National Library
of Canada

Bibliothèque nationale
du Canada

Canadian Theses Service

Service des thèses canadiennes

Ottawa, Canada
K1A 0N4

NOTICE

The quality of this microform is heavily dependent upon the quality of the original thesis submitted for microfilming. Every effort has been made to ensure the highest quality of reproduction possible.

If pages are missing, contact the university which granted the degree.

Some pages may have indistinct print especially if the original pages were typed with a poor typewriter ribbon or if the university sent us an inferior photocopy.

Reproduction in full or in part of this microform is governed by the Canadian Copyright Act, R.S.C. 1970, c. C-30, and subsequent amendments.

AVIS

La qualité de cette microforme dépend grandement de la qualité de la thèse soumise au microfilmage. Nous avons tout fait pour assurer une qualité supérieure de reproduction.

S'il manque des pages, veuillez communiquer avec l'université qui a conféré le grade.

La qualité d'impression de certaines pages peut laisser à désirer, surtout si les pages originales ont été dactylographiées à l'aide d'un ruban usé ou si l'université nous a fait parvenir une photocopie de qualité inférieure.

La reproduction, même partielle, de cette microforme est soumise à la Loi canadienne sur le droit d'auteur, SRC 1970, c. C-30, et ses amendements subséquents.

Investigation of Coding and Equalization Algorithms and VLSI Architectures for a High Bit-Rate Digital Subscriber Loop System

by

P. Mohanraj, B.Eng., M.Tech

A thesis submitted to the
Faculty of Graduate Studies and Research
in partial fulfilment of the requirements
for the degree of
Doctor of Philosophy

Ottawa-Carleton Institute for Electrical Engineering

Department of Electronics

Faculty of Engineering

Carleton University

December 1990

©P. Mohanraj, 1990

National Library
of Canada

Bibliothèque nationale
du Canada

Canadian Theses Service Service des thèses canadiennes

Ottawa, Canada
K1A 0N4

The author has granted an irrevocable non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of his/her thesis by any means and in any form or format, making this thesis available to interested persons.

The author retains ownership of the copyright in his/her thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without his/her permission.

L'auteur a accordé une licence irrévocable et non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de sa thèse de quelque manière et sous quelque forme que ce soit pour mettre des exemplaires de cette thèse à la disposition des personnes intéressées.

L'auteur conserve la propriété du droit d'auteur qui protège sa thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

ISBN 0-315-68855-6

Canada

The undersigned hereby recommend to
the Faculty of Graduate Studies and Research
acceptance of the thesis,

**Investigation of Coding and Equalization Algorithms
and VLSI Architectures for a High Bit-Rate
Digital Subscriber Loop System**

submitted by Palanisamy Mohanraj,
in partial fulfilment of the requirements for the degree of
Doctor of Philosophy

Ted Kwoinicurdu

Thesis Supervisor

David Falcon

Thesis Supervisor

JS Wright

Chairman,
Department of Electronics

Department of Electronics
Faculty of Engineering
Carleton University
February, 1991

Abstract

The objective of this thesis has been to find suitable transmission schemes that would enhance the data rate of the digital subscriber loop (DSL) system and explore efficient VLSI architectures for their implementation. The channel capacity results indicate that data rates significantly higher than the current standard of 160 Kbits/s are possible on this channel. The characteristics of the subscriber loop channel suggests that bandwidth and power efficient coded modulation systems would be appropriate for this channel. One such method, called trellis coded modulation using baseband transmission scheme is considered in this study.

We determine the asymptotic performance of encoders of different complexity by evaluating bounds on the minimum Euclidean distance d_{min} of these codes on the subscriber loop channel. We show that more complex codes with large numbers of states are needed to obtain reasonable coding gains on this channel. The minimum distance indicates the performance achievable only with an ideal maximum-likelihood sequence estimating receiver, which is very complex and physically impossible to realize. Hence, we investigate the performance of a practical receiver based on reduced state sequence estimation. Through simulation studies, we determine the performance (bit error rate) of the reduced complexity sequence estimator implemented by the M-algorithm. The symbol error rate results for different loop lengths are presented. The simulation results indicate that the coded system could improve the transmission range of the high bit rate DSL system by about 1 Kft.

A fractionally spaced forward filter is used at the receiver front end to partially equalize the channel and also to whiten the crosstalk at the input of the sequence estimator. The sensitivity of the fractionally spaced forward filter performance to the parameters of the system is also presented.

Considering the importance of the VLSI realization of the transceiver, we have

proposed efficient VLSI architectures for some of its functional units. Implementation of the M-algorithm based sequence estimator, fractionally spaced forward filter and the transversal filter of an echo canceller or DFE are addressed. We estimate that approximately 30 sq.mm of silicon area may be required for an 8 state M-algorithm receiver. In order to simplify the implementation of the forward filter, we have explored the possibility of echo cancellation in the oversampled domain of the sigma-delta modulation based A/D converter.

The transversal filter function is often used in the echo canceller and the DFE. The complexity of the filter is proportional to the number of tap coefficients. The impulse response of the echo path and the channel itself extend over a few hundred baud intervals at the high bit rates considered in this study, and have a slowly decaying tail. We propose a piecewise linear approximation of the tail to reduce the implementation complexity of these transversal filters. For a given hardware complexity, about 20 dB improvement in the SNR is shown to be possible.

Acknowledgements

I would like to express my sincere gratitude to my thesis supervisors Dr. T. A. Kwasniewski and Dr. D. D. Falconer for their guidance and advice throughout the course of this study.

I wish to thank my parents, my wife Usharani and daughter Bhavana for their support and understanding.

I am immensely thankful to my graduate colleagues and the staff who have helped me throughout this programme.

The financial support provided by the Canadian Commonwealth Scholarship Committee and the Telecommunications Research Institute of Ontario (TRIO) is gratefully acknowledged.

Finally, I would like thank my employer Semiconductor Complex Ltd, India, for granting me study leave to do my doctoral programme.

Contents

Abstract	i
Acknowledgement	iii
1 Introduction	1
1.1 Thesis motivation	3
1.2 Contributions of the thesis	5
1.3 Thesis overview	6
2 High Bit Rate Digital Subscriber Loop System	8
2.1 Introduction	8
2.1.1 Carrier serving area	9
2.1.2 Channel capacity	9
2.2 Channel impairments	10
2.3 Multichannel transmission	16
2.4 Trellis precoding	17
2.5 QAM transmission	19

2.6	Coordinated transmission	20
2.7	Summary	21
3	Equalization Under Cyclostationary Interference	22
3.1	Introduction	22
3.2	Fractionally spaced forward filter for DFE or sequence estimator	24
3.3	Effect of background white noise level on the equalizer performance	26
3.4	Effect of the lowpass receive filter bandwidth on the equalizer performance	30
3.5	Conclusion	32
4	Trellis Coded Modulation for Subscriber Loop Channel	35
4.1	Introduction	35
4.2	Trellis coding	37
4.3	Minimum distance of the coded system on an ISI channel	38
4.4	Combined equalization and decoding	44
4.5	Simulation set up	51
4.6	Simulation results	52
4.7	Conclusion	54
5	VLSI Design Considerations of the Sequence Estimator	58
5.1	Introduction	58
5.2	M-algorithm for sequence estimation on ISI channels	61

5.3	Branch and Path metric computation	63
5.3.1	Duplicate state elimination	64
5.3.2	Metric sorting	68
5.3.3	Path memory updating	72
5.4	Silicon area estimate for the sequence estimator	73
5.5	Conclusion	83
6	Oversampled Echo Cancellation	84
6.1	Introduction	84
6.2	Conventional echo cancellation	85
6.3	Oversampled echo cancellation	87
6.4	Adaptation of the oversampled echo canceller	93
6.5	Combined decimation and forward filtering	94
6.6	Silicon area estimate	97
6.7	Conclusion	103
7	Reduced Complexity Transversal Filter Implementation	104
7.1	Introduction	104
7.2	Piecewise linear approximation	106
7.3	Implementation and complexity	112
7.4	Performance comparison	115
7.5	Adaptation behaviour	116
7.6	Adaptive segment length determination	118

7.7	Adaptation of PLF filter using sign algorithm	124
7.8	Hardware complexity (wordlength determination)	126
7.9	Silicon area estimation for the PLF and conventional methods	128
7.10	Conclusion	131
8	Conclusion	136
8.1	Future Work	139
	Bibliography	141
	Appendices	
A	Optimal piecewise linear fit filter coefficients	150
B	Convergence behaviour of the piecewise linear fit filter	156

List of Figures

2.1	Attenuation of twisted pair wires	11
2.2	Sample pulse response of the loop	12
2.3	Echo cancellation system	14
2.4	Crosstalk paths on the subscriber loop	15
2.5	Multichannel transmission system	18
3.1	Digital transmission system with fractionally spaced equalizer	24
3.2	Equalized pulse responses	27
3.3	Total noise spectrum	28
3.4	Equalized crosstalk path power spectra	33
3.5	Equalized crosstalk path pulse responses	34
4.1	Trellis encoder	38
4.2	Trellis diagram ideal channel	39
4.3	Trellis diagram of dispersive channel	41
4.4	Flowchart of dmin algorithm	43
4.5	Lower bound (stationary crosstalk)	45

4.6	Upper bound (stationary crosstalk)	46
4.7	Lower bound (cyclostationary crosstalk)	47
4.8	Upper bound (cyclostationary crosstalk)	48
4.9	Receiver with DFE and trellis decoder	51
4.10	Combined equalization and decoding receiver	55
4.11	Probability of error-stationary crosstalk	56
4.12	Performance-cyclostationary crosstalk	57
5.1	Paths considered in the M-algorithm	62
5.2	Branch metric computation	65
5.3	Channel state comparison	67
5.4	Duplicate state comparator	69
5.5	State metric sort	71
5.6	Path memory register Cell	72
5.7	Path memory	74
5.8	M-algorithm based sequence estimator	76
5.9	Parallel multiplier	77
5.10	Digital comparator	78
6.1	Conventional echo cancellation	86
6.2	Sigma-delta modulator	88
6.3	Oversampled echo cancellation	89
6.4	Spectrum of the received signal at the Sigma-delta modulator output	91

6.5	Spectrum with the echo cancelled	92
6.6	Residual echo by conventional definition	95
6.7	Residual echo after decimation filter	96
6.8	Combined decimation and forward filter response	98
6.9	Architecture of the forward filter in the conventional scheme	99
6.10	Architecture of the combined decimation and forward filter	102
7.1	Sample impulse response of the loop	107
7.2	Piecewise linear approximation of the impulse response	108
7.3	Piecewise linear approximation of a segment	109
7.4	Approximation error	111
7.5	Performance (SNR) vs computational complexity	117
7.6	Adaptive filter	119
7.7	Mean squared error performance	120
7.8	Test loops	122
7.9	Impulse responses of the test loops	123
7.10	Convergence behaviour of the sign algorithm	125
7.11	Wordlength of the PLF filter variables	129
7.12	Architecture of the PLF filter	130
7.13	Wordlength of the conventional filter variables	133
A.1	Segmentation of the impulse response	151

List of Tables

3.1	SNR of FSE under stationary and cyclostationary noise	26
3.2	Effect of white noise level on the performance	29
3.3	Effect of bandwidth on the performance	31
5.1	Computational time of the processing steps in the M-algorithm	82
6.1	Silicon area estimate for the conventional scheme	101
7.1	Computational complexity	115
7.2	Partioning schemes with different complexity	116
7.3	Optimal partitioning of the test loops	123
7.4	Performance under various partitioning schemes	124
7.5	PLF filter area estimate	132
7.6	Conventional transversal filter area estimate	134

List of symbols

ISDN	Integrated Services Digital Network
HDSL	High bit rate Digital Subscriber Loops
CSA	Carrier Serving Area
ISI	Inter Symbol Interference
TCM	Trellis Coded Modulation
SNR	Signal-to-Noise Ratio
VLSI	Very Large Scale Integrated circuits
d_{min}	minimum distance
MLSE	Maximum Likelihood Sequence Estimator
FSE	Fractionally Spaced Equalizer
CMOS	Complementary Metal Oxide Semiconductor
PAM	Pulse Amplitude Modulation
QAM	Quadrature Amplitude Modulation
NEXT	Near-end crosstalk
DFT	Discrete Fourier Transform
IIR	Infinite Impulse Response
FIR	Finite Impulse Response
DFE	Decision Feedback Equalizer
N_0	Noise power spectral density
$\phi_0(t)$	end-to-end impulse response
$\phi_l(t)$	NEXT impulse response

Khz	Kilo hertz
Mhz	Mega hertz
Hz	Hertz
Kft	Kilo feet
AWGN	Additive White Gaussian Noise
DFSE	Decision Feedback Sequence Estimation
RSSE	Reduced State Sequence Estimation
PLF	Piecewise Linear Fit
LMS	Least Mean Squared
sq.mm	square millimeter
MSE	Mean Squared Error

Chapter 1

Introduction

This thesis is concerned with the methods and their realization to improve the data rate of the digital transmission system on the subscriber loop channel. Digital transmission on the subscriber loop is the last link in providing digital end-to-end connectivity. Of the competing techniques such as echo cancellation (EC) and time compression multiplexing (TCMUX)[ABG81], the echo cancellation method is adopted as a standard in North America due to its longer reach. A data rate of 160 Kbits/sec has been chosen as the standard for basic rate Integrated Services Digital Network (ISDN)[ANS88]. Several studies[JF87][HCKC89] have been done to determine the channel capacity of the subscriber loop channel. They indicate that data rates significantly higher than 160 Kbits/sec can be supported on this channel.

Recently there is some effort to evolve a standard at 800 Kbits/sec on a limited range of the subscriber loops called Carrier Serving Area (CSA)[Lec88]. An immediate use for this technology is that two such pairs can be used to provide DS1 rate (1.544 Mbits/sec) transmission to the customers without having to do any special engineering of the line. The major impairment to transmission at this high rate is the near end crosstalk from other pairs in the same cable and the intersymbol interference (ISI) due to severe attenuation and phase distortion of the channel. Coding

is generally used in digital transmission systems to gain some noise immunity while equalization is used to minimize the effects of intersymbol interference. To achieve low error rates (10^{-7}) with a suitable noise margin, coded systems are found to be necessary [Jos88].

Conventional coding schemes such as block coding and convolutional coding introduce redundancy by increasing the symbol rate on the line to obtain some coding gain (i.e., a reduction in the required signal-to-noise ratio (SNR) for a given error rate). These coding methods are used extensively in the low SNR channels such as satellite channels due to the availability of large bandwidth. But they may not be suitable for subscriber loop channel where the SNR is relatively high and the bandwidth is at a premium. Also, an increase in the symbol rate on the line is not desirable in the subscriber loop channel because the near end crosstalk interference level increases with the line rate thus nullifying part of the gain made by the coder.

Trellis coded modulation (TCM) is a method that does not increase the bandwidth required for transmission, but increases the number of signal levels to provide for the redundancy, to obtain the coding gain. Coding gains of the order of 3 to 6 dB have been shown [Ung82] to be possible with this approach on an additive gaussian noise channel. Communication standards (V.32 and V.33) based on this technique [V.3a][V.3b] have been developed for high speed voice band modems. Data rates as high as 19.2 Kbits/sec (close to the capacity of this channel) have been achieved [Cod] using complex multi-dimensional trellis codes.

Trellis coded systems necessitate the use of sequence estimation receivers which are more complex than the symbol by symbol detector such as the decision feedback equalizer used in the basic rate transmission scheme. To economically provide the ISDN service, VLSI implementation of the transceiver is mandatory. Methods for reducing the signal processing complexity of the receiver and their VLSI design considerations are explored.

1.1 Thesis motivation

The impressive performance of the trellis coded modulation technique in the voice-band data transmission systems motivated us to look at the possibility of applying this method to high bit rate transmission on the subscriber loop channel. The coding gains of the TCM mentioned above are obtained on an ideal channel with no ISI. But the subscriber loop channel introduces severe ISI. Hence it is important to evaluate the performance of these codes on an ISI channel if one considers using it on these channels. At high SNR, the performance is related to the minimum distance d_{min} between the coded sequences. On an ideal channel the distance between the received signal sequences is the same as that of the encoder output. But in an ISI channel the distance between the received signal sequences is altered by the channel impulse response. While the minimum distance indicates the performance achievable with an impractically complex receiver, the performance of a practical system with limited hardware complexity must be evaluated to determine the usefulness of the coded system for this application. We evaluate the symbol error probability of receivers with reduced state decoders by simulation.

Equalization is generally used to reduce the intersymbol interference on dispersive channels. For a coded system on an ISI channel the tasks of equalization and decoding can be performed either separately or jointly.

One way of removing the intersymbol interference (ISI) is to equalize the channel using a linear equalizer and force the overall channel to an ideal channel with no ISI. There is a penalty for equalizing the channel, which is an enhancement of the noise. For a subscriber loop channel with severe amplitude distortion the noise enhancement is severe and degrades the performance of the digital transmission system. Another method is to use a decision feedback equalizer (DFE). The forward filter of the DFE partially equalizes the channel response to suppress any precursor ISI, and the feedback filter subtracts any post cursor interference. The noise enhancement is

less severe in this scheme. These receivers with symbol by symbol detectors may not provide the required performance for the high bit rate system. Hence, more complex receivers using sequence estimation may be needed. Maximum likelihood sequence estimation (MLSE) has been shown[For72] to be the optimum receiver (in the sense that it minimizes the error event probability) for data transmission on dispersive channels. The MLSE of a coded system on ISI channel would use the combined code and ISI trellis to optimally decode the transmitted sequence.

The complexity of the transceiver and its VLSI implementation is an important consideration in the design of the digital transmission scheme for subscriber loop channel. The Viterbi algorithm is an efficient method for implementing a MLSE receiver. The complexity of the Viterbi algorithm receiver on ISI channel increases exponentially with the ISI length. For subscriber loops the ISI length is of the order of few hundred symbol intervals at the high bit rate being considered here. Hence a full Viterbi receiver is impractical to implement. There have been methods proposed to reduce the complexity of the sequence estimator with minimal loss in the performance. One such method called M-algorithm is considered in this study. Efficient architectures for the VLSI implementation of the reduced state sequence estimator are explored.

Transversal filter operation is often used in the transceiver to perform echo cancellation or branch metric computation of the sequence estimator or decision feedback equalization. The complexity of this filter is proportional to the number of tap coefficients of the filter, which tends to be large in the high bit rate system. Any attempt to reduce the complexity of the implementation of this filter would be worthwhile. We propose a novel method in which the implementation complexity increases logarithmically rather than linearly with the length of the filter.

In general this study aims to evaluate the applicability of trellis coded modulation as a means of improving the range (loop length) of the digital subscriber loop system

and determine the feasibility of VLSI realization of the transceiver.

1.2 Contributions of the thesis

We have determined the range (loop length) of the high bit rate (800 Kbits/sec) digital subscriber loop system employing trellis coded modulation at the transmitter and a reduced complexity combined equalization and decoding at the receiver. The coded system was found to provide approximately 1 Kft of additional range as compared to an uncoded system with a simple DFE receiver. We observe that encoders with large numbers of states provide better performance on ISI channels. We found that reduced complexity sequence estimator implemented by the M-algorithm has better performance as compared to that of a DFSE. The system range under cyclostationary crosstalk is shown to be significantly longer than that under stationary crosstalk.

The performance (SNR) of the fractionally spaced equalizer at the receiver front end is shown, under cyclostationary crosstalk, to be sensitive to the level of background white noise in the system as well as the bandwidth of the anti-aliasing lowpass filter that precedes the forward filter.

Efficient VLSI architectures for the key functional units of the M-algorithm based sequence estimator are proposed in this thesis. We estimate that the VLSI realization of such a sequence estimator would require a silicon area of 30 sq. mm. in 1μ CMOS technology. In an effort to reduce the hardware required for the implementation of the fractionally spaced forward filter, and the decimation filter of an oversampled A/D converter at the receiver front end, we examined a new method for echo cancellation in the oversampled domain.

A piecewise linear fit method to reduce the computational complexity of the transversal filter, used in the echo cancellation and the DFE, is presented in this

study. An improvement of 20 to 30 dB in SNR is shown to be possible, for a given number of multiplications and additions, by this approach as compared to the conventional implementation.

1.3 Thesis overview

The following describes an overview of the thesis.

Chapter 2 describes the approaches proposed for high bit rate digital subscriber loop system. The factors limiting the data rate on this channel are described. Several approaches to improve the data rate considered by researchers in this area are briefly described.

The high bit rate DSL(HDSL) systems may need a fractionally spaced forward filter. Optimum fractionally spaced equalizer (FSE) for HDSL system is obtained in Chapter 3. The performance achievable with FSE under cyclostationary crosstalk condition is compared to its performance under stationary crosstalk. The crosstalk interference would have stationary statistics when the interferers have randomly distributed phases. But it has cyclostationary behaviour when all the disturbing pairs in the cable are synchronized in phase and frequency to the same clock. These forward filters are used at the front end of the sequence estimating receivers considered in Chapter 4. The receive filter bandwidth is shown to have a significant effect on the performance of FSE under cyclostationary interference.

The trellis coded modulation technique to improve the range of the high bit rate DSL system is presented in Chapter 4. The performance of codes with different complexity is evaluated by computing the minimum distance d_{min} of the received sequences. A combined equalization and decoding receiver with reduced number of states is described. The probability of symbol error of the coded modulation system is obtained for various loop lengths under stationary and cyclostationary crosstalk

conditions by simulation. These simulations are performed for receivers with 8 states using the M-algorithm for sequence estimation.

Chapter 5 discusses the VLSI design considerations of the sequence estimator. Efficient VLSI architectures are proposed for the path memory update, sorting and branch metric calculation in an M-algorithm based sequence estimator. The silicon area and the required speed of the processing element are estimated for its VLSI implementation using 1μ CMOS technology.

A new approach for cancelling echo in the oversampled domain is considered in Chapter 6. The purpose of this is to combine the decimation filter of the oversampled A/D converter and the fractionally spaced forward filter, so that the silicon area required for the implementation of these filters can be reduced. The adaptation behaviour of the echo canceller in this scheme and its performance are determined by simulation. Estimates of the silicon area required for the implementation of the combined filter is compared to that of a conventional implementation with two separate filters. Only a preliminary analysis of the problem is done in this study.

Chapter 7 presents a new approach called piecewise linear fit technique for reducing the complexity of the transversal filter operation that is common to many functional units such as branch metric calculation or ISI cancellation or echo cancellation of the receiver. Significant improvement in the SNR or residual error is obtained by this method under fixed hardware complexity condition, compared to a transversal filter implementation. The adaptation behaviour of the new method is studied in detail. A VLSI architecture is proposed for the implementation of the new scheme and the silicon area required for the implementation of this scheme is compared to that of the transversal filter scheme.

The contributions of this thesis and suggestions for further work in this area are described in Chapter 8.

Chapter 2

High Bit Rate Digital Subscriber Loop System

2.1 Introduction

Demand for higher rates stems from the desirability of providing primary rate access and T1 services (1.544 Mbits/s) over non-repeated loops. As there is a significant difference between the theoretical capacity of the subscriber line channel and the current standard of 160 Kbits/sec, many approaches to bridge this gap are being considered. These include trellis coded modulation technique using baseband PAM proposed in this thesis and others such as trellis precoding[EFBP90], multichannel transceiver[TCDC89] and trellis coded QAM technique proposed in[ATT89]. A brief review of each of these methods is given in this chapter. The impairments limiting the data rate in the digital subscriber loop system are echo, intersymbol interference, crosstalk and impulse noise. The characteristics of these impairments and the loop plant will be discussed in this chapter. At the proposed rate of 800 Kbits/sec the problems of echo, crosstalk and intersymbol interference (ISI) are more difficult to solve than at the basic line rate of 160 Kbits/sec.

2.1.1 Carrier serving area

The local loop plant consists of loops of various lengths with or without bridged taps. Out of this total population of loops about 75 to 80% of the loops fall in the category called Carrier Serving Area (CSA). This CSA designation[Lec89] limits the length of the loop to 9 Kft if any section of the loop contains 26 gauge wire and to 12 Kft if there is none. The total bridged tap length is limited to 2.5 Kft with no single tap greater than 2.0 Kft. Preliminary estimates[Lec89] indicate that the high bit rate ISDN service can be provided to customers within the CSA limits.

2.1.2 Channel capacity

The capacity of a dispersive channel is given by[Gal68]

$$C = \int_0^{BW} \log(K \cdot \frac{|c(f)|^2}{N(f)}) df \quad (2.1)$$

where K is a constant that is determined by the total transmitted power,

$|c(f)|$ is the channel frequency response, and

$N(f)$ is the power spectral density of the noise (assumed to be stationary).

Using this equation and other means there have been several studies to determine the capacity of the digital subscriber loop channel[JF87][HCKC89][Law89][KS90][AC89][CA85]. These studies indicate that the capacity under near end crosstalk (NEXT) is much less compared to that under background white noise. The capacity of the CSA loops has been determined to be about 2 Mbits/sec under NEXT interference. Hence transmission schemes approaching the channel capacity are necessarily more complicated.

2.2 Channel impairments

The major impairments to data transmission on subscriber loops are

- 1) Thermal noise
- 2) Intersymbol Interference
- 3) Echo
- 4) Crosstalk
- 5) Impulse noise

The thermal noise is due to the random motion of electrons in the wire-pair and other electronic components of the system. The power spectral density of this noise has been estimated to be about -110 dBm/Hz.

The intersymbol interference is caused by the severe amplitude and phase distortion of the subscriber loop channel[EMS79]. The attenuation of the 26 gauge loop is about 10 dB/Km at 100 kHz as shown in Figure 2.1. The pulse response of a typical CSA loop is shown in Figure 2.2. It can be observed that the post cursor of the pulse response extends over several hundred baud periods causing intersymbol interference. Also, most of the energy is concentrated in the first few baud periods. It has been shown[Clas5] that the pulse response of a minimum phase network consists of only trailing echos. The subscriber loop channels have been observed to have the characteristics of minimum phase networks.

The echo cancelling scheme[Mes86][AHM32] for full duplex transmission of data on a single pair of wires uses a hybrid to separate the signals transmitted in the opposite directions. A schematic diagram of this scheme is shown in Figure 2.3. The echo from the near end, which could be about 35 to 40 dB stronger compared to the far end signal, should be subtracted from the received signal before detecting the far end signal. Echo suppression by about 60 dB is quite common in the basic rate

Figure 2.1: Attenuation of twisted pair wires

Figure 2.2: Sample pulse response of the loop

ISDN systems. The requirements of the echo canceller in the high bit rate system are expected to be less severe because the far end signal is attenuated less due to shorter loops in the Carrier Serving Area. Still the echo path impulse response spans many symbols due to the higher baud rate. As will be seen later, it may be desirable to use a fractionally spaced $(T/2)$ forward filter in high bit rate DSL which necessitates the use of an echo canceller with twice the complexity of the baud rate echo canceller.

The crosstalk interference is due to coupling of signals from other pairs in the same cable. There are two types of crosstalk. One is the near end crosstalk and the other is far end crosstalk. A schematic diagram of the crosstalk coupling paths is shown in Figure 2.4. The near end crosstalk is a major impairment in the high bit rate DSL system[Pie86][Sze89][FLF89]. The crosstalk coupling depends on the cable structure. The capacitive nature of the coupling increases the coupling at high frequencies. The mean squared near end crosstalk coupling power spectrum, averaged over many lines, is observed[CGS83] to be given by the expression

$$N^2(f) = K_N \cdot f^{3/2} \quad (2.2)$$

where ' K_N ' is the loss parameter that depends on the cable structure and the number of crosstalkers. ' K_N ' was calculated to be 10^{-13} for 49 interferers in the DSL system[Lin90].

The far end crosstalk level is usually much lower than the near end crosstalk level because the disturbing signal is already attenuated by the line loss. The mean squared far end crosstalk coupling power spectrum, averaged over many lines, is observed [CGS83] to be given by

$$F^2(f) = K_F \cdot \frac{l}{l_0} \cdot \frac{f}{f_0} \quad (2.3)$$

where ' K_F ' is the loss parameter that is measured at the same reference length l_0 and reference frequency f_0 .

Impulse noise is due to natural causes such as lightning or due to switching equip-

Figure 2.3: Echo cancellation system

Figure 2.4: Crosstalk paths on the subscriber loop

ments in the central office or other heavy electrical equipments at the customer premises or the central office. These disturbances are short-lived large amplitude noise pulses. The impulse noise voltages range from 10 to 40mV in magnitude and 40 to 70 μ secs in duration [Kap89][Wer89]. The details of the noise varies from one situation to the other.

There have been several methods proposed to mitigate the effects of the impulse noise. One such method is the use of smear and desmear filtering proposed by[Ber87]. The desmearing filter reduces the amplitude of the impulses by spreading them in time. The signal distortion introduced by this in the far end signal is compensated by using a complementary smearing filter at the transmitter. Another method proposed to counter the impulse noise is the shifting of the transmitted signal spectrum above the frequency band where most of the impulse noise is present. As the impulse noise is believed to be concentrated in the frequency band below 40 kHz, a QAM method that translates the signal spectrum above 40 kHz is proposed in[ATT89][Wer89]. Even though this incurs higher attenuation of the received signal, it is claimed that the reduction in the impulse noise more than compensates for the higher attenuation of the signal.

2.3 Multichannel transmission

In this approach the channel bandwidth is divided into a number of parallel subchannels with very narrow bandwidth and data is transmitted on these independent channels[CTC90]. The number of bits/symbol on a subchannel is allotted depending on the SNR available on that particular channel. It has been shown[WE71] that the DFT(discrete fourier transform) can be used to obviate the need for narrow bandpass filters. The input data is divided into blocks of suitable length and mapped into the frequency bins of the DFT. The inverse DFT of these frequency samples are trans-

mitted serially on the channel. The ISI between the received signals is equalized by a decision feedback equalizer. The equalizer output is formed into blocks of the same length as that of the transmitter. Then the DFT of this block gives the frequency domain transmitted sequence which is demapped to obtain the information sequence. A schematic diagram of this scheme is shown in Figure 2.5. Trellis coding can be concatenated with this transmission scheme[RC87]. In this case the encoder output is mapped into the frequency bins. A Viterbi receiver is used instead of a demapper to decode the information bits.

It has been shown[TCDC89] that the multichannel approach can achieve a throughput of 1.544 Mbits/sec on most of the CSA loops. But with a required additional noise margin of 12 dB (to account for implementation and impairment uncertainties[MV89]) the achievable data rate is reduced by a great extent. Under this condition even with a powerful trellis code having 5.5 dB coding gain, the data rate was estimated to be about 695 Kbits/sec on the worst case CSA loop. A drawback of this scheme is the delay through the link. As all the N samples of the IDFT/DFT pair have to be received before any decoding takes place, this delay could be a problem in some applications. This scheme is less susceptible to impulse noise because the data is detected by averaging the noise sequence over a long interval. But any single tone interference is more damaging in this scheme. Optimization of the bit allocation to the frequency bins requires the knowledge of the impulse response of the channel at the transmitter which introduces additional complexity in the implementation.

2.4 Trellis precoding

The trellis precoding technique[EF90] uses the knowledge of the channel impulse response at the transmitter to modify the output of the conventional trellis code in such a way that the coding gain that is achievable on an ideal channel is maintained

Figure 2.5: Multichannel transmission system

on the nonideal channel as well. The decoding complexity is the same as that for an ideal channel. A combination of trellis coding and Tomlinson precoding was reported in [FC89] for a simple (1-D) channel. The transmitted power is limited by the modulo operation in the Tomlinson precoder.

A generalization of this to any linear dispersive channel with impulse response $h(D)$ was described in [EF89]. It was noted that a simple modification of the trellis coding with Tomlinson precoding does not achieve shaping gain. The shaping gain [For89] was observed to be easily achievable after 3 or 4 dB of coding gain. To obtain shaping gain, the filtered trellis code (filtered by the inverse of the channel response) is modified by a shaping trellis code [EF89]. This increases the complexity of the transmitter by requiring a minimum distance decoder of the filtered trellis code. But the coding gain and the shaping gain are additive and hence an improved performance is obtained. One disadvantage with this technique is that the transmitted symbols are not one of the 'm' levels in the conventional PAM system. They can assume continuous values. This would make the complexity of the echo canceller increase drastically, as the convolutions would have to be done on multibit numbers. An IIR implementation of the echo canceller would be more appropriate for this technique. But the adaptation of the IIR filter is much more difficult than the FIR filter. An advantage is that the decoder complexity is the same as that on an ideal channel. This method also requires knowledge of the channel response at the transmitter.

2.5 QAM transmission

The QAM technique has been proposed for high bit rate DSL in [ATT89][Wer89]. One of the reasons cited for using QAM was the avoidance of the impulse noise which was claimed to be concentrated in the low frequency region. Other studies [Kap89] indicated, however, that impulse noise energy also exists at higher frequencies. The

narrow bandwidth occupied by the QAM transmission scheme with large signal constellation results in a shorter impulse response and the echo response. Thus the implementation of the DFE and the echo canceller is simplified. But the shift in the signal spectrum to higher frequency incurs larger amount of crosstalk which is a major impairment in the DSL.

An explicit carrier generation is not necessary for QAM transmission. It has been shown that the information bits can be filtered by a Hilbert pair $\{h(t), \tilde{h}(t)\}$ and then added to obtain the equivalent of a carrier modulation. This technique is called carrierless AM/PM[Fal87][SWZ90]. An advantage of this technique is that the transmitted signal spectrum can be optimized by choosing a suitable filter $\{h(t)\}$. A trellis coded 128 QAM system is being proposed for 1.544 Mbits/sec data rate on CSA loops. Other studies[YFS90][Pot90][Sis90] indicate that there is no significant advantage to using the QAM scheme for HDSL. This is mainly due to larger crosstalk and smaller far end signal power. With no crosstalk, the QAM was shown to be superior to the PAM system.

2.6 Coordinated transmission

Another method that exploits the correlation between the noise voltages on a pair of loops is proposed for high bit rate transmission on loops[Lec90]. In this scheme two pairs are used to achieve 1.544 Mbits/sec data rate in each direction. The signal levels on the two pairs are modified according to the correlation coefficient of the noise voltages on these pairs. It has been shown that there is a 50% chance that a processing gain of 1.5dB can be obtained by exploiting the correlation between the noise voltages. The transmitted symbols are a vector of two elements. There is very little additional complexity in the implementation of this scheme called coordinated transmission.

2.7 Summary

The transmission range and the data rate of the high bit rate system have been described in this chapter. The impairments limiting the performance of the digital transmission system at this rate have been discussed. Various schemes to improve the range and/or the rate of HDSL, being considered by researchers in this area were described and their relative merits discussed. We observe that a cost effective solution meeting the performance requirements is needed in this application.

Chapter 3

Equalization Under Cyclostationary Interference

3.1 Introduction

Equalization is normally done to shape the received signal spectrum so that the interference from adjacent transmitted symbols are minimized[LSW68]. This could take the form of a linear equalizer or a decision feedback equalizer (DFE). In the digital subscriber loop system the DFE is used because it has a better performance compared to the linear equalizer due to reduced noise enhancement. The general form of a DFE consists of a forward filter and a feedback filter. The forward filter suppresses any precursor intersymbol interference and the feedback filter cancels the postcursor ISI by using the detector decisions.

Generally the channel noise is assumed to be stationary in the equalizer design studies. But in the digital subscriber loop the major component of channel noise is due to near end crosstalk from other loops, carrying similar data at the same rate, in the same cable. This crosstalk has been shown[FLF89] to have a cyclostationary (pe-

riodically time varying power) statistics, if all the disturbing transmitters are aligned in phase at the central office. It has been shown[Jos88] that this cyclostationary characteristics can be used to advantage in the equalizer design to gain some noise immunity .

The forward filter in a decision feedback equalizer can be realized by sampling the received signal at the rate $1/T$ and filtering it by a transversal filter with a tap spacing of T sec. Performance of such a T spaced forward filter has been shown[Qur85] to be very sensitive to the timing phase. This could be more so in the cyclostationary interference case with the instantaneous power of the interference varying periodically with the baud period. On the other hand a fractionally spaced forward filter can synthesize the best adaptive filter that minimizes the mean squared error[GW81]. Signal-to-noise ratio improvement of the order of 5 to 10 dB seems possible[Abd89] with a fractionally spaced $(T/2)$ equalizer.

We obtain these optimal fractionally spaced forward filters for different loop configurations and use them as front end receive filters for the sequence estimators considered in this thesis.

Though better performance was shown to be possible under cyclostationary crosstalk, it would be of interest to evaluate the sensitivity of the equalizer performance to other system parameters. In addition to the crosstalk there is also certain amount of white noise on the channel. The level (power) of the white noise was found to have a significant effect on the degree of cancellation of the cyclostationary crosstalk. The SNR of the fractionally spaced equalizer under different noise levels are obtained in this chapter.

The degree of cyclostationarity of the crosstalk interference depends on the lowpass filter bandwidth at the receiver front end. The effect of the bandwidth of the receive filter on the cancellation of crosstalk is discussed in this chapter.

Figure 3.1: Fractionally spaced equalizer system

3.2 Fractionally spaced forward filter for DFE or sequence estimator

The problem of finding the optimum fractionally spaced forward filter is described in [Abd89]. A brief review of this procedure is given here to introduce the symbols and notation involved. The block diagram of the digital transmission system with a fractionally spaced equalizer is shown in Figure 3.1.

The received signal can be expressed by the equation

$$r(t) = \sum_{-\infty}^{\infty} a_k \Phi_0(t - kT) + \sum_{l=1}^N \sum_{k=-\infty}^{\infty} b_{kl} \Phi_l(t - kT - \Omega_l) + n(t) \quad (3.1)$$

where

$n(t)$ is the additive white gaussian noise of power spectral density $N_0/2$,

a_k 's are the far end data symbols to be detected,

$\Phi_0(t)$ is the channel impulse response,

b_{kl} 's are the data symbols from the l^{th} interferer,

Ω_l is the phase of the l^{th} interferer,

$\Phi_l(t)$ is the impulse response of the crosstalk path from the l^{th} interferer, and

N is the number of interferers.

The forward filter coefficients $W_i, i = 0, 1, \dots, N_f$, and the feedback coefficients $f_i, i = 1, 2, \dots, N_b$ are chosen such that the mean squared error at the input of the detector is minimized. The optimum coefficients are obtained by solving a set of linear equations as described in [Abd89]. A corresponding expression for the minimum mean squared error has also been derived in [Abd89]. The crosstalk interference is cyclostationary when the phase Ω_l of all the interferers are the same and stationary when they are uniformly distributed over the baud interval. The performance (SNR of the DFE) of the optimum fractionally spaced receive filter with $N_f = 20$ and $N_b = 60$ was calculated for the worst case CSA loop (9Kft, 26 Gauge) with 49 interferers¹ with identical pulse shapes and independent data under cyclostationary and stationary noise conditions. In both cases the crosstalk is 20 dB, and the white noise is 45 dB, below the far end received signal level at the input of the forward filter. The results are shown in Table 3.1. It can be observed that a significantly better performance can be obtained if the interferers are phase aligned to the same clock.

The equalized discrete-time end-to-end responses for this loop under the same conditions are shown in Figure 3.2. The shape of the equalized channel response under stationary crosstalk looks very similar to that of the loop response itself, while it is modified significantly under cyclostationary crosstalk since the forward filter is

¹The cables used in the loop plant typically consist of 50 twisted pairs.

Noise characteristics	SNR in dB
stationary	14.05
cyclostationary	25.82

Table 3.1: SNR of FSE under stationary and cyclostationary interference

shaping the crosstalk response to suppress it at the equalizer sampling instants as well as suppressing ISI.

It was observed in [Abd89] that if the number of forward taps is reduced below 20, there is a sharp decline in the performance of the equalizer, while the dependence on the number of feedback taps is not so significant.

The total noise spectrum (both crosstalk and white noise) at the output of the forward filter is shown in Figure 3.3 for the stationary and cyclostationary cases. It can be observed that the noise is very nearly white, implying optimality of the maximum likelihood sequence estimator that follows the forward filter. The deviations at certain frequencies is due to the limited number of tap coefficients used in the forward and feedback filters. By increasing the number of taps, one can obtain a white spectrum for the output noise.

3.3 Effect of background white noise level on the equalizer performance

The computations described in the previous section were performed with a white noise power spectral density² of 10^{-14} watts/Hz. The mean squared error at the

²This noise spectral density corresponds to a signal-to-white noise ratio of 45 dB. A termination impedance of 130 ohms is assumed.

Figure 3.2: Equalized pulse responses

Figure 3.3: Total noise spectrum

White noise level	SNR	
	Cyclostationary	Stationary
$1 \cdot 10^{-14}$	22.49	14.04
$2.5 \cdot 10^{-15}$	25.82	14.05
$2.5 \cdot 10^{-16}$	29.75	14.05

Table 3.2: Effect of white noise level on the performance

input of the detector consists of residual ISI, forward filtered crosstalk and white noise. As the optimum equalizer tries to minimize the total mean squared error, the level of the white noise, though much lower than the crosstalk level, was found to have a significant effect on the equalizer performance under cyclostationary crosstalk interference conditions. This is not so in the stationary interference case because the ability of the forward filter to cancel the stationary crosstalk is very limited. It was observed in [Abd89] that the performance of a fractionally spaced equalizer is relatively better than that of a baud rate equalizer even under stationary crosstalk condition.

The performance (SNR of the DFE) of the fractionally spaced forward filter for three different white noise levels were calculated and shown in Table 3.2. The corresponding results for the stationary noise case are also shown in the table.

The results indicate that the performance under cyclostationary crosstalk is very sensitive to the level of the background white noise, even though the level of the white noise is much lower (20 dB below crosstalk) than that of the crosstalk.

3.4 Effect of the lowpass receive filter bandwidth on the equalizer performance

The bandwidth of the lowpass filter at the receiver front end has a significant effect on the performance of the fractionally spaced equalizer under cyclostationary interference conditions. The statistics of the synchronous pulse amplitude modulated signal

$$x(t) = \sum_{n=-\infty}^{\infty} a_n \Phi(t - nT) \quad (3.2)$$

were observed to have cyclostationary behaviour by Bennett[Ben58]. This means that the mean $m_x(t)$ and autocorrelation function $R_{xx}(\tau)$ of $x(t)$ are periodic with period T . By using a Fourier series representation of the autocorrelation function $R_{xx}(\tau)$, Gardner[GF75] relates the degree of cyclostationarity to the magnitude of the components at the fundamental and higher harmonics of the Fourier series representation. When the random process $x(t)$ is strictly band limited to $1/2T$, only the energy around zero frequency contributes to the total power of the random process and $x(t)$ becomes stationary[CGS83]. Hence a cyclostationary random process strictly band limited to $1/2T$ is stationary. In the digital subscriber loop system the near end crosstalk is bandlimited by the transmit and receive filters. In this study identical low pass filters at the transmitter and the receiver are used.

The performance of the optimum fractionally spaced filter for three different lowpass filter bandwidths (6th order Butterworth filter with 3 dB cutoff frequency at 150 kHz, 200 kHz, 250 kHz) was calculated. In all these cases it was assumed that the NEXT is from 49 identical, phase synchronized crosstalkers. The results shown in Table 3.3 indicate that when the interference is band limited to $< 1/2T$, ($1/2T = 200$ kHz), the performance is approximately the same as for the stationary case, but better performance is obtained by increasing the lowpass filter bandwidth. The reason that the performance is better with increasing bandwidth is that the degree of

Bandwidth Khz	SNR	
	Cyclostationary	Stationary
150	14.05	13.88
200	25.82	14.05
250	40.45	13.96

Table 3.3: Effect of bandwidth on the performance

cyclostationarity increases with the bandwidth of the lowpass filter. The optimum forward filter then suppresses the crosstalk much more effectively in this case. In the stationary crosstalk case any increase in the bandwidth of the lowpass filter would degrade the performance of the system. Also, the SNR at a bandwidth of 150 kHz is less than that at 200 kHz. This is due to reduction in the signal energy available at the lowpass filter output when the bandwidth is 150 kHz. This reduction in the signal strength may be more than the reduction in the crosstalk power. As the bandwidth is increased to 200 kHz, there is a small improvement in the performance. At larger bandwidths the amount of crosstalk³ overwhelms the marginal increase in the signal power and hence the performance deteriorates.

The equalized crosstalk path spectra are shown in Figure 3.4 (under the assumptions mentioned previously) for the three bandwidths considered along with the stationary crosstalk equalised spectrum. The similarity between the stationary case and the 150 Khz bandwidth case is noticeable. Corresponding pulse responses for the four cases are shown in Figure 3.5. The effective suppression of the cyclostationary crosstalk with increasing bandwidth is clearly evident. The response for the 250 Khz bandwidth case is too small to be observed on this scale.

³The crosstalk path power transfer function increases exponentially ($f^{3/2}$) with frequency.

3.5 Conclusion

The advantages of a fractionally spaced equalizer (FSE) for a digital subscriber loop system was described. A method for finding the optimal FSE was reviewed. The performance (SNR) of such an optimal equalizer under stationary and cyclostationary interference conditions was determined. It is observed that the performance under cyclostationary interference is significantly better than that of the stationary interference case. This is mainly due to the ability of the FSE to cancel the cyclostationary interference. While the performance improvement achievable using a FSE has been reported earlier[Abd89], in this study we investigated the sensitivity of the performance to other system parameters. The SNR, under cyclostationary interference, was shown to be very sensitive to the level of background white noise. The degree of cyclostationarity was related to the bandwidth of the transmit and receive filters. The performance of the FSE was found to increase with bandwidth.

Figure 3.4: Equalized crosstalk path power spectra

Figure 3.5: Equalized crosstalk path pulse responses

Chapter 4

Trellis Coded Modulation for Subscriber Loop Channel

4.1 Introduction

Trellis coded modulation was initially proposed by Ungerboeck[Ung82] to gain noise immunity for digital transmission on band limited channels. On an ideal AWGN channel, without any linear distortion, coding gains of the order of 3 to 6 dB were obtained using codes with 4 to 256 states. When these codes are used on dispersive channels, there is a considerable degradation in the coding gain. This is due to reduction in the Euclidean distance between the received signal sequences. Ketchum[Ket87] has shown that on $(1 - D)$ channel these codes could incur upto 3 dB loss in the coding gain. On subscriber loop channels with severe high frequency attenuation similar losses in performance can be expected. Trellis coded modulation technique has been applied to voice band data transmission system in[CE89] and to partial response systems such as $(1 - \alpha D)$ in[EQ89]. Significant improvement in the performance (i.e reduction in SNR for a particular error rate) has been reported. The performance of such codes on the subscriber loop channel is determined by simulation in this chapter.

The most important parameter in a coded system is the minimum Euclidean distance d_{min} between the received signal sequences. The performance of trellis codes of different complexity (*i.e.*, number of states) on the subscriber loop channel are evaluated in this chapter by determining the d_{min} for these codes. The equalized responses obtained in the previous chapter are used in these calculations. This quantity is indicative of the performance of a Viterbi receiver operating in the presence of white gaussian noise. The results indicate that the severe distortion of the subscriber loop channel necessitates the use of codes with large numbers of states to obtain reasonable coding gain.

The coded systems implicitly assume a sequence estimation receiver. The receiver architecture chosen for this study consists of a fractionally ($T/2$) spaced adaptive DFE(decision feedback equalizer) forward filter followed by a sequence estimator. The fractionally-spaced DFE forward filter acts as a pre-whitening filter for the sequence estimator[For72][CD89] and has been shown to suppress cyclostationary crosstalk [Abd89].

The complexity of the maximum likelihood sequence estimator in a coded system for ISI channels is proportional to the product of the number of coder states and the channel states. As the number of channel states increase exponentially with the ISI length, a full Viterbi algorithm sequence estimation receiver is not practicable for the subscriber loop channels with ISI lengths of the order of few hundred baud intervals. Hence, reduced state sequence estimators such as the M-algorithm[MS86][JF90] or DFSE[DHH89] (Decision Feedback Sequence Estimator) are considered in this study. These receivers provide a significant portion of the performance achievable with a full Viterbi algorithm sequence estimator at much less complexity. Preliminary study presented in Chapter 5 indicates that it may be possible to integrate an 8 state M-algorithm based sequence estimator on a single VLSI chip. Simulations to determine the performance (range of the loops) of the trellis coded system using such receivers are presented in this chapter.

4.2 Trellis coding

Conventional coding techniques, such as block coding and convolutional coding [VO79], gain noise immunity by adding redundant bits to the transmitted data. The redundant bits are added in such a way that the Hamming distance between the received signal sequences is maximized. The added bits increase the bit rate on the line and hence the bandwidth of the signal for a given information rate. For channels with severe high frequency attenuation, such as digital subscriber loops, an increase in the bandwidth is undesirable. Also the crosstalk interference would increase with the signal bandwidth as the crosstalk path transfer function has a high pass nature.

Trellis coded modulation introduces redundancy by increasing the number of levels that are transmitted, thus avoiding the undesirable effects of bandwidth expansion. For example, a simple trellis code with four states and eight signal levels can be used to transmit two bits of information per symbol. The encoder of such a code is shown in Figure 4.1. It consists of a convolutional encoder that transforms the two information bits into three encoded bits and a signal mapping rule that maps the encoded bits into one of eight signals according to the set partitioning rule developed by Ungerboeck[Ung82]. The time progression of the transmitted symbols of the code can be represented by the trellis diagram shown in Figure 4.2. The coding gain of the trellis code is defined by

$$\text{coding gain} = \frac{\frac{\text{minimum squared distance of coded system}}{\text{average power of coded symbols}}}{\frac{\text{minimum squared distance of uncoded system}}{\text{average power of uncoded signal}}} \quad (4.1)$$

For the code shown in Figure 4.2 the coding gain can be calculated to be

$$\begin{aligned} \text{coding gain} &= \frac{\frac{36}{21}}{\frac{4}{5}} \\ &= 2.1 \text{ or } 3.3 \text{ dB} \end{aligned} \quad (4.2)$$

where the average power of the 8 level coded symbols ($\pm 1, \pm 3, \pm 5, \pm 7$) is 21 and the

Figure 4.1: A 4 state trellis encoder

average power of the uncoded symbols ($\pm 1, \pm 3$) is 5. More complex codes with large number of states provide larger coding gain.

4.3 Minimum distance of the coded system on an ISI channel

The coding gains described in the previous section are obtained on an ideal channel without any dispersion. When these codes are used on an ISI channel the distance structure of the received signal sequences are modified by the channel impulse response. For example, the received signal sequences for the same code on a (1-D) channel are shown in Figure 4.3. The squared minimum distance on the ISI channel is 20 as compared to 36 on an ideal channel. This indicates that codes designed for an ideal channel could incur large penalty in coding gain on an ISI channel. In addition to the minimum distance the number of sequences at this distance, called path

Figure 4.2: 4 state trellis diagram

multiplicity, is another important parameter in determining the probability of error.

Unlike a convolutional code, the minimum distance d_{min} between the received sequences in a trellis coded system depends on the transmitted sequence. The minimum distance of the received sequences can be expressed as

$$d_{min}^2 = \min_{\epsilon_0 \neq 0} \left\| \sum_{n=0}^N \epsilon_n h_n \right\|^2 \quad (4.3)$$

where the minimum is over all the error sequence $(\epsilon_1, \epsilon_2, \dots, \epsilon_N)$ and for all N , and $\{h_n\}$ are the impulse response coefficients of the channel.

In this section we compute bounds on minimum distance, which are indicative of the performance of ideal maximum-likelihood sequence estimation receivers. The computational time required to calculate d_{min} increases exponentially with the length of the error sequence. Also, in the trellis coded system the minimum distance must be evaluated for all possible transmitted sequences. There are several methods proposed in the literature, to evaluate the d_{min} . It was suggested[She86] that the Viterbi algorithm can be used on a trellis consisting of combined coder and channel states to determine the minimum distance. As the number of states to be considered is very large (of the order of $N * m^L$, where N is the number of coder states, $2 * m$ is the number of symbols in the signal set and L is the length of the ISI) this is not practical. A modified form of the transfer function approach, which is commonly used in the case of convolutional codes, was proposed in[Big84]. This method also requires consideration of a product trellis with a large number of states.

Messerschmitt[Mes73] has shown that a sequence of upper and lower bounds for the d_{min} can be calculated by increasing the error sequence length and the d_{min} can be determined when these bounds are sufficiently close. The bounds are calculated following the approach presented in[LM88].

$$d^2 = 1/2[(3-7)^2 + (5-7)^2 + (7-5)^2 + (3-7)^2]$$

$$= 20$$

Figure 4.3: ISI Trellis

Lower bound

$$d_{min}^2 \geq \min_{\substack{\epsilon_n \\ \epsilon_0 \neq 0 \\ \epsilon_n = 0, n < 0}} \sum_{n=1}^N \left\| \sum_{j=0}^L \epsilon_{n-j} h_j \right\|^2 \quad (4.4)$$

Upper bound

$$d_{min}^2 \leq \min_{\substack{\epsilon_n \\ \epsilon_0 \neq 0 \\ \epsilon_n = 0; n < 0, n > N}} \sum_{n=0}^{N+L} \left\| \sum_{j=0}^L \epsilon_{n-j} h_j \right\|^2 \quad (4.5)$$

where $\{h_j\}$ is the equalized channel response of length 'L' and 'N' is the length of the error sequence considered.

It has also been found[Mes73] that in an uncoded system, relatively short error sequences create minimum distance error events for highly dispersive channels. This was found to be true in the coded system as well.

A computationally efficient algorithm shown in Figure 4.4 was developed to evaluate the minimum distances for many random transmitted sequences. Exhaustive searching of all possible incorrect sequences is avoided by pruning parts of the error tree when the unmerged error event has a larger distance than that of a merged error event. This prevents the exponential increase in the computation time as the error sequence length is increased. Error sequences up to a length of 15 symbols were considered. These bounds were calculated for a 4 state code shown in Figure 4.1 and a 64 state code[CM84]. The equalized channel response used in the minimum distance calculation is that shown in Figure 3.2 for stationary crosstalk with the power $\sum |h(i)|^2$ normalized to unity. This corresponds to a fractionally spaced forward filtered response of a 9 Kft, 26 Gauge loop.

Figure 4.4: Flowchart of the algorithm

The distance spectrum in Figure 4.5 shows the frequency of occurrence of error events with squared distance $d_{\min}^2$ (lower bound) in a trial of 100 different (random) transmitted sequences for the two codes. Figure 4.6 shows the results for the upper-bound. From these figures we observe that the lower and upper bounds for the 4 state code are very close to each other in a majority of the cases, while there is some difference between these bounds in the case of the 64 state code. Also, the 64 state code tends to have a higher proportion of the sequences with larger than minimum distance. It should be noted that while these results yield some insight, they do not reveal the performance of a practical trellis coded modulation system, which would have to use reduced state decoding techniques. The performance of these two codes are compared by simulation in the following sections.

The squared minimum distance of an uncoded system for the same channel was found to be 8.02. The squared minimum distance of the uncoded system is scaled by $P_{\text{coded}}/P_{\text{uncoded}}$ so that the transmitted power is the same in both cases. It should be noted that the minimum distance of the uncoded system does not depend on the transmitted sequence.

The distance bounds of the two codes on the same channel with the forward filter optimized under cyclostationary interference were also obtained. The equalized response shown in Figure 3.2 for the cyclostationary case is used in these calculations. The lower and upper bounds in this case are shown in Figure 4.7 and Figure 4.8 respectively.

4.4 Combined equalization and decoding

There are three receiver architectures possible for equalizing and decoding the trellis coded signals on ISI channels. One is to equalize the channel using a linear equalizer and then use a trellis decoder to detect the received data sequence. The

Figure 4.5: d_{min} lower bound (stationary crosstalk)

Figure 4.6: d_{min} upper bound (stationary crosstalk)

Figure 4.7: d_{min} lower bound (filter optimized for cyclostationary crosstalk)

Figure 4.8: d_{min} upper bound (filter optimized for cyclostationary crosstalk)

linear equalizer causes the equalized channel to be a Nyquist channel with no ISI. This method has been shown[CE89] to degrade the performance of the system because of the noise enhancement.

The second is a concatenation of a DFE and a trellis decoder as shown in Figure 4.9. The DFE removes all the ISI and the trellis decoder acts on the encoder trellis to decode the transmitted symbols. The performance of this coded system was shown[Jos88] to be worse than a simple uncoded system with a DFE receiver due to error propagation in the DFE, as the decisions made by the DFE are unreliable because of the reduced distance between the symbol levels in the coded system.

The third is a maximum likelihood sequence estimator (MLSE) for the combined code and channel trellis as shown in Figure 4.10. The number of states in the MLSE of the combined trellis is equal to $N \cdot m^L$, where N is the number of states in the encoder, $2 \cdot m$ is the number of channel symbols and L is the length of the equalized channel response. Even for small values of N, m and L , the number of states to be considered is large. A Viterbi algorithm based MLSE is not practical because of the large number of states. Hence reduced complexity sequence estimators such as RSSE or M -algorithm must be used.

One way of reducing the complexity of the MLSE is to use a linear equalizer [QN73][FM73] to reduce the length of the ISI, hence the number of states, and use the Viterbi algorithm on this reduced trellis. Though the noise enhancement would be less compared to the case when ISI is completely removed by linear equalizer, for the subscriber loop channel with severe distortion the noise penalty is still substantial. Lee and Hill[LH77] suggested that the DFE can be used to cancel the tail of the ISI, while a Viterbi receiver could take care of the initial few baud periods of the impulse response. The delayed decisions of the Viterbi receiver, which are more reliable, can be used in the DFE. Even in this case any error made by the detector would tend to induce more errors.

A method of minimizing the error propagation of the DFE is to incorporate the decision feedback in the metric calculation of the sequence estimator itself. This approach called Decision Feedback Sequence Estimation (DFSE) was proposed for uncoded transmission systems in [DHH89]. Extension of this method to the coded system with a structured way of reducing the channel states was proposed in [EQ89]. This method called Reduced State Sequence Estimation (RSSE), reduces the number of states in the receiver by combining the channel states using set partitioning principles. Then a Viterbi algorithm is used to search the reduced trellis for decoding the data symbols. A special case of the RSSE was observed to be the DFSE receiver developed by Duel and Heegard [DHH89]. The minimum number of states possible in a RSSE or DFSE receiver is equal to the number of states in the encoder. As discussed in the previous section trellis codes with large number of states (for example 64 states) may be necessary to obtain reasonable coding gains on digital subscriber loop channel. It may not be possible to realize such complex receivers in current VLSI technology.

The M-algorithm based sequence estimator overcomes this problem. In this receiver only M (say 8) states, representing M data sequences closest to the received sequence, are retained irrespective of the number of states in the encoder. At each baud interval, the M retained sequences are extended into $m * M$ new sequences and the best M paths are selected again. Preliminary estimates, described in a later chapter, indicates that an 8 state receiver may be implemented in a VLSI with the state-of-the-art VLSI technology. Hence an 8 state M-algorithm based sequence estimator is considered for further study in this chapter.

Figure 4.9: Receiver with DFE and trellis decoder

4.5 Simulation set up

Simulations were performed to evaluate the performance (in symbol error rate) of the trellis coded modulation system for high bit rate digital subscriber loop transmission systems. Figure 4.10 shows the system block diagram that was used in the simulation. The parameters of the system are as follows.

data rate : 800 Kbits/sec

baud rate : 400 Ksymbols/sec

encoders : 4 state trellis code shown in Figure 4.2

64 state trellis code described in[CM84]

loop characteristics : 26 gauge loops of different length

receiver filter : fractionally spaced ($T/2$) forward filter
 with 21 taps
 number of taps in
 the equalized response : 60 taps
 receiver : 8 state M-algorithm receiver
 : Decoder delay = 100
 noise characteristics : near end crosstalk (49 interferers, 99% worst case);
 stationary and cyclostationary statistics;
 The interferers are identical and synchronized
 in the cyclostationary case;
 -110 dBm/Hz White noise
 noise margin : 12 dB (amount by which the above level of
 crosstalk is scaled up)[MV89]

The symbol error rate at different loop lengths were obtained by running the simulation with 10^5 transmitted symbols.

4.6 Simulation results

Probability of symbol error vs loop length for two different trellis codes are shown in Figure 4.11 under stationary crosstalk conditions. Note that the loop length decreases from left to right. The stationary behaviour of the crosstalk was simulated by distributing the 49 crosstalkers uniformly among the eight sampling phases within a baud interval. It was observed in Section 4.3 that more complex codes with large number of states tend to have larger minimum distance and hence better performance. Even though the range of the loops for the two codes are very close at the high error

rate of 10^{-4} , the larger slope of the error curve for 64 state code compared to 4 state code in Figure 4.11 indicates that the performance of the 64 state code would be better at low (10^{-7}) error rate. Also shown in Figure 4.11 is the symbol error curve for an uncoded system with a simple DFE receiver. The large error rate for the DFE receiver was found to be due to severe error propagation. From an extrapolation of the curves in the figure, it is expected that the coded systems could provide an additional 1 km loop coverage (of course with at least 8 times more complex receiver) at an error rate of 10^{-7} . It should be noted that even the coded systems do not meet the transmission range requirements (CSA loop range, which is 9 Kft of 26 gauge wire or 12 Kft of 24 gauge wire) of the HDSL systems being proposed currently.

Similar results under cyclostationary crosstalk conditions are shown in Figure 4.12. All the 49 crosstalkers are identical and synchronized to the same clock (with a sampling phase of zero) in this case. The loop coverage under cyclostationary noise is significantly larger than that under stationary noise conditions for all the three transmission schemes, but the relative performance remains the same. Furthermore, there is a more significant performance difference between the 4 and 64 state coded systems. The performance improvement is mainly due to the cancellation of crosstalk interference by the fractionally spaced forward filter. These results were presented in [MFK90].

The performance of codes with intermediate number of states (*i.e.*, between 4 and 64 states) would lie between the error curves of these two codes. From an implementation point of view there is no disadvantage to using a more complex code because the hardware complexity of the encoder is very small compared to that of the decoder.

To compare the performance of a DFSE receiver of the same complexity as that of the M-algorithm receiver considered above, simulations were performed with an 8 state trellis code and an 8 state DFSE receiver. The results are shown in Figure 4.11. It can be seen that the DFSE receiver performance is worse than that of the M-

algorithm receiver. Also, the performance of an uncoded system with an 8 state M-algorithm receiver is shown in Figure 4.11. It can be observed that the coded systems provide some additional range as compared to the uncoded system.

4.7 Conclusion

The principles of trellis coded modulation was described in the chapter. The coding gain of these codes on AWGN channels was illustrated by a simple example. The asymptotic performance of the code was related to the minimum distance d_{min} of the received signal sequences. The degradation in the minimum distance on dispersive channels was illustrated by an example. A computationally efficient algorithm was developed to determine the d_{min} for codes of various complexity on severely dispersive channels such as subscriber loops. It is observed that more complex codes with large number of states (for example 64 states) are necessary to obtain reasonable coding gains on subscriber loop channels. As the complexity of a full Viterbi receiver is enormous, reduced complexity sequence estimator such as M-algorithm receivers are considered in this study.

The tasks of equalization and decoding are combined in the M-algorithm receiver. The performance of the trellis coded modulation system, with a 64 state code and an 8 state M-algorithm based receiver, was determined by computer simulation. It is novel to have fewer receiver states than the number of encoder states. The simulations were performed under stationary and cyclostationary interference conditions. It is observed that with a noise margin of 12 dB, even the coded systems do not seem to have the range required for the worst case CSA loop (9 Kft, 26 Gauge) under stationary noise conditions, while loops longer than 9 Kft can be covered under cyclostationary noise. These results are compared to the performance of an uncoded system with a simple DFE receiver.

Figure 4.10: Combined equalization and decoding receiver

Figure 4.11: Performance under stationary crosstalk

Figure 4.12: Performance under cyclostationary crosstalk

Chapter 5

VLSI Design Considerations of the Sequence Estimator

5.1 Introduction

An efficient VLSI architecture for the sequence estimator is explored in this chapter. Conventional receivers such as those using a DFE do not meet the performance requirements of the high bit rate DSL (HDSL) systems. Also, trellis coded systems make the use of sequence estimators mandatory. Obviously the complexity of the sequence estimation receiver is much more than that of the DFE receiver. The operation of an M state sequence estimator can be thought of as M independent DFE receivers. The computation involved in the sequence estimation can be divided into three steps.

- 1) Branch metric calculation.
- 2) Path metric calculation and survivor selection.
- 3) Path memory updating.

There have been several papers on these topics in the literature, each addressing these three steps independently assuming that other steps are much simpler than the one considered. A systolic array structure for branch metric calculation was proposed in [PG88]. This implementation requires systolic array cells having a $\log_2 m * B$ bit multiplier, where ' m ' is the number of symbols in the signal set and ' B ' is the wordlength of the impulse response coefficient. For an ISI length of ' L ' symbols, an array of $m * L$ processing elements are required in this method. In the case of digital subscriber loop channels with an equalized response length ' L ' in the range of few tens, it may not be possible to integrate all the processing elements on the same chip. Also other components of the sequence estimator such as path memory, path metric comparison unit etc., require substantial silicon area.

The path metric comparison and survivor selection procedure, also known as Add-Compare-Select (ACS) in the literature, of the sequence estimator involves interconnection of processing elements that are not nearest neighbours. This means that a lot of silicon area is required for the interconnecting wires. An efficient placement of processing elements that minimizes the requirements for the interconnection was proposed in [GS86]. But this paper does not address issues related to the branch metric and path metric computation.

A systolic array based survivor selection method is discussed in [CY]. It was concluded that in a straightforward implementation of the Viterbi algorithm, the processing elements would waste many cycles doing no useful work. This is due to the sparseness of the connectivity between the states of a sequence estimator. To improve the efficiency of the processing elements, it was suggested that a fully connected trellis with branch metric corresponding to $\log_2 M$ (M is the number of states) baud intervals could be used in the sequence estimator. In this method also it was assumed that the branch metrics were computed by other means.

The computations involved in the Viterbi algorithm implementation of the sequence estimator is sequential in the sense that all the computations in a baud interval should be completed before the computations for the next baud interval could commence. In high speed communication systems this could be a severe constraint. A parallel approach breaking this ACS- bottleneck was proposed in[FM89]. There is a substantial increase in the hardware required by this approach.

All the approaches mentioned above are only for Viterbi algorithm based sequence estimator. An additional processing stage involved in the M-algorithm based sequence estimator is that of duplicate state elimination and path metric sorting. A multiprocessor VLSI architecture was proposed in[MS86] for the implementation of the M-algorithm. This method uses a modified Batcher network[Bat68] for sorting. The duplicate state elimination requires routing of the state parameters through the sorting network. For subscriber loop channel with long memory, this would require a large routing area. We propose a new approach to this problem in this chapter.

As the sorting procedure involves a considerable increase in the complexity, Simmons [Sim90] proposed a method to eliminate this step in the M-algorithm. His approach, called the T-algorithm, uses a threshold for rejecting the less likely paths. All the paths with path metric below the threshold are retained and those having path metric above the threshold are eliminated. Though this method eliminates the sorting procedure, the complexity of the control circuitry increases because the number of retained paths is a random variable. Hence the processing time for each baud interval would vary with the number of retained states.

In this chapter all the three steps are addressed together so as to arrive at an efficient VLSI architecture. To meet the hardware complexity limitations in the VLSI transceiver for an HDSL system, as other components of the system such as echo cancellation, timