            0.5650,4.9376,1.0473,-1.9766,
C    1            -0.7165,0.5944,0.2544,-0.3636,
C    1            -0.1544,-0.0228,0.0167,-0.0610/
C        DATA txfdddi /7.3452,31.9050,48.7718,29.8080,
C    1            -3.0208,-11.4980,-0.9823,3.5053,
C    1             0.3116,-0.7219,0.2045,0.1085,
C    1            -0.3287,-0.0636,0.0279,0.0186/

C    RECEIVER FILTER SAMPLES.
C    ------------------------------------------------
        DATA rxfr /-1.9417691,-15.9797864,-35.1417733,-34.4788717,
     1            -11.2301982,7.8155160,7.5124057,-0.5057505,
     1            -3.3707125,-0.6759166,1.0482656,0.3621876,
     1            -0.3105902,0.0438410,0.0738947,-0.0646936,
     1            0.00000000,0.00000000,0.00000000,0.00000000,
     1            0.00000000,0.00000000,0.00000000,0.00000000,
     1            0.00000000,0.00000000,0.00000000,0.00000000,
     1            0.00000000,0.00000000/
        DATA rxfi /1.3625952,11.5941040,27.3342937,28.0870086,
     1            7.2714615,-9.2602472,-5.0954462,3.2326498,
     1            1.8975352,-1.2813604,-0.4830313,0.7614804,
     1            0.1979014,-0.1532672,0.0940330,-0.0312132,
     1            0.00,0.00,0.00,0.00,
     1            0.00,0.00,0.00,0.00,
     1            0.00,0.00,0.00,0.00,
     1            0.00,0.00/

C       OPEN(UNIT=10,FILE='nmphaser1',FORM='UNFORMATTED')
        OPEN(11,FILE='fmphaser1',FORM='UNFORMATTED')
        OPEN(26,FILE='moutch1',FORM='FORMATTED')

C    RAYLEIGH FADING SIMULATOR.

        NSAM=2480
C    Channel 1.
        ID=75
        NSKY=3
        Fsp=2
        DCG=19659.0
C    Channel 2.
C       ID=57
C       NSKY=2
C       Fsp=2
C       DCG=15270.0
C    Channel 3.
C       ID=57
C       NSKY=2
C       Fsp=1
C       DCG=310590.3
C       DCG=1.0D0
```

```
      DCG=1.0/DCG

C     Filter coefficient for  Fsp=2.0Hz and sampling rate (SR =100).
      C1=-1.80322972
      C2=0.8152066804
      C3=-1.852182882
      C4=0.8678845458
      C5=-0.8948130729

C     Filter coefficient for  Fsp=1.0Hz and sampling rate (SR =100).
C     C1=-1.899738312
C     C2=0.902887967
C     C3=-1.927534516
C     C4=0.931603213
C     C5=-0.945945597

      STDVN1=SQRT(1.0)

C     Initialising The Delays In The Filter.
      DT1=0.0
      DT2=0.0
      DT3=0.0
      DT4=0.0
      DT5=0.0
      DMG=0.0
C     Seed Integer Generator.
      CALL G05CBF(ID)

      JQ=30+NSAM
      JQ1=JQ+1
C     Generation Q_i samples.
      DO 250 I=1,nk
      JA=1
C     Testing The Run.
        TBF3=0.0
        TVBF3=0.0
        TBF3DCG=0.0
        TVBF3DCG=0.0
C       BF3DCG=0.0
C       BF3=0.0
        DO 240 J=1,JQ1
C     Generating The Guassian Input.
C           BF0=1.0d0
            BF0=G05DDF(0.0d0,STDVN1)
            BF1=BF0-(DT1*C1+DT2*C2)
            BF2=BF1-(DT3*C3+DT4*C4)
            BF3=BF2-(DT5*C5)
            BF3DCG=BF3*DCG
            DT5=BF3
            DT4=DT3
            DT3=BF2
            DT2=DT1
            DT1=BF1
            IF(J.LE.31) GO TO 240
            Q(I,JA)=BF3DCG
            JA=JA+1
            TBF3=TBF3+BF3
            TVBF3=TVBF3+(BF3**2)
            TBF3DCG=TBF3DCG+BF3DCG
```

```
            TVBF3DCG=TVBF3DCG+(BF3DCG**2)
240         CONTINUE

C       Printing The Results.
            EBF3=TBF3/(NSAM+1)
            VARBF3=TVBF3/(NSAM+1)
            EBF3DCG=TBF3DCG/(NSAM+1)
            VARBF3DCG(I)=TVBF3DCG/(NSAM+1)

            WRITE(26,244)EBF3,VARBF3
            WRITE(26,246)EBF3DCG,VARBF3DCG(I)
244         FORMAT('MEANOFBF3DCG=',1X,E20.10,2X,'Var. of BF3=',1X,E20.10)
246         FORMAT('Mean of BF3DCG=',1X,E20.10,2X,'Variance of BF3DCG=',E20.10)
            DMG=DMG+VARBF3DCG(I)

250         CONTINUE
            DCG=SQRT(DMG)
C           PRINT * ,'TESTED VALUE OF DCG=',DCG

C       Values of variables.
            pio=acos(-1.0)
            nproot=0
            nloop=1230
            mloop=2*nloop
            istep=48
            step=1.0/istep
c       Time delay for channel 1.
            del1=1.1
            del2=3.0
C       Time delay for channel 2.
C           del1=3.0
C           del2=0.0
C       Time delay for channel 3.
C           del1=2.0
C           del2=0.0
            samprat=2.4
            sfact=1.0/(samprat*2*1000.0)
            idel1=int(samprat*2*del1)
            idel2=int(samprat*2*del2)
            kmpl=16
C       For channel 1.
            kmp=idel2+kmpl
C       For channel 2 and 3.
C           kmp=idel1+kmpl
            kmp1=kmp-1
            kmconv=(kmp+kmpl-1)/2
            indtn=75
            pos=-1.0
            sq=0.0
            jcount=0
            icount=0
            nkcount=0
            DO 1010 i=1,kmp
               DO 1005 j=1,kmp
                 qr(i,j)=0.0
                 qi(i,j)=0.0
1005           CONTINUE
1010        CONTINUE
            DO 1011 i=1,kp
```

```
            ymr(i)=0.0
            ymi(i)=0.
101         CONTINUE
            DO 1020 i=1,nk
               eq(I)=0.0
               vq(I)=0.0
1020        CONTINUE
            CALL G05CBF(indtn)

C     Entering main loop.
C     ---------------------------
            DO 7000 KMAIN=1,MLOOP

C     Linear interpolation.
               DO 1510 i=1,nk
                  con(i)=(q(I,kmain+1)-q(i,kmain))*step
1510           CONTINUE

C     Entering secondary loop.
C     ----------------------------------
               DO 6000 KSEC=1,ISTEP

                 icount=icount+1
                 count=real(icount)
                 DO 1520 i=1,nk
                    qq(i)=q(i,kmain)+((ksec-1)*con(I))
1520             CONTINUE
                 DO 1530 i=1,nk
                      eq(I)=eq(I)+qq(I)
                      vq(I)=vq(I)+(qq(I)*QQ(I))
1530             CONTINUE
                 qq(2)=-qq(2)
                 qq(4)=-qq(4)
                 qq(6)=-qq(6)
C     Shifting arrays for convolution process.
                 DO 2010 i=1,kmp
                    DO 2005 j=1,kmp1
                       qr(i,kmp+1-j)=qr(i,kmp-j)
                       qi(i,kmp+1-j)=qi(i,kmp-j)
2005                CONTINUE
2010             CONTINUE
                 DO 2015 i=1,kmp
                  qr(I,1)=0.0
                  qi(I,1)=0.0
2015              CONTINUE

C     Convolution (to obtain the impulse response of the channel), begins :
                 DO 2020 i=1,kmpl
                    qr(i,1)=txfr(I)*QQ(1)-txfi(I)*QQ(2)
                    qi(I,1)=txfr(i)*QQ(2)+txfi(I)*QQ(1)
2020             CONTINUE
C     For delay =1.1ms.
                 DO 2030 i=1,kmpl
                    qr(i+idel1,1)=qr(i+idel1,1)+txfdr(I)*QQ(3)-txfdi(i)*QQ(4)
                    qi(i+idel1,1)=qi(i+idel1,1)+txfdr(I)*QQ(4)+txfdi(i)*QQ(3)
2030             CONTINUE
c     For delay =3.0ms.
                 DO 2040 i=1,kmpl
                    qr(i+idel2,1)=qr(i+idel2,1)+txfddr(i)*qq(5)-txfddi(i)*qq(6)
```

```
                qi(i+idel2,1)=qi(i+idel2,1)+txfddr(i)*qq(6)+txfddi(i)*qq(5)
2040            CONTINUE
                pos=-pos
                IF (pos.lt.0) go to 6000
                io=0
                jcount=jcount+1
                dcount=real(jcount)
C               time=dcount/2400.0
                DO 2060 i=1,kmp,2
                   io=io+1
                    pr(io)=0.0
                    pi(io)=0.0
                    DO 2050 j=1,i
                       pr(io)=pr(io)+qr(j,i+1-j)*rxfr(i+1-J)-qi(j,i+1-j)*rxfi(i+1-j)
                       pi(io)=pi(io)+qi(j,i+1-j)*rxfr(i+1-J)+qr(j,i+1-j)*rxfi(i+1-J)
2050                CONTINUE
                       pr(io)=pr(io)*sfact
                       pi(io)=pi(io)*sfact
2060            CONTINUE
                IF (mod(kmp,2).eq.0) then
                  go to 2070
                ELSE
                  go to 2100
                END IF
2070            DO 2090 i=1,kmp1,2
                   io=io+1
                   pr(io)=0.0
                   pi(io)=0.0
                   mconv=i+1
                   DO 2080 j=mconv,kmp
                      kcvt=kmp+1+i-j
                      pr(io)=pr(io)+qr(j,kcvt)*rxfr(kcvt)-qi(j,kcvt)*rxfi(kcvt)
                      pi(io)=pi(io)+qi(j,kcvt)*rxfr(kcvt)+qr(j,kcvt)*rxfi(kcvt)
2080                CONTINUE
                   pr(io)=pr(io)*sfact
                   pi(io)=pi(io)*sfact
2090            CONTINUE
                go to 2150

C     End of even convolution process.
2100            DO 2120 i=1,kmp1,2
                   io=io+1
                   pr(io)=0.0
                   pi(io)=0.0
                   mconv=i+2
                   DO 2110 j=mconv,kmp
                      pr(io)=pr(io)+qr(j,kmp+2+i-j)*rxfr(kmp+2+i-j)
     1                        -qi(j,kmp+2+i-j)*rxfi(kmp+2+i-j)
                      pi(io)=pi(io)+qi(j,kmp+2+i-j)*rxfr(kmp+2+i-j)
     1                        +qr(j,kmp+2+i-j)*rxfi(kmp+2+i-j)
2110                CONTINUE
                pr(io)=pr(io)*sfact
                pi(io)=pi(io)*sfact
2120            CONTINUE
2150            CONTINUE

                IF (jcount.gt.400) then
                      nkcount=nkcount+1
                      pcount=REAL(nkcount)
                      DO 2159 j=1,kp
```

```
                    ymr(j)=pr(j)
                    ymi(j)=pi(j)
2159                CONTINUE
C   Store the Actual SIR of the channel.
C                   write(10)(pr(i),i=1,kmconv-2)
C                   write(10)(pi(i),i=1,kmconv-2)

                    CALL ADFILT(nkcount)

C   Store the minimum phased SIR of the channel.
                    write(11)(syr(i),i=1,kp)
                    write(11)(syi(i),i=1,kp)
C   Checking the sum of the squares.
                    YYSQ=0.0
                    ysq=0.0
                    DO 6666 I=1,kp
                        YSQ=YSQ+(PR(I)*PR(I)+PI(I)*PI(I))
                        YYSQ=YYSQ+(SYR(I)*SYR(I)+SYI(I)*SYI(I))
6666                CONTINUE
                    SQ=SQ+YSQ
                    SSQ=SSQ+YYSQ
C   For three dimension graphs.
                    DO 6001 J=1,kp
                        GGY(J)=PR(J)*PR(J)+PI(J)*PI(J)
                        GGSY(J)=SYR(J)*SYR(J)+SYI(J)*SYI(J)
6001                CONTINUE

C                   DO 5950 I=1,NTAP
C                       GGSYF(I)=YFR(I)*YFR(I)+YFI(I)*YFI(I)
C 5950                  CONTINUE
C                   write(27)(ggy(j),j=1,20)
C                   write(28)(ggsy(j),j=1,20)
C                   write(29)(ggsyf(j),j=1,50)
                ENDIF

6000            CONTINUE
7000    CONTINUE

        DO 7010 i=1,nk
            fmean(i)=eq(i)/count
            fvar(i)=vq(i)/count
7010    CONTINUE
        avsq=sq/pcount
        avssq=ssq/pcount

C   Printing The Results.
        write(26,9203)fsp,del1,del2,kmp,samprat
9203    format('frequency spread    =',f6.2/
   1            'delay between sky-waves no.1and2=',f6.3/
   1            'delay between sky-waves no.2and3=',f6.3/
   1            'number of elements in SIR=',i2/
   1            'sampling rate in kilo-baud   =',f6.3)

        write(26,9210)indtn,icount,jcount,kmconv,mloop,nsam,nsky,nkcount
9210    format('indtn=',i7/'icount=',i7/'jcount=',i7/'kmconv=',i4/
   1            'mloop=',i5/'nsam=',i5/'nsky=',i2/'nkcount=',i7/)
        write(26,8105)
8105    format('result of MPA : converting the SIR to Min. phased')
        write(26,8170)
```

```
8170    format('the square sum of SIR is :')
        write(26,8171)avsq,avssq
8171    format(1x,f21.10,2x,f21.10)
        write(26,8175)
8175    format(' The last sample impulse response =')

        DO 8180 i=1,kp
             write(26,8182)pr(i),pi(i),ggy(i),syr(i),syi(i),ggsy(i)
8180    CONTINUE
8182    format(1x,f10.6,2x,f10.6,1x,f11.7,3x,f10.6,2x,f10.6,1x,f11.7)
        write(26,9215)
9215    format('mean value of the QQ-signals')
        write(26,9225)(fmean(i),i=1,nk)
        write(26,9220)
9220     format(/'variances of the QQ-signals')
        write(26,9225)(fvar(i),i=1,nk)
9225     format(f20.10/)
        write(26,9227)dcg
9227     format('DC-GAIN = ',2x,f20.8)

9265        CONTINUE
              STOP
              END


                    SUBROUTINE ADFILT(nkcount)
c     This subroutine is the minimum phase algorithm for allocating the roots of the sample
c       impulse response of the HF channel. The resultant SIR will be minimum-phased.

        IMPLICIT DOUBLE PRECISION (A-H,O-Z)
        PARAMETFER(KP=20,NSIR=20,NTAP=50,NEST=NSIR,NINRT=9)
        DOUBLE PRECISION fer(nsir+1),fei(nsir+1)
        DOUBLE PRECISION afr(nsir+1),afi(nsir+1)
        DOUBLE PRECISION flreal(nsir),flimg(nsir)
        DOUBLE PRECISION spfr(ninrt),spfi(ninrt)
        DOUBLE PRECISION azr(ntap+1),azi(ntap+1)
c       DOUBLE precision fpreal(nsir),fpimg(nsir)

        COMMON/adf/ymr(kp),ymi(kp),syr(kp),syi(kp),yafr(ntap+1),yafi(ntap+1)
        COMMON/adfa/ggy(kp),ggsy(kp),ggsyf(ntap)
        COMMON/padf/nproot
        COMMON/stdf/splr(kp),spli(kp)

C     The nine starting point .
        DATA spfr /0.00000,0.90909,0.00000,0.00000,-0.90909,
     1              0.64282,0.64282,-0.64282,-0.64282/
        DATA spfi /0.00000,0.00000,-0.90909,0.90909,0.00000,
     1              -0.64282,0.64282,0.64282,-0.64282/

C     Values of variables and arrays.
        ntapm1=ntap-1
        ntapm2=ntap-2
C     Values of variables and arrays for filter.
        fc=1.0
        fdelta=10.0e-10
        fd=1.05
        falthd=1.0/fd
        nroot=0
        nstrpf=1
```

```
      nsirp1=nsir+1
      nsirp2=nsir-2
      ntapp1=ntap+1
      nfitr=40

      ispfr=int(spfr(ninrt)*100.0)
      ispfi=int(spfi(ninrt)*100.0)

      IF (nkcount .eq. 1) then
            DO 50 i=1,ninrt
                splr(i)=spfr(i)
                spli(i)=spfi(i)
 50             CONTINUE
                DO 60 i=ninrt+1,nsir
                    splr(i)=0.0
                    spli(i)=0.0
 60             CONTINUE
            ENDIF
            DO 100 i=1,nsir
                syr(i)=ymr(i)
                syi(i)=ymi(i)
                flreal(i)=0.0
                flimg(i)=0.0
 100        CONTINUE

 200        CONTINUE

C     One tap feedback filter.
            fcon1r=splr(nstrpf)
            fcon1i=spli(nstrpf)
            DO 1000 i=1,nfitr
                DO 510 j=1,nsirp1
                    fer(j)=0.0
                    fei(j)=0.0
 510            CONTINUE
                fcon2r=0.0
                fcon2i=0.0
                DO 520 j=1,nsir
                    j1=nsir+1-j
                    fxr=syr(j1)-(fcon1r*fcon2r-fcon1i*fcon2i)
                    fxi=syi(j1)-(fcon1r*fcon2i+fcon1i*fcon2r)
                    fcon2r=fxr
                    fcon2i=fxi
                    fer(j1)=fxr
                    fei(j1)=fxi
 520            CONTINUE
C     Estimate value of epsilon.
                    fcon3r=-fcon1r
                    fcon3i=-fcon1i
                    fepsr=fer(nsir)
                    fepsi=fei(nsir)
                    DO 530 J=1,NSIRP2
                        j2=nsir-j
                       pfcon3r=fepsr*fcon3r-fepsi*fcon3i
                        pfcon3i=fepsr*fcon3i+fepsi*fcon3r
                        fepsr=pfcon3r+fer(j2)
                        fepsi=pfcon3i+fei(j2)
C                       PRINT *,FEPSR,FEPSI
 530                CONTINUE
```

```
C    Obtain conditions for convergence.
                fepsam=fepsr*fepsr+fepsi*fepsi
C               PRINT *,FEPSAM,'FEPSR=',FEPSR,FEPSI
                fdelr=(fer(1)*fepsr+fei(1)*fepsi)/fepsam
                fdeli=(fei(1)*fepsr-fer(1)*fepsi)/fepsam
C               PRINT *,'FDELR=',FDELR,FDELI
                fdelma=fdelr*fdelr+fdeli*fdeli
C               PRINT *,'FDELMA=',FDELMA

C    As long as convergence is not achieved or algorithm  has not diverged, repeat the iteration.
                IF  (fdelma.lt.fdelta) then
                   go to 2050
                ELSE
                   fdelr=fc*fdelr
                   fdeli=fc*fdeli
                   fcon1r=fcon1r+fdelr
                   fcon1i=fcon1i+fdeli
                   dl1=sqrt(fcon1r*fcon1r+fcon1i*fcon1i)
                   IF  (dl1.gt.falthd) then
                       go to 1050
                    ENDIF
                ENDIF

1000    CONTINUE

1050    CONTINUE

C    Find new starting point after divergence.
IF (nkcount .eq. 1) then
       IF (nstrpf.lt.ninrt) then
           nstrpf=nstrpf+1
           go to 200
        ELSE
           go to 2200
       ENDIF
ELSE
       isplr=int(splr(nstrpf)*100.0)
       ispli=int(spli(nstrpf)*100.0)
       IF (isplr.eq.ispfr.and.ispli.eq.ispfi) then
           go to 2200
        ELSE
           nstrpf=nstrpf+1
           go to 200
       ENDIF
ENDIF

2050    CONTINUE

C    Alter channel with obtained root by passing through two tap feed forward filter.
       nroot=nroot+1
       flreal(nroot)=fcon1r
       flimg(nroot)=fcon1i
       afr(1)=fer(1)
       afi(1)=fei(1)
       DO 540 j=2,nsirp1
         j3=j-1
            afr(j)=fer(j)+fer(j3)*fcon1r+fei(j3)*fcon1i
            afi(j)=fei(j)+fei(j3)*fcon1r-fer(j3)*fcon1i
```

```
                syr(j3)=afr(j)
                syi(j3)=afi(j)
 540      CONTINUE

C     Find new starting point after convergence.
          IF (nkcount.eq.1) then
                IF (nstrpf.lt.ninrt) then
                   nstrpf=nstrpf+1
                   go to 200
                ELSE
                   go to 2200
                ENDIF
          ELSE
                IF (nstrpf.le.nproot) then
                   nstrpf=nstrpf+1
                   go to 200
                ELSE
                   go to 2200
                ENDIF
          ENDIF

 2200     CONTINUE

C     Setting up starting points for next time instant when all roots have been found.
          ntot1=nroot+ninrt+1
          nproot=nroot
          DO 2250 j=1,nroot
                splr(j)=flreal(j)
                spli(j)=flimg(j)
 2250     CONTINUE
          DO 2260 j=1,ninrt
                splr(nroot+j)=spfr(j)
                spli(nroot+j)=spfi(j)
 2260     CONTINUE
          IF (ntot1.le.nsir) then
                DO 2270 j=ntot1,nsir
                   splr(j)=0.0
                   spli(j)=0.0
 2270           CONTINUE
          ELSE
                go to 3000
          ENDIF
 3000     CONTINUE

C         DO 3005 J=1,NROOT
C              FEATR=FLREAL(J)
C              FEATI=FLIMG(J)
C              RT=FEATR*FEATR+FEATI*FEATI
C              FPREAL(J)=(-1.0*FEATR)/RT
C              PRINT *,'RT',RT,FPREAL(J)
C              FPIMG(J)=FEATI/RT
C              PRINT *,'J=',J,FPIMG(J)
C 3005   CONTINUE
C     At this point (syr) and (syi) contain the minimum phased version of (fcon2r) and (fcon2i)
C     for the present time instant. Start calculation of the tap gains of the filter.
C               Initialise tap gains.
            DO 4500 J=1,NTAPP1
                YFR(J)=0.0
                YFI(J)=0.0
```

```
4500    CONTINUE
        YFR(NTAP)=1.0

C   Adjust the tap gains of the filter by passing each converged root in accordance with the
C   given algorithm.
        DO 5000 IJK=1,NROOT
            FGMAR=FLREAL(IJK)
            FGMAI=FLIMG(IJK)
C       PRINT *,IJK,FGMAR,FGMAI,NROOT
        DO 4610 J=1,NTAPP1
            AZR(J)=0.0
            AZI(J)=0.0
4610    CONTINUE

C   Feeding through two tap feedforward filter.
        AZR(1)=YFR(1)
        AZI(1)=YFI(1)
        DO 4620 J=1,NTAP
            J4=J+1
            AZR(J4)=YFR(J4)+(YFR(J)*FGMAR+YFI(J)*FGMAI)
            AZI(J4)=YFI(J4)+(YFI(J)*FGMAR-YFR(J)*FGMAI)
4620    CONTINUE

C   Feeding output from above filter through a one tap feedback filter. The sequence is fed in
C   reverse order. the output from this one tap filter is also obtained in reverse order.
        FQGR=0.0
        FQGI=0.0
        DO 4630 J=1,NTAP
            J5=NTAP+2-J
            QAFR=AZR(J5)-(FGMAR*FQGR-FGMAI*FQGI)
            QAFI=AZI(J5)-(FGMAR*FQGI+FGMAI*FQGR)
            FQGR=QAFR
            FQGI=QAFI
            YFR(J5-1)=QAFR
            YFI(J5-1)=QAFI
4630    CONTINUE

5000    CONTINUE


        RETURN
        END
```

APPENDIX G

COMBINED DETECTOR, ESTIMATOR AND ADAPTIVE EQUALISER FOR 4800 BIT/SEC. HF MODEM.

```
      PROGRAM COMSYS

c     This program is a combined estimator detector that takes sampled impulse response
c     (SIR) of  HF channel that are non-minimum phased and uses an adaptive algorithm,
c     differential coding,  channel estimator, and modified near-maximum likelihood
c     detector.

      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      PARAMETER (kv=16,nsir=20,ntap=50,n=81,nn=32,ninrt=9,nexp=4)
      PARAMETER (kv1=2,kv2=4,kv3=kv-kv1-kv2,kvtot=4*kv1+2*kv2+kv3)
      LOGICAL iosrk,idec,iest,ifgm,ipred
      COMMON/dat/iar(n),iai(n),ib1(n),ib2(n),idb1(n),idb2(n)
      COMMON/symb/ar(n),ai(n)
      COMMON/lkupt/itb1(32),itb2(32),itb3(32),itlp1(16,2),itlp2(16,2)
      COMMON/clkup/itlp3(16,2),indsr(nexp),indsi(nexp)
      COMMON/rfnois/hfr(nsir),hfi(nsir),wfr(nsir),wfi(nsir)
      COMMON/main1/rr(ntap),ri(ntap),ydi(ntap,nsir),ydr(ntap,nsir)
      COMMON/mpnor/yfnr(ntap),yfni(ntap),ympnr(nsir),ympni(nsir)
      COMMON/mpadf/yr(nsir),yi(nsir),ympr(nsir),ympi(nsir)
      COMMON/adf/yfr(ntap+1),yfi(ntap+1),spfr(ninrt),spfi(ninrt)
      COMMON/adfsplr/splr(nsir),spli(nsir)
      COMMON/fpr/jfroot
      COMMON/srk/uur(nsir*nsir),uui(nsir*nsir),dd(nsir),bb(nsir)
      COMMON/estm/yestr(nsir),yesti(nsir),ycr(nsir),yci(nsir),b(nsir)
      COMMON/isi/gr(kv),gi(kv)
      COMMON/det/cost(kv),ixr(kv,nn),ixi(kv,nn),xr(kv,nn),xi(kv,nn)
      COMMON/result/iersym,ier1,ier2,ider1,ider2,ipdb1,ipdb2
      COMMON/rec/rrr,www,xx,stdvn
      COMMON/parm/nsam,icount,itrain,jwrm,irpts,n1,nn1,iq,NCH
      COMMON/pardet/idec
      COMMON/lest/iosrk,iest,ifgm,ipred
      COMMON/paradf/ntapp1,nsirp1,nsirp2,nfitr,fc,fd,fad,faltha
      COMMON/parest/omega,qq,hqq,eata,theta,theta1,theta2,alfa,delta,bc
      COMMON/rslest/snr,nwrm,count,tprer,yeror
      COMMON/funit/llout,indat

c     INIPAR subroutine which inputs all parameters and constants for estimater, adaptive filter
c     and detector.
          CALL INIPAR()
      stdvn=10.0**(-snr/20.0)
          CALL G05CBF(IQ)

c     Initialisation subroutine which initialises all arrays, vectors and parameters for detector,
c     estimater.
              CALL INITIL()

c     Entering the main loop.
       DO 40 ITIME=1,NSAM
           icount=icount+1
      count=real(icount)
```

```
c     Training sequences subroutine.
            CALL TRAINSH()

        IF (TPRER.GT.500000.0) GOTO 50

            IF (ICOUNT.EQ.1.OR.ICOUNT.EQ.NTAP.OR.MOD(ICOUNT,8).EQ.0)THEN
c     Filter subroutine which gives an estimate of the current  min. phased HF channel and the
c     corresponding filter taps.
              CALL ADFILT(ICOUNT)
            ENDIF

c     Obtain scaled version of the min. phased channel and appropriately scaled version of the
c     filter taps. These will be held in arrays (ympnr), (ympni), (yfnr) and (yfni)  respectively.
        sysqq=ympr(1)*ympr(1)+ympi(1)*ympi(1)

c     Normalised the Min. phase S.I.R.
            do 20 j=1,nsir
                ympnr(j)=(ympr(1)*ympr(j)+ympi(1)*ympi(j))/sysqq
                ympni(j)=(ympr(1)*ympi(j)-ympi(1)*ympr(j))/sysqq
 20         continue

c     Normalised the tap-gains of the adaptive filter
            do 25 i=1,ntap
                yfnr(i)=(ympr(1)*yfr(i)+ympi(1)*yfi(i))/sysqq
                yfni(i)=(ympr(1)*yfi(i)-ympi(1)*yfr(i))/sysqq
 25         continue

c     REC&ISI subroutine which generate the transmitted symbol and calculate ISI and received
c     signal Ri, Rr.
                CALL RECISI(N1,VFR,VFI)

c     Detector subroutine  which gives the detected data symbols, and calculates the error of
c     detected signal.
                CALL DETECTOR(VFR,VFI,ICOUNT,NWRM)

c     Estimater subroutine which gives the estimated sample impulse response of the channel.
                CALL ESTIMATOR(ICOUNT)

 40    CONTINUE
                CALL FRESULT()

       REWIND(indat)
 50    CONTINUE
 55    CONTINUE

       write(llout,5)nsam,nwrm,iq,NCH
  5    format(' nsam=',i10/' nwrm=',i10/' iq=',i10/'NCH=',i3)
       write(llout,7)kv1,kv2,kv3,kvtot,kv
  7    format('kv1=',i2/'kv2=',i2/'kv3 =',i2/'kvtot =',i2/'kv=',i2/)
       write(llout,9)icount,count
  9    format(' icount=',i10/' count=',f12.2)
       write(llout,11)irpts,jwrm
 11    format(' irpts=',i10/'jwrm =',i10/)
       write(llout,13)iosrk,idec,iest,ifgm
 13    format('iosrk=',L/'idec=',L/'iest=',L/'ifgm=',L)

    STOP
    END
```

```
      SUBROUTINE INIPAR()
IMPLICIT DOUBLE PRECISION (A-H,O-Z)
PARAMETER (kv=16,nsir=20,ntap=50,n=81,nn=32,ninrt=9,nexp=4)
LOGICAL iosrk,idec,iest,ifgm,ipred
COMMON/lkupt/itb1(32),itb2(32),itb3(32),itlp1(16,2),itlp2(16,2)
COMMON/rfnois/hfr(nsir),hfi(nsir),wfr(nsir),wfi(nsir)
COMMON/clkup/itlp3(16,2),indsr(nexp),indsi(nexp)
COMMON/adf/yfr(ntap+1),yfi(ntap+1),spfr(ninrt),spfi(ninrt)
COMMON/parm/nsam,icount,itrain,jwrm,irpts,n1,nn1,iq,NCH
COMMON/pardet/idec
COMMON/lest/iosrk,iest,ifgm,ipred
COMMON/paradf/ntapp1,nsirp1,nsirp2,nfitr,fc,fd,fad,faltha
COMMON/parest/omega,qq,hqq,eata,theta,theta1,theta2,alfa,delta,bc
COMMON/rslest/snr,nwrm,count,tprer,yeror
COMMON/funit/llout,indat

      indat=10
      llout=59
c     Values for variables and arrays.
c     Fixed LMS algorithm with degree-0, FLMS:D-0.
c       ifgm=.true.
c       iest=.false.
c       ipred=.false.
c       iosrk=.false.
c     Fixed LMS algorithm with degree-0, FLMS:D-1.
c       ifgm=.true.
c       ipred=.true.
c       iest=.false.
c       iosrk=.false.
c     Adaptive LMS  algorithm with degree-0, ALMS:D-0.
c       iest=.true.
c       ipred=.false.
c       iosrk=.false.
c       ifgm=.false.
c     Adaptive LMS  algorithm with degree-1, ALMS:D-1.
        iest=.true.
        ipred=.true.
        iosrk=.false.
        ifgm=.false.
c     SRK algorithm Degree-0.
c       iosrk=.true.
c       ifgm=.true.
c       iest=.false.
c       ipred=.false.
c     SRK algorithm Degree-1.
c       iosrk=.true.
c       ifgm=.false.
c       iest=.false.
c       ipred=.false.
c     For original detector.
        idec=.true.
c       For revese detector.
c       idec=.false.
        iq=75
        NCH=3
        nsam=58640
        nwrm=1040
        itrain=960
        train=real(itrain)
        rpts=nsam/train
```

```
      irpts=int(rpts)
      nn1=nn-1
      jwrm=nn1*(irpts-1)
      n1=n-1
      snr=25.0
c      stdvn=10.0**(-snr/20.0)
c     Estimator parameter,  for system1 (FLMS D-0)
c       BC=0.024
c     For system2 (FLMS D-1)
c       BC=0.24
c       theta=0.956
c     For system3 (ALMS D-0)
c       BC=0.024
c     For system4 (ALMS D-1)
        BC=1.0
        theta=0.976
c     For system5 (SRK D-0)
c       omega=0.925
c     For system6 (SRK D-1)
        omega=0.920
        theta1=0.005
        theta2=0.99
        eata=0.0
        delta=0.003
        hqq=1.0/omega
        qq=hqq-1
c     Values of variables for adaptive filter.
        fc=1.0
        fd=10.0e-10
        fad=1.05
        faltha=1.0/fad
        nfitr=40
        nsirp1=nsir+1
        nsirp2=nsir-2
        ntapp1=ntap+1

c     Starting point for adaptive filter.

      DATA spfr /0.00000,0.90909,0.00000,0.00000,-0.90909,
     1     0.64282,0.64282,-0.64282,-0.64282/
      DATA spfi /0.00000,0.00000,-0.90909,0.90909,0.00000,
     1     -0.64282,0.64282,0.64282,-0.64282/
      DATA WFR /-0.02805,-0.23081,-0.50758,-0.49800,
     1   -0.16221,0.11289,0.10851,-0.00731,
     1   -0.04869,-0.00976,0.01514,0.00523,
     1   -0.00449,0.00063,0.00107,-0.00093,
     1   0.00000,0.00000,0.00000,0.00000/
      DATA WFI /0.01968,0.16746,0.39481,0.40568,
     1   0.10503,-0.13375,-0.07360,0.04669,
     1   0.02741,-0.01851,-0.00698,0.01100,
     1   0.00286,-0.00221,0.00136,-0.00045,
     1    0.00000,0.00000,0.00000,0.00000/
      DATA itb1/0,0,0,1,1,0,1,1,0,1,1,1,0,0,1,0,
     1     1,0,0,0,1,1,0,1,1,1,1,0,0,1,0,0/
      DATA itb2/1,1,-1,1,1,-1,-1,-1,-1,1,-1,-1,1,1,1,-1,
     1     1,-1,1,1,-1,-1,-1,1,-1,-1,1,-1,-1,1,1,1/
      DATA itb3/0,0,0,1,1,0,1,1,1,0,0,0,1,1,0,1,
     1     0,1,1,1,0,0,1,0,1,1,1,0,0,1,0,0/
```

```
c     4-QPSK constellation
      DATA (indsr(j),j=1,4)/1,-1,-1,1/
      DATA (indsi(j),j=1,4)/1,1,-1,-1/

      OPEN(indat,FILE='nmphaser1',FORM='UNFORMATTED')
      OPEN(17,FILE='ercsys',FORM='FORMATTED')
      OPEN(llout,FILE='ocsys',IOSTAT=JOSS)
      WRITE(LLOUT,72)IOSS,JOSS
   72 FORMAT(/' ERRORS IN THE EXEC OF THE OPEN STAT=',I3,I3/' ')
      WRITE(llout,73)
   73 FORMAT('Performance of combined detector, estimator and adaptive filter')
      WRITE(llout,*)
      klk1=0
      do 79 i=1,16
       do 77 j=1,2
          itlp1(i,j)=itb1(klk1+j)
          itlp2(i,j)=itb2(klk1+j)
          itlp3(i,j)=itb3(klk1+j)
   77     continue
      klk1=klk1+2
   79 continue
      RETURN
      END


            SUBROUTINE INITIL()
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      PARAMETER (kv=16,nsir=20,ntap=50,n=81,nn=32,ninrt=9)
      COMMON/dat/iar(n),iai(n),ib1(n),ib2(n),idb1(n),idb2(n)
      COMMON/symb/ar(n),ai(n)
      COMMON/rfnois/hfr(nsir),hfi(nsir),wfr(nsir),wfi(nsir)
      COMMON/main1/rr(ntap),ri(ntap),ydi(ntap,nsir),ydr(ntap,nsir)
      COMMON/mpadf/yr(nsir),yi(nsir),ympr(nsir),ympi(nsir)
      COMMON/srk/uur(nsir*nsir),uui(nsir*nsir),dd(nsir),bb(nsir)
      COMMON/estm/yestr(nsir),yesti(nsir),ycr(nsir),yci(nsir),b(nsir)
      COMMON/det/cost(kv),ixr(kv,nn),ixi(kv,nn),xr(kv,nn),xi(kv,nn)
      COMMON/result/iersym,ier1,ier2,ider1,ider2,ipdb1,ipdb2
      COMMON/rec/rrr,www,xx,stdvn
      COMMON/parm/nsam,icount,itrain,jwrm,irpts,n1,nn1,iq,NCH
      COMMON/rslest/snr,nwrm,count,tprer,yeror
      COMMON/fpr/jfroot

      xx=0.0
      jfroot=0
       icount=0
      count=0.0
      rrr=0.0
        www=0.0
        tprer=0.0
        yeror=0.0
        iersym=0
        ier1=0
        ier2=0
        ider1=0
        ider2=0
        ipdb1=0
        ipdb2=0
c     Initialising signal and noise arrays.
        do 110 i=1,n
```

```
            ib1(i)=0
            ib2(i)=0
            idb1(i)=0
            idb2(i)=0
            ar(i)=1.0
            ai(i)=1.0
            iar(i)=1
            iai(i)=1
110     continue
        do 115 i=1,ntap
            rr(i)=1.0
            ri(i)=1.0
115     continue
        do 120 i=1,nsir
            hfr(i)=0.0
            hfi(i)=0.0
 120    continue
c       Initialising stored vectors and costs.
        do 125 i=1,kv
            cost(i)=10000.0
            do 123 j=1,nn
              xr(i,j)=1.0
              xi(i,j)=1.0
              ixr(i,j)=1
              ixi(i,j)=1
 123        continue
 125    continue
        cost(1)=0.0
c       Initialising arrays where SIR's will be held.
        do 133 i=1,ntap
            do 130 j=1,nsir
              ydr(i,j)=0.0
              ydi(i,j)=0.0
 130        continue
        ydr(i,1)=1.0
        ydr(i,2)=-1.0
        ydi(i,1)=0.0
 133    continue
c       Initialising the arrays for SRK algorithm.
        is=(nsir-1)*(nsir-1)
        do 140 j=1,is/2
            uur(j)=0.0
            uui(j)=0.0
 140    continue
        do 143 j=1,(nsir-1)
            dd(j)=1.0
 143    continue
c       Initialising the arrays where predictions are held.
        do 145 i=1,nsir
            yestr(i)=0.0
            yesti(i)=0.0
            ycr(i)=0.0
            yci(i)=0.0
            bb(i)=0.0
            b(i)=0.0
145     continue
        yestr(1)=1.0
        yestr(2)=-0.1
        RETURN
        END
```

```
      SUBROUTINE TRAINSH()

IMPLICIT DOUBLE PRECISION (A-H,O-Z)
PARAMETER (kv=16,nsir=20,ntap=50,n=81,nn=32)
COMMON/dat/iar(n),iai(n),ib1(n),ib2(n),idb1(n),idb2(n)
COMMON/symb/ar(n),ai(n)
COMMON/main1/rr(ntap),ri(ntap),ydi(ntap,nsir),ydr(ntap,nsir)
COMMON/mpadf/yr(nsir),yi(nsir),ympr(nsir),ympi(nsir)
COMMON/srk/uur(nsir*nsir),uui(nsir*nsir),dd(nsir),bb(nsir)
COMMON/estm/yestr(nsir),yesti(nsir),ycr(nsir),yci(nsir),b(nsir)
COMMON/det/cost(kv),ixr(kv,nn),ixi(kv,nn),xr(kv,nn),xi(kv,nn)
COMMON/parm/nsam,icount,itrain,jwrm,irpts,n1,nn1,iq,NCH
COMMON/rslest/snr,nwrm,count,tprer,yeror
COMMON/funit/llout,indat

c     Shuffling the signal generators for incoming data signal.
      do 210 i=1,n1
         jx1=i+1
         ib1(i)=ib1(jx1)
         ib2(i)=ib2(jx1)
         idb1(i)=idb1(jx1)
         idb2(i)=idb2(jx1)
         ar(i)=ar(jx1)
         ai(i)=ai(jx1)
         iar(i)=iar(jx1)
         iai(i)=iai(jx1)
 210  continue
      do 215 i=1,(ntap-1)
         jj1=i+1
         rr(i)=rr(jj1)
         ri(i)=ri(jj1)
 215  continue
c     Shifting stored vectors to accommodate new values.
      do 225 i=1,kv
         do 220 j=1,nn1
           jx2=j+1
           xr(i,j)=xr(i,jx2)
           xi(i,j)=xi(i,jx2)
           ixr(i,j)=ixr(i,jx2)
           ixi(i,j)=ixi(i,jx2)
 220     continue
 225  continue
c     Shifting SIR array in preparation for current SIR.
      do 235 i=1,(ntap-1)
        do 230 j=1,nsir
           jj2=i+1
           ydr(i,j)=ydr(jj2,j)
           ydi(i,j)=ydi(jj2,j)
 230    continue
 235  continue

c     Input sampled impulse responses from 2-sky wave HF channel.
      READ(indat)(ydr(NTAP,I),I=1,nsir)
      READ(indat)(ydi(NTAP,I),I=1,nsir)
      IF (ICOUNT.EQ.NTAP.OR.MOD(ICOUNT,ITRAIN).EQ.0)THEN
c     IF (ICOUNT.EQ.NTAP)THEN
c     IF (ICOUNT .EQ. 1)THEN
      do 240 i=1,nsir
        yestr(i)=ydr(1,i)
        yesti(i)=ydi(1,i)
```

```
c            yestr(i)=0.0
c            yesti(i)=0.0
            ycr(i)=0.0
             yci(i)=0.0
 240   continue
       iss=(nsir-1)*(nsir-1)
       do 245 j=1,iss/2
            uur(j)=0.0
            uui(j)=0.0
 245   continue
       do 247 j=1,(nsir-1)
            dd(j)=1.0
 247   continue
       do 255 i=1,kv
          do 250 j=1,nn1
                xr(i,j)=ar(j)
                xi(i,j)=ai(j)
               ixr(i,j)=iar(j)
               ixi(i,j)=iai(j)
 250        continue
       cost(i)=10000.0
 255   continue
       cost(1)=0.0
       endif
       do 260 i=1,nsir
          YR(I)=yestr(I)
          YI(I)=yesti(I)
 260   continue
c      Calculation of prediction errors.
       if (icount.gt.nwrm) then
        yeror=0.0
       do 270 i=1,nsir
          PRER1=(yestr(I)-ydr(1,I))**2
          PRER2=(yesti(I)-ydi(1,I))**2
          yeror=yeror+prer1+prer2
270       continue
      tprer=tprer+yeror
      endif
      RETURN
      END


         SUBROUTINE RECISI(n1,vfr,vfi)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      PARAMETER (kv=16,nsir=20,ntap=50,n=81,nn=32,nexp=4)
       DOUBLE PRECISION G05DDF
      COMMON/dat/iar(n),iai(n),ib1(n),ib2(n),idb1(n),idb2(n)
      COMMON/symb/ar(n),ai(n)
      COMMON/lkupt/itb1(32),itb2(32),itb3(32),itlp1(16,2),itlp2(16,2)
      COMMON/clkup/itlp3(16,2),indsr(nexp),indsi(nexp)
      COMMON/rfnois/hfr(nsir),hfi(nsir),wfr(nsir),wfi(nsir)
      COMMON/mpnor/yfnr(ntap),yfni(ntap),ympnr(nsir),ympni(nsir)
      COMMON/main1/rr(ntap),ri(ntap),ydi(ntap,nsir),ydr(ntap,nsir)
      COMMON/rec/rrr,www,xx,stdvn
      COMMON/isi/gr(kv),gi(kv)
      COMMON/det/cost(kv),ixr(kv,nn),ixi(kv,nn),xr(kv,nn),xi(kv,nn)
      COMMON/rslest/snr,nwrm,count,tprer,yeror

c     Generation signal in bit form.
       xx=g05caf(xx)
```

```
      if(xx-0.5)640,640,645
640       ib1(n)=0
          go to 650
645       ib1(n)=1
650       xx=g05caf(xx)
          if(xx-0.5)655,655,660
655         ib2(n)=0
            go to 665
660         ib2(n)=1
665       continue
c     Differential encoding and generation of QPSK signal.
        jen1=idb1(n1)*8+idb2(n1)*4+ib1(n)*2+ib2(n)+1
          idb1(n)=itlp1(jen1,1)
          idb2(n)=itlp1(jen1,2)
          iar(N)=itlp2(JEN1,1)
          iai(N)=itlp2(JEN1,2)
          ar(n)=real(iar(n))
          ai(n)=real(iai(n))
c     Noise Generation.
        do 680 lnm=1,2
            do 670 I=1,nsir-1
                    JX4=I+1
              HFR(I)=HFR(JX4)
              HFI(I)=HFI(JX4)
670           continue
            HFR(nsir)=G05DDF(0.0d0,STDVN)
             HFI(nsir)=G05DDF(0.0d0,STDVN)
            WR=0.0
            WI=0.0
            do 675 I=1,nsir
                JX5=nsir-I+1
              WR=WR+HFR(JX5)*WFR(I)-HFI(JX5)*WFI(I)
              WI=WI+HFR(JX5)*WFI(I)-HFI(JX5)*WFR(I)
675         continue
680     continue
c     Uncorrelated noise
c       WR=G05DDF(0.0d0,STDVN)
c       WI=G05DDF(0.0d0,STDVN)
        www=www+wr*wr+wi*wi
c     Calculating inter-symbol  interference.
        do 690 i=1,kv
          gr(i)=0.0
          gi(i)=0.0
           do 685 j=2,nsir
                jx3=nn+1-j
            gr(i)=gr(i)+xr(i,jx3)*ympnr(j)-xi(i,jx3)*ympni(j)
            gi(i)=gi(i)+xr(i,jx3)*ympni(j)+xi(i,jx3)*ympnr(j)
685       continue
690     continue
c     Calculation of received signal.
        rr(ntap)=0.0
        ri(ntap)=0.0
        do 700 i=1,nsir
              jx6=n+1-i
          rr(ntap)=rr(ntap)+ar(jx6)*ydr(ntap,i)-ai(jx6)*ydi(ntap,i)
          ri(ntap)=ri(ntap)+ar(jx6)*ydi(ntap,i)+ai(jx6)*ydr(ntap,i)
700     continue
c       print *,rr(ntap),ri(ntap),wr,wi
c     Receive signal with noise
        rr(ntap)=rr(ntap)+wr
```

```
      ri(ntap)=ri(ntap)+wi
        rrr=rrr+rr(ntap)*rr(ntap)+ri(ntap)*ri(ntap)

c     Passing received signal through the adaptive filter.
      vfr=0.0
      vfi=0.0
      do 705 i=1,ntap
            jj3=ntap-i+1
        vfr=vfr+yfnr(i)*rr(jj3)-yfni(i)*ri(jj3)
        vfi=vfi+yfnr(i)*ri(jj3)+yfni(i)*rr(jj3)
705   continue
      RETURN
      END
```

```
c           Detection process begins.
            SUBROUTINE DETECTOR(vfr,vfi,icount,nwrm)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      PARAMETER (kv=16,nsir=20,ntap=50,n=81,nn=32,nexp=4)
      PARAMETER (kv1=2,kv2=4,kv3=kv-kv1-kv2,kvtot=4*kv1+2*kv2+kv3)
      INTEGER IXTR(KVTOT,NN),IXTI(KVTOT,NN),INN(KV)
      DOUBLE PRECISION vr(kv),vi(kv),vvr(kv),vvi(kv)
      DOUBLE PRECISION ct1(kvtot),ct2(kv),ct3(kv)
      LOGICAL idec
      COMMON/dat/iar(n),iai(n),ib1(n),ib2(n),idb1(n),idb2(n)
      COMMON/clkup/itlp3(16,2),indsr(nexp),indsi(nexp)
      COMMON/isi/gr(kv),gi(kv)
      COMMON/det/cost(kv),ixr(kv,nn),ixi(kv,nn),xr(kv,nn),xi(kv,nn)
      COMMON/result/iersym,ier1,ier2,ider1,ider2,ipdb1,ipdb2
      COMMON/pardet/idec
```

```
      mkvtot=kvtot
      mkv1=kv1
      mkv2=kv2
      mkv3=kv3
c     Threshold detector to obtain the lowest cost.
c     Calculation of the costs of these expanded vectors.
      do 750 i=1,kv
          v1=vfr-gr(i)
          v2=vfi-gi(i)
          vr(i)=v1
          vi(i)=v2
          if (v1 .ge. 0) then
              xr(i,nn)=1.0
              ixr(i,nn)=1
          else
              xr(i,nn)=-1.0
          ixr(i,nn)=-1
      endif
      if (v2 .ge. 0) then
          xi(i,nn)=1.0
          ixi(i,nn)=1
      else
          xi(i,nn)=-1.0
          ixi(i,nn)=-1
      endif
      dr=v1-xr(i,nn)
      di=v2-xi(i,nn)
      vvr(i)=dr
      vvi(i)=di
```

```
          ct3(i)=cost(i)+dr*dr+di*di
 750   continue
c      Forming k*nexp vectors and calculation of the costs of these expanded vector.
       if (idec) then
          jn=0
       else
           jn=(2*kv2+kv3)
       end if

       do 767 jjj=1,kv1,1
          if (idec) then
            jj=jjj
          else
            jj=(jjj+kv2+kv3)
          end if
          do 765 i=1,4
            jn=jn+1
            do 760 j=1,nn
                ixtr(jn,j)=ixr(jj,j)
                ixti(jn,j)=ixi(jj,j)
 760        continue
            ixtr(jn,nn)=indsr(i)
            ixti(jn,nn)=indsi(i)
            ddr=vr(jj)-real(ixtr(jn,nn))
            ddi=vi(jj)-real(ixti(jn,nn))
            ct1(jn)=cost(jj)+ddr*ddr+ddi*ddi
 765      continue
 767   continue
       if (idec) then
          jd1=jn
       else
          jd1=0
       endif
c      jd1=jn
       do 800 II=1,kv2
          if (idec) then
              I=II+kv1
          else
              I=II
          endif
          do 773 jj=1,2
              jd1=jd1+1
              do 770 j=1,nn
                  ixtr(jd1,j)=ixr(i,j)
                  ixti(jd1,j)=ixi(i,j)
 770          continue
              ct1(jd1)=ct3(i)
 773      continue
          isxr=ixtr(jd1,nn)
          isxi=ixti(jd1,nn)
          vsr=sign(2.1,vvr(i))
          ivsr=INT(vsr)
          vsi=sign(2.1,vvi(i))
          ivsi=INT(vsi)
c         print *,vsr,vsi
          IF(ABS(vvi(i))-ABS(vvr(i)))775,775,785
 775          ixtr(jd1,nn)=ivsr+isxr
c             print *,jd2,ixtr(jd2,nn)
          IF(ABS(ixtr(jd1,nn))-2)781,779,779
 779          ixtr(jd1,nn)=isxr
```

```
c           print *,jd2,ixtr(jd2,nn)
 781      continue
          ixti(jd1,nn)=ivsi+isxi
          IF(ABS(ixti(jd1,nn))-2)784,782,782
 782         ixti(jd1,nn)=isxi-ivsi
 784      continue
          goto 795
 785      ixti(jd1,nn)=ivsi+isxi
          IF(ABS(ixti(jd1,nn))-2)795,787,787
 787         ixtr(jd1,nn)=isxr+ivsr
             ixti(jd1,nn)=isxi
          IF(ABS(ixtr(jd1,nn))-2)795,790,790
 790         ixtr(jd1,nn)=isxr-ivsr
 795         continue
             dr=vr(i)-REAL(ixtr(jd1,nn))
             di=vi(i)-REAL(ixti(jd1,nn))
             ct1(jd1)=cost(i)+dr*dr+di*di
cc           mc1=mc1+1
c            jd1=jd1+2
 800   continue
       if (kv3.eq.0) goto 810
          jd2=jd1
          do 810  j3=1,kv3
             jd2=jd2+1
             if (idec) then
                j=kv1+kv2+j3
             else
                j=j3+kv2
             endif
             do 805 i=1,nn
                ixtr(jd2,i)=ixr(j,i)
                ixti(jd2,i)=ixi(j,i)
 805         continue
             ct1(jd2)=ct3(j)
 810   continue
c                 Vector selection
c     Select the best vector from k_vtot expanded vectors.
        cc=1000000.0D0
        do 820 i=1,kvtot
           if(CT1(I).lt.cc) then
              cc=ct1(i)
               lc1=i
           endif
 820   continue
       ct2(1)=cc
       inn(1)=lc1
       ct1(lc1)=10000000.0D0
c     Obtaining bit value of detected signal.
       kad=ixtr(lc1,1)+ixti(Lc1,1)
       if(kad-0)823,828,825
 823      idbb1=1
          idbb2=1
          go to 830
 825            idbb1=0
                idbb2=0
                go to 830
 828   if(ixtr(lc1,1).eq.1)then
          idbb1=1
          idbb2=0
       else
```

```
        idbb1=0
        idbb2=1
      end if
  830 continue
c           Differential decoding operation
      jen2=ipdb1*8+ipdb2*4+idbb1*2+idbb2+1
        isb1=itlp3(jen2,1)
        isb2=itlp3(Jen2,2)
        if(icount .le. nwrm) go to 850
          if(iar(1)-ixtr(lc1,1))835,833,835
  833           if(iai(1)-ixti(lc1,1))835,840,835
  835             iersym=iersym+1
  840       if(idb1(1).ne.idbb1)ier1=ier1+1
              if(idb2(1).ne.idbb2)ier2=ier2+1
                if(ib1(1).ne.isb1)ider1=ider1+1
                  f(ib2(1).ne.isb2)ider2=ider2+1
  850 continue
c     Discarding the vectors which do not agree.
        do 860 i=1,kvtot
          if(ixtr(I,1)-ixtr(lc1,1))857,855,857
  855     if(ixti(I,1)-ixti(lc1,1))857,860,857
  857     ct1(i)=10000000.0D0
  860 continue

c     Selecting ((kv-1) lowest-cost vectors from (kv*nexp) vectors.
        do 870 i=1,(kv-1)
          cc=1000000.0d0
          do 865 j=1,kvtot
            If(Ct1(j).lt.cc) then
              cc=ct1(j)
              lc2=j
            endif
  865       continue
            ct2(i+1)=cc
            inn(i+1)=lc2
            ct1(lc2)=10000000.0D0
  870   continue
c     Converting temporary costs of selected vectors into permanent costs.
        do 875 i=1,kv
          cost(i)=ct2(i)
  875   continue
c     Transferring the final kv vectors and costs to their original store ready for the next detection
c     process.
        do 885 i=1,kv
          ifn=inn(i)
          do 880 j=1,nn
            ixr(i,j)=ixtr(ifn,j)
            ixi(i,j)=ixti(ifn,j)
            xr(i,j)=REAL(ixr(i,j))
            xi(i,j)=REAL(ixi(i,j))
  880       continue
  885   continue

c     Subtract costs to bring the smallest cost to zero.
        do 890 i=2,kv
          cost(i)=cost(i)-cost(1)
  890   continue
        cost(1)=0.0d0
        ipdb1=idbb1
        ipdb2=idbb2
```

```
RETURN
END


      SUBROUTINE ESTIMATOR(icount)
IMPLICIT DOUBLE PRECISION (A-H,O-Z)
PARAMETER (kv=16,nsir=20,ntap=50,n=81,nn=32)
DOUBLE PRECISION aa(nsir),fr(nsir),fi(nsir),ggr(nsir),ggi(nsir)
DOUBLE PRECISION eri(nsir),err(nsir)
DOUBLE PRECISION uyestr(nsir),uyesti(nsir)
LOGICAL iosrk,iest,ifgm,ipred
COMMON/main1/rr(ntap),ri(ntap),ydi(ntap,nsir),ydr(ntap,nsir)
COMMON/srk/uur(nsir*nsir),uui(nsir*nsir),dd(nsir),bb(nsir)
COMMON/estm/yestr(nsir),yesti(nsir),ycr(nsir),yci(nsir),b(nsir)
COMMON/det/cost(kv),ixr(kv,nn),ixi(kv,nn),xr(kv,nn),xi(kv,nn)
COMMON/lest/iosrk,iest,ifgm,ipred
COMMON/parest/omega,qq,hqq,eata,theta,theta1,theta2,alfa,delta,bc

c     Forming the estimate of the received signal and calculating  the error in the estimation.
      dum1=(1.0-theta)*(1.0-theta)
      dum2=1.0-(theta*theta)
      rer=0.0
      rei=0.0
      do 1010 i=1,nsir
         kx1=nn+1-i
         rer=rer+xr(1,kx1)*yestr(i)-xi(1,kx1)*yesti(i)
         rei=rei+xr(1,kx1)*yesti(i)+xi(1,kx1)*yestr(i)
1010  continue
      erear=rr(1)-rer
      ereai=ri(1)-rei
      if (iest) then
        do 1015 j=1,nsir
          yvar=(yestr(j)*yestr(j)+yesti(j)*yesti(j))**0.5
          bb(j)=yvar
1015    continue
        do 1020 j=1,nsir
           b(j)=bb(j)**0.5
           if (b(j).lt.delta) then
              b(j)=0.000001
           endif
1020    continue
      endif
      if (iosrk) then
c     Updating channel using SRK algorithm.
      fr(1)=xr(1,nn)
      fi(1)=-xi(1,nn)
      ggr(1)=dd(1)*fr(1)
      ggi(1)=dd(1)*fi(1)
      zzr=ggr(1)*fr(1)+ggi(1)*fi(1)
      zzi=ggi(1)*fr(1)-ggr(1)*fi(1)
      aa(1)=eata+zzr
      nnn=0
      do 1030  j=2,nsir
         IX=nn+1-J
         fr(j)=xr(1,ix)
         fi(j)=-xi(1,ix)
         do 1025 i=1,(j-1)
            nnn=nnn+1
            fr(j)=fr(j)+(uur(nnn)*xr(1,nn+1-i)+uui(nnn)*xi(1,nn+1-i))
            fi(j)=fi(j)+(uui(nnn)*xr(1,nn+1-i)-uur(nnn)*xi(1,nn+1-i))
```

```
1025      continue
          ggr(j)=dd(j)*fr(j)
          ggi(j)=dd(j)*fi(j)
          aa(j)=aa(j-1)+(ggr(j)*fr(j)+ggi(j)*fi(j))
1030  continue
      ht=0.0
      Pkv=1.0/(aa(1)+ht)
      dd(1)=dd(1)*hqq*(eata+ht)*Pkv
      nnn=0
      do 1040 j=2,nsir
         beta=aa(j-1)+ht
         plamdar=fr(j)*Pkv
         plamdai=fi(j)*Pkv
         Pkv=1.0/(aa(j)+ht)
         dd(j)=dd(j)*hqq*beta*Pkv
         do 1035 i=1,(j-1)
            nnn=nnn+1
            beta1r=uur(nnn)
            beta1i=uui(nnn)
            uur(nnn)=beta1r-(ggr(i)*plamdar+ggi(i)*plamdai)
            uui(nnn)=beta1i-(ggr(i)*plamdai-ggi(i)*plamdar)
            ggr(i)=ggr(i)+(ggr(j)*beta1r+ggi(j)*beta1i)
            ggi(i)=ggi(i)+(ggi(j)*beta1r-ggr(j)*beta1i)
1035     continue
1040  continue
      beta1r=erear/aa(nsir)
      beta1i=ereai/aa(nsir)
      if (iosrk.and.ifgm) then
c     SRK:D-0
         do 1045 j=1,nsir
            yestr(j)=yestr(j)+(ggr(j)*beta1r-ggi(j)*beta1i)
            yesti(j)=yesti(j)+(ggi(j)*beta1r+ggr(j)*beta1i)
1045     continue
         go to 1077
      endif
      endif
c     For adaptive channel estimater (sys6).
      if (iest) then
      do 1048 j=1,nsir
         IX6=nn+1-J
         uyestr(J)=yestr(j)+Bc*B(J)*(erear*xR(1,IX6)+ereai*xI(1,IX6))
         uyesti(J)=yesti(j)+Bc*B(J)*(ereai*xR(1,IX6)-erear*xI(1,IX6))
1048  continue
c     For adaptive SRK algorithm.
      elseif(iosrk) then
         do 1050 j=1,nsir
            uyestr(j)=yestr(j)+(ggr(j)*beta1r-ggi(j)*beta1i)
            uyesti(j)=yesti(j)+(ggi(j)*beta1r+ggr(j)*beta1i)
1050        continue
c     For FLMS algorithm.
      elseif(ifgm) then
         do 1052 j=1,nsir
            IX6=nn+1-J
            uyestr(J)=yestr(j)+Bc*(erear*xR(1,IX6)+ereai*xI(1,IX6))
            uyesti(J)=yesti(j)+Bc*(ereai*xR(1,IX6)-erear*xI(1,IX6))
1052  continue
      endif
c     Error in updating the SIR.
      do 1060 i=1,nsir
         err(i)=uyestr(i)-yestr(i)
```

```
            eri(i)=uyesti(i)-yesti(i)
 1060  continue
c      Prediction process begins "degree-1 for gradient estimator".
        if (ipred) then
            do 1070 i=1,nsir
                ycr(i)=ycr(i)+(dum1*err(i))
                yci(i)=yci(i)+(dum1*eri(i))
                yestr(i)=yestr(i)+ycr(i)+(dum2*err(i))
                yesti(i)=yesti(i)+yci(i)+(dum2*eri(i))
 1070  continue
c      Prediction process begins "degree-1 for SRK estimator".
        elseif (iosrk) then
            do 1073 i=1,nsir
                ycr(i)=ycr(i)+(theta1*err(i))
                yci(i)=yci(i)+(theta1*eri(i))
                yestr(i)=yestr(i)+ycr(i)+(theta2*err(i))
                yesti(i)=yesti(i)+yci(i)+(theta2*eri(i))
 1073       continue
        else
c      Prediction using L.M.S. 'Gradient algorithm' with degree-0.
        do 1075 i=1,nsir
            yestr(i)=yestr(i)+err(i)
            yesti(i)=yesti(i)+eri(i)
 1075  continue
        endif
 1077  continue
        RETURN
        END


              SUBROUTINE FRESULT()
        IMPLICIT DOUBLE PRECISION (A-H,O-Z)
c       PARAMETER (kv=16,nsir=20,ntap=50,n=81,nn=32,ninrt=9)
        COMMON/result/iersym,ier1,ier2,ider1,ider2,ipdb1,ipdb2
        COMMON/rec/rrr,www,xx,stdvn
        COMMON/parm/nsam,icount,itrain,jwrm,irpts,n1,nn1,iq,NCH
        COMMON/paradf/ntapp1,nsirp1,nsirp2,ndivg,c,d,ad,althd
        COMMON/parest/omega,qq,hqq,eata,theta,theta1,theta2,alfa,delta,bc
        COMMON/rslest/snr,nwrm,count,tprer,yeror
        COMMON/funit/llout,indat

c       Confirmation of SNR value and calculation of the error rates.
        snrc=10.0*log10(rrr/www)
        tsnr=10.0*log10(1.0/(stdvn*stdvn))
        ersym=iersym/(count-nwrm-jwrm)
        erbit=(ier1+ier2)/(2.0*(count-nwrm-jwrm))
        erdbit=(ider1+ider2)/(2.0*(count-nwrm-jwrm))
        if (tprer.ne.0)then
            avprer=10.0*log10(tprer/(count-nwrm))
        else
            avprer=0.0
        endif
c       Printing useful results.
        write(llout,1200)bc,alfa,delta,theta
 1200  format(' bc=',f8.4/'alfa=',f8.4",f10.6/'theta=',f8.4/)
        write(llout,1205)eata,omega,theta1,theta2
 1205  format('eata=',f8.4/'W=',f8.4/'O1=',f10.6/'O2=',f10.6)
        write(llout,1210)tsnr,snrc
 1210  format('theoretical snr ratio=',f9.3,'db','snrc=',f9.3/)
        write(llout,1215)avprer
```

```
1215  format(' average error in one step prediction=',f20.7,'db')
      write(llout,1220)tprer
1220  format('error in one step prediction=',f20.4/)
      write(llout,1225)ersym
1225  format(' symbol error rate   =',e20.10)
      write(llout,1230)erbit
1230  format(' bit error rate      =',e20.10)
      write(llout,1235)erdbit
1235  format(' differential bit error rate =',e20.10)
      write(17,1240)theta,avprer,tprer,snrc,erbit
1240  format(f8.4,2x,e18.10,2x,f18.4,1x,f8.3,1x,e18.10)
      RETURN
      END
```

```
c     Subroutine for adaptive filter and root finding algorithm.
          SUBROUTINE ADFILT(icount)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      PARAMETER (nsir=20,ntap=50,ninrt=9)
      DOUBLE PRECISION fer(nsir+1),fei(nsir+1)
      DOUBLE PRECISION afr(nsir+1),afi(nsir+1)
      DOUBLE PRECISION flreal(nsir),flimg(nsir)
      DOUBLE PRECISION azr(ntap+1),azi(ntap+1)
c     DOUBLE PRECISION fpreal(nsir),fpimg(nsir)
      COMMON/mpadf/yr(nsir),yi(nsir),ympr(nsir),ympi(nsir)
      COMMON/adf/yfr(ntap+1),yfi(ntap+1),spfr(ninrt),spfi(ninrt)
      COMMON/adfsplr/splr(nsir),spli(nsir)
      COMMON/fpr/jfroot
      COMMON/paradf/ntapp1,nsirp1,nsirp2,nfitr,fc,fd,fad,faltha
      nstrpf=1
      nroot=0
      ispfr=int(spfr(ninrt)*100.0)
      ispfi=int(spfi(ninrt)*100.0)
      if(icount .eq. 1) then
          do 320 i=1,ninrt
              splr(i)=spfr(i)
              spli(i)=spfi(i)
320       continue
          do 325 i=ninrt+1,nsir
                splr(i)=0.0
                spli(i)=0.0
325       continue
      endif
      do 330 i=1,nsir
          ympr(i)=yr(i)
          ympi(i)=yi(i)
          flreal(i)=0.0
              flimg(i)=0.0
 330  continue
 350  continue
c     One tap feedback filter.
        fcon1r=splr(nstrpf)
        fcon1i=spli(nstrpf)
        do 390 i=1,nfitr
              do 360 j=1,nsirp1
                  fer(j)=0.0
                  fei(j)=0.0
 360          continue
              fcon2r=0.0
```

```
                fcon2i=0.0
                do 370 j=1,nsir
                    j1=nsir+1-j
                    fxr=ympr(j1)-(fcon1r*fcon2r-fcon1i*fcon2i)
                    fxi=ympi(j1)-(fcon1r*fcon2i+fcon1i*fcon2r)
                    fcon2r=fxr
                    fcon2i=fxi
                    fer(j1)=fxr
                    fei(j1)=fxi
c                   PRINT *,'fer=',fer(J1),'fei=',fei(J1)
 370            continue
c       Estimate value of epsilon.
                fcon3r=-fcon1r
                fcon3i=-fcon1i
                fepsr=fer(nsir)
                fepsi=fei(nsir)
c               PRINT *,'FEPSR=',FEPSR,'FEPSI=',FEPSI
                do 375 j=1,nsirp2
                      j2=nsir-j
                      pfcon3r=fepsr*fcon3r-fepsi*fcon3i
                      pfcon3i=fepsr*fcon3i+fepsi*fcon3r
                      fepsr=pfcon3r+fer(j2)
                      fepsi=pfcon3i+fei(j2)
 375            continue
c       Obtain conditions for convergence.
                fepsam=fepsr*fepsr+fepsi*fepsi
c               PRINT *,FEPSAM
                fdelr=(fer(1)*fepsr+fei(1)*fepsi)/fepsam
                fdeli=(fei(1)*fepsr-fer(1)*fepsi)/fepsam
                fdelam=fdelr*fdelr+fdeli*fdeli
c       As long as convergence is not achieved or algorithm has not diverged, repeat the iteration.
                if (fdelam.lt.fd)then
                      go to 405
                else
                      fdelr=fc*fdelr
                      fdeli=fc*fdeli
                      fcon1r=fcon1r+fdelr
                      fcon1i=fcon1i+fdeli
                      fdl1=sqrt(fcon1r*fcon1r+fcon1i*fcon1i)
                      if (fdl1.gt.faltha) then
                          go to 400
                      endif
                endif
 390    continue
 400    continue
c       find new starting point after divergence.
        if(icount .eq. 1)then
            if(nstrpf.lt.ninrt)then
                nstrpf=nstrpf+1
                go to 350
        else
            go to 450
        endif
    else
          isplr=int(splr(nstrpf)*100.0)
          ispli=int(spli(nstrpf)*100.0)
        if(isplr.eq.ispfr.and.ispli.eq.ispfi)then
            go to 450
        else
            nstrpf=nstrpf+1
```

```
                go to 350
            endif
        endif
405     continue
c       Alter channel with obtained root by passing through  two tap feed-forward filter.
        nroot=nroot+1
        flreal(nroot)=fcon1r
        flimg(nroot)=fcon1i
        afr(1)=fer(1)
        afi(1)=fei(1)
        do 410 j=2,nsirp1
            j3=j-1
            afr(j)=fer(j)+fer(j3)*fcon1r+fei(j3)*fcon1i
            afi(j)=fei(j)+fei(j3)*fcon1r-fer(j3)*fcon1i
            ympr(j3)=afr(j)
            ympi(j3)=afi(j)
410     continue
c       find new starting point after convergence.
        if(icount.eq.1)then
            if(nstrpf.lt.ninrt)then
                nstrpf=nstrpf+1
                go to 350
            else
                go to 450
            endif
        else
            if(nstrpf.le.jfroot)then
                nstrpf=nstrpf+1
                go to 350
            else
c               go to 350
                go to 450
            endif
        endif
450     continue
c       Setting up starting points for next time instant when all roots have been found.
        ntot1=nroot+ninrt+1
        jfroot=nroot
        do 470 j=1,nroot
            splr(j)=flreal(j)
            spli(j)=flimg(j)
470     continue
        do 480 j=1,ninrt
            splr(nroot+j)=spfr(j)
            spli(nroot+j)=spfi(j)
480     continue
        if (ntot1.le.nsir) then
            do 490 j=ntot1,nsir
                splr(j)=0.0
                spli(j)=0.0
490         continue
        else
            go to 500
        endif
500     continue

cc      do 510 j=1,nroot
c               fetar=flreal(j)
c       fetai=flimg(j)
c       rt=fetar*fetar+fetai*fetai
```

REFERENCES

1. Proakis, J.G. "Digital communications", McGraw-Hill, 1989.
2. Picquenard, A. "Radio Wave Propagation", Macmillan Press, 1974.
3. Stein, S. "Fading Channel Issues in System Engineering" , IEEE Journal on Selected Areas in Communications, Vol.SAC-5, No. 2, pp. 86-89, Feb., 1987.
4. Goldberg, B. "300KHz-30MHz MF/HF", IEEE Trans. on Communication Technology, Vol. COM-14, No.6, pp. 767-784, Dec. 1966.
5. Price, R. "A Communication Technique for Multipath Channels", Proceeding of the IRE, pp. 555-570, March, 1958.
6. Monsen, P. "Fading Channel Communication", IEEE Communications Magazine, pp.16-25, 1980.
7. CCIR "Ionospheric Properties", XVth Plenary Assembly CCIR, Vol. VI, Report 7251, pp. 1-15 , 1982.
8. CCIR, "Multi-path Propagation on HF Radio circuits ", Xth Plenary Assembly CCIR, Vol. VI, Report 203 , pp. 49-51 , 1963.
9. CCIR , "Ionospheric Propagation Characteristics Pertinent to Radio Communication System Design-(Fading)", XIIIth Plenary Assembly CCIR , Vol. VI , Report 266-3, pp. 207-217, 1974.
10. Ralphs, J.D. and Sladen , F.M.E. "An HF Channel Simulator using a new Rayleigh Fading Method", The Radio and Electronic Engineerr, Vol.46, No.12, pp.579-587, Dec.1976.
11. CCIR, "HF Ionospheric Channel Simulators", XIIIth Plenary Assembly CCIR, Vol. III, Report 549-1, pp. 47-52, 66-75, 1974.
12. Matley, W. and Bywater, R.E.H. "A Digital High-Frequency Multi-path Propagation Simulator", The Radio and Electronic Engineer, Vol.47, No.7, pp. 305-314, 1977.
13. Ehraman, L., Bates, L.B. and Kates, J.M. "Real-Time Software Simulation of the HF Radio Channel", IEEE , Trans. on Comm. Vol.COM-30, No.8, pp. 1809-1816, 1982.
14. Roger L., Freeman, "Telecommunication Transmission Handbook", Third-Edition, 1991, J. Wiley and Sons.
15. Clark, A.P. "Principles of Digital Data Transmission", 1983, Pentech Press.
16. Watterson, G.C., Juroshek, J.R. and Beusema, W.D. "Experimental Confirmation of an HF Channel Model", IEEE Trans. on Comm. Technology, Vol.COM-18, No.6, pp. 792-803, Dec. 1970.
17. Najdi, H.Y. "Digital Data Transmission Over Voice Channel", Ph.D thesis, Lough. University 1982.

18. Kuo, F.F. "Network Analysis and Synthesis", John Wiley, 1966.
19. Rabiner, L.K. and Gold, B."Theory and Application of Digital Signal Processing", Prentic Hall, 1975.
20. Stein, S. and Johns, J. J. "Modern Communication Principles with Application to Digital Signalling", 1967.
21. Lindsey, W. C. and Simon, M. K. "Carrier Synhronisation and Detection of Poly-phase Signals", IEEE, Trans. on commun., COM-20, pp. 441-454, 1972.
22. Harvey, J. D. "Synchronisation of a Synchronous Modem", science research council Report, No. GR/A/1200.7, Loughborough university, 1980.
23. Clark, A. P. "Adaptive Detectors for Digital Modems", Pentech Press, 1989.
24. Fairfield, M. J. "The Chosen Equipment Filter (410)", Internal Report, Dept. of Electronic and Electrical Eng., Loughborough university, 1978.
25. Clark, A. P. and Najdi, H. Y. "Detection Process of a 9600 bits/sec Serial Modem for HF Radio Links", IEE Proc. Part F, Vol.-130, pp. 368-376, Aug. 1983.
26. Abdullah, S. N., "Data Transmission at 9600 bit/sec over an HF Radio Link", Ph.D., Thesis, Loughborough university, 1986.
27. Harun, R. "Techniques of Channel Estimation for HF Radio Links", Ph.D., Thesis, Loughborough university, 1984.
28. Hariharan, S. "Channel Estinmators for HF Radio Links", Ph.D., Thesis, Loughborough university, 1988.
29. Clark, A. P. "Advanced Data Transmission Systems ", Pentech Press, 1977.
30.Clark, A. P., Najdi, H. Y. and MacVerry, F. "Performance of a 9600 bit/s Serial Modem over a Model of an HF Radio Link", IEE Conf. Publication No.224, Radio Spectrum Conservation Techniques, pp. 151-155, Sept., 1983.
31. Clark, A. P., and MacVerry, F. "Performance of 2400bit/s Serial and Parallel Modems over an HF Channel Simulator", IERE Conf. on Digital Signal processing of Signals in Communications, Proc. No.49, pp. 167-179, Lough., 1981.
32. Bateman, S. C., Clark, A. P. and Ameen, S. Y. "Adjustment of Predetection Filters in high-speed Data Transmission Systems", IEE Proc. Pt. I, No.6, Vol.137, pp.355-364, Dec., 1990.
33. Oppenheim, A. V. and Shafer, R. W. "Digital Signal Processing", Prentic-Hall, 1975.
34. Marlin, P. R. "Alternative in Digital Communications", Proc. IEEE, Vol.61, No.6, pp.703-721, June 1973.
35. Jayasinghe, S. G. "Techniques of Detection, Estimation and Coding for Fading Channels", Ph.D., Thesis, Loughborough university, 1989.
36. Clark, A. P. "Equalisation and Detection Techniques", IEE Colloquium on a Review of Modem Techniques, Jan. 1981.

37. Forney, G. D. "Maximum Likelihood Sequence Estimation in the Presence of Inter-Symbol Interference", IEEE, Trans. on Information Theory, Vol.IT-18, pp. 363-378, May 1972.

38. Forney, G. D. "The Viterbi Algorithm", IEEE, Proc., Vol.-61, No.3, pp. 268-278 March, 1973.

39. Clark, A. P., and Hau, S. F. "Adaptive Adjustment of Receiver for Distorted Digital Signals", IEE Proc., Pt. F, Vol.131, pp. 526-536, Aug., 1984.

40. Clark, A. P. "Equalisers for Digital Modems", Pentech Press, 1985.

41. Qureshi, S. U. H. and Newhall, E. E. "An Adaptive Receiver for Data Transmission over Time Dispersive Channels", IEEE, Trans. on Information Theory, Vol.IT-19, pp.448- 457, July 1973.

42. Foschini, G. J. "A Reduced-State Variant of Maximum Likelihood Sequence Detection attaining optimum Performance for High Signal-To-Noise Ratios", IEEE, Trans.on Information Theory, Vol.IT-23, No.5, pp. 605-609, Sept., 1977.

43. Clark, A. P., Harvey, J. D. and Driscoll, J. D. "Near Maximum Likelihood Detection Process for Distorted Digital Signals", Radio and Electronic Engineer, 48, pp.301-309, June 1978.

44. Clark, A. P., and Fairfield, M. J. "Detection Processes for a 9600 bit/s Modem", Radio and Electronic Engineer, 51, pp. 455-465, 1981.

45. Clark, A. P., and Clayden, M. "Pseudo-binary Viterbi Detector", IEE Proc., Pt. F, Vol.131, No.2, pp. 208-218, April 1984.

46. Clark, A. P., and Asghar, S. M. "Detection of Digital Signals Transmitted over a Time-Varying Channel", IEE Proc., Vol.128, Pt. F, No.3, pp. 167-174, June, 1981.

47. Bic, J. C., Duponteil, D. and Imbeaus, J. C. "Elements of Digital Communication", J. Wiley and Sons, 1991.

48. Hsu, F. S. "Square Root Kalman Filtering for high Speed Data Received over Fading Dispersive HF Channels ", IEEE, Trans. on Information Theory, Vol.IT-28, No.5, pp. 753-763, Sept. 1982.

49. Monsen, P. "Feed-back Equalisation for Fading Dispersive Channels", IEEE Trans. on Information Theory, Vol.IT-17, pp. 56-64, Janu., 1971.

50. Mulgrew, B. and Cowan, C. F. N. "Adaptive Filters and Equalisers", 1988.

51. Widrow, B. and Stearns, S.D. "Adaptive signal processing", Prentice-Hall, 1985.

52. Qureshi, S. U. H. "Adaptive Equalisation", IEEE Proc., Vol.73, No.9, pp. 1349-1387, Sept. 1985.

53. Monsen, P. "Theoretical and Measured Performance of a DFE Modem on a Fading Multipath Channel", IEEE, Trans. on Commun., Vol. COM-25, pp. 1144-1153, Oct. 1977.

54. Salz, J. "Optimum Mean-Square Decision Feed-back Equalisation", BST. J., No.52, pp. 1341-1373, 1973.

55. Clark, A. P. and Tint, U. S. "Linear and non-linear Transversal Equalisers for Base-band Channels", Radio and Electron. Eng., No.45, pp. 271-283, 1975.

56. Belfiore, C. A. and Park, J. H. jr. "Decision Feed-back Equalisation", Proc. IEEE, Vol.67, pp. 1143-1156, Aug. 1979.

57. Lucky, R. W. "Techniques for Adaptive Equalisation of Digital Communication System", BST. J., Vol.45, No.2, pp. 255-286, Feb. 1966.

58. Clark, A. P. and Abdullah, S. N. "Near Maximum Likelihood Detector for Voice-band Channels", IEE Proc., Part F, Vol.134, No.3, pp. 217-226, June 1987.

59. Ungerboeck, G. "Adaptive Maximum Likelihood Receiver for Carrier Modulated Data Transmission System", IEEE Trans. on Commun., Vol.COM-22, No.5, pp. 624-636, May 1974

60. McVerry, F. "High Speed Data Transmission over HF Radio Link", Ph.D. Thesis, Dept. of Electronic and Electrical Eng. Loughborough Univ. of Technology, 1982.

61. Clark, A. P. and McVerry, F. "Channel Estimation for an HF Radio Link ", IEE. Proc., Part F, Vol.128(1), pp. 33-42, Feb. 1981.

62. Clark, A. P., Kwong, C P. and McVerry, F. "Estimation of the Sampled Impulse Response of a Channel", Signal Processing, Vol.2, pp. 39-53, 1980.

63. Haykin, S. "Adaptive Filter Theory", Prentice-Hall, 1986.

64. Morrison, N. "Introduction to Sequential Smoothing and Prediction", McGraw-Hill, 1968.

65. Magee, F. R. and Proakis, J. G. "Adaptive Maximum Likelihood Sequence Estimation for Digital Signalling in the Presence of Inter symbol Interference", IEEE Trans. Inform. Theory, Vol.IT-19, pp. 120-124, 1973.

66. Treichler, J. R., Johnson, C. R., and Larimore, M. G. "Theory and Design of Adaptive Filters", John Wiley and Sons, 1987.

67. Clark, A. P. and Hariharan, S. "Channel Estimator for Voice-band HF Radio Links ", Publication No.82, pp. 213-218, Fifth International Confer. on Digital Processing of Signals in Commun., Loughborough UK., Sept. 1988.

68. Andrews, A. "A Square Root Formulation of the Kalman Covariance Equations", AIAA. J., Vol.6, No.6, June, 1968.

69. Godard, D. "Channel Equalisation using a Kalman Filter for Fast Data Transmission" IBM J. RES. Develop., pp. 267-272, May 1974.

70. Ljung, G. L., Morf, M. and Falconer, D. "Fast Calculation of Gain Matrices for Recursive Estimation Schemes", INT. J. Control, Vol.27, No.1, pp. 1-19, 1978.

71. Richards, G. A. "Implementation of Kalman Filters for Process Identification", GEC, Journal of Research, Vol.1, No.2, pp. 100-107, 1983.

72. Kalman, K. E. and Bucy, R. S. "New Results in Linear Filtering and Prediction Theory", Trans. ASME J. Basic Eng., Vol.83-D, No.1, pp. 95-108. Mar.1961.

73. Clark, A. P. and McVerry, F. "Improved Channel Estimator for an HF Radio Link", Signal Processing, Vol.5, pp. 241-255, May 1983.

74. Mark, J. W. "A Note on the Modified Kalman Filter for Channel Equalisation", Proc. IEEE, Vol.61, pp. 481-482, 1973.

75. Widrow, B., McCool, J. and Ball, M. "The Complex LMS Algorithm", Proc. IEEE, Vol.63, pp. 719-720, 1975.

76. Benvenuto, N. G. L. and Marchesaqni, R. "Channel Estimation in Time Varying Multipath Environments", Proc. 6th Int. Conf. on HF Radio System and Techniques, York, England, July 1994.

77. Clark, A. P. "Design Techniques for Non-linear Equalisers", Proc. IEE, Vol.120, pp.329-333, 1973.

78. Clark, A. P. and Harun, R. "Assessment of Kalman Filter Channel Estimators for an HF Radio Link", IEE Proc., Part F, Vol.133, pp. 513-521, Oct. 1986.

79. Beare, C. T. "The Choice of the Desired Impulse Response in Combined Linear-Viterbi Algorithm Equalisers", IEEE Trans. on Commun., Vol.COM-26, pp.1301-1307, Aug. 1978.

80. McLaughlin, S., Mulgrew, B. and Cowan, C. F. N. "Performance Comparison of Least Squares and Least Mean Squares Algorithms as HF Channel Estimators", Proc. ICASSP 87, Dallas, pp. 49.21-49.2.4, April 1987.

81. Proakis, J. G. and Ling, F. "Recursive Least Squares Algorithm for Adaptive Equalisation of Time Varying Multi-path Channels", P{roc. Int. Conf. Commun., Amsterdam, The Netherlands, May 1984

82. Barham, D. M. and Humphries, D. E. "Derivation of the Kalman Filtering Equations from Elementary Statistics Principles", Technical Report 69095, Royal Aircraft Establishment, May 1989.

83. Mueller, M. S. "Least Squares Algorithm for Adaptive Equalisers", BST. J., Vol.60, pp. 1905-1925, Oct. 1981.

84. Alexender, S. T. "Adaptive Signal Processing-Theory and Application", Springer-Verlag, 1986.

85. Carlson, N. A. "Fast Triangular Formulation of the Square Root Filter", AIAA J., Vol.11, No.9, Sept. 1973.

86. Satorius, E. H. and Alexander, S. T. "Channel Equalisation using Adaptive Lattice Algorithm", IEEE Trans. on Commun., Vol.COM-27, No.6, pp. 899-905 1979.

87. Hawksford, M. J. and Rezaee, N. "Adaptive Mean-Square-Error Transversal Equaliser", IEE Proc. Part F., Vol.128, pp. 296-304, 1981.

88. Eleftheriou, E. and Falconer, D. D. "Adaptive Equalisation Techniques for HFChannels", IEEE Trans. on Selected Areas in Commune., Vol.SAC-5, pp. 238-247,1987.

89. Luvison, A., Sacch, L. and Tambuselli, G. "Theory and Implementation of Adaptive Equaliser", IEEE, CH1435-7/79, pp. 45.6.1-45.6.6, 1979.

90. Ungerboek, G. "Theory on the Speed of Convergence in Adaptive Equalisers for Digital Communication", IBM. J., pp. 546-555, Nov. 1972.

91. Widrow and Walach, E. "On the Statistical efficiency of the LMS Algorithm with Non Stationary inputs", IEEE, Trans. on infor. theory, Vol.IT-30, No.2, pp.211-221, 1984.

92. Ziegler, R. A. and Cioffi, J. M. "Estimation of Time-Varying Digital Radio Channels", IEEE, Trans. on Vehicular Technology, Vol.41, No.2, pp.134-151, May 1992.

93. Treichler, J. R. and Agee, B. G. "A new Approach to Multipath Correction of Constant Modulus Signals", IEEE, Trans. on ASSP, Vol.ASSP-31, No.2, pp. 459-471, April 1983.

94. Wehrman, R., Vander List, J. and Meissner, P. "A Noise-Insensitive Compromise Gradient Method for the Adjustment of Adaptive Echo Cancellers", IEEE, Trans. on Commun., Vol. COM-28, No.5, pp. 753-759, May 1980.

95. Roy, S. and Shynk, J. J. "Analysis of the Momentum LMS Algorithm", IEEE, Trans. on ASSP, Vol.ASSP-38, No.12, pp. 2088-2097, Dec. 1990.

96. Eweda, E. and Macchi, O. "Equalisation of Rapid Selective Fading with Unknown and Time-Varying Form", IEEE, CH1746-7/82/, pp. 1390-1393, 1982.

97. Lawrence, R.E and Kaufman, H. "The Kalman Filter for the Equalisation of a Digital Communication Channel", IEEE Trans. on Commun. Vol, COM-19, pp. 1137-1141, 1971.

98. Cioffi, J. M. and Kailath, T. "Fast Recursive Least-square Transversal Filters for Adaptive Filtering", IEEE, Trans. on Acoustics Speech and Signal Processing, Vol. ASSP-32, pp. 304-337, 1984.

99. Widrow, B. and Hoff, M. E. "Adaptive Switching Circuits", in IRE Wescon Conv. Rec., Pt.4, pp. 96-104, 1960.

100. Ser, W. "Detection of Digital Signals Transmitted over a known Time Invariant Channel", Ph.D. Thesis, Loughborough university of technology, 1982.

101. Clark, A.P., Zhu, Z.C. and Joshi, J.K. "Fast Start-up Channel Estimation", IEE Proc., 131, Pt.F, pp. 375-382, 1984.

102. Soliman, S.S. and ScholtZ, R. A. "Synchronization over Fading Dispersive Channels", IEEE, Trans. on Commun. Vol. COM-36, No.4, pp. 499-505, Apr. 1988.

103. Clark, A.P., Abdullah, S. N. and Hau, S.F. "Improved Time Synchronisation for an HF Radio Modem", IEE Proc., Vol.136, Pt.I, No.1, pp. 47-56, Feb. 1989.

104. Maslin, N.M. "High Data Transmission over HF Links", The Radio and Electronic Engineer, Vol.52, No.2, pp. 75-78, Feb. 1982.

105. Hodgkiss, W., Turnner, L.F., and Pennington, J., " Serial Data Transmission over HF Radio Links ", IEE Proc., Vol.131, Pt.F, No.2, pp. 107-116, April 1984.

106. Pennington, J., "Techniques for Medium-speed Data Transmission over HF Channels", IEE Proc., Vol.136, Pt.I, No.1, pp. 11-19, Feb., 1989.

107. Darnell, M., " Medium-speed Digital Data Transmission over HF Channels", Proc. IERE Conf. on Digital processing of signal in communications, Loughborough, pp. 393-402, 1977.

108. Brakemeier, A. and Lindner, J. " Single-tone HF Data Transmission with 2.4KBps and 4.8KBps ", AEG Research Center, Federal Republic of Germany.

109. Hsu, F. M., " A Single-tone HF Serial Modem ", IEEE Conf., 1984.

110. Perl, J. M. and Aharoni, M., "TMS320C10 Implementation of an Improved Kineplex Type HF Modem ", Tadiran, Digital Equipment Plant, Communication Division.

111. Pennington, J., "Comparative Measurements of Parallel and Serial 2.4Kbps Modem" IEE HF Communications Systems and Techniques Confer., London, pp.141-144, Feb., 1982.

112. Mosier, R. and Clabaugh, R. G., " Kineplex : a Bandwidth Efficient Binary Data Transmission System ", AIEE Trans. Commun. Elec., No.76, pp.723-728, 1958.

113. Bello, P.A., " Selective Fading Limitation of Kathryn Modem and some System Design Considerations ", IEEE Trans. on Commun., COM-13, pp.320-333, 1965.

114. Rizk, Z. and Marshall, W.G., " Some Detection and Estimation Techniques for Fading dispersive HF Digital Channels ", Submitted to the Proc. IEE, 1995.

115. The TMS320C3x User'S Guide, Texas Instruments, 1993.

116. The TMS320C4x User'S Guide, Texas Instruments, 1993.

117. Digital Signal Processing Applications with the TMS320 Family, Texas Instruments, 1990.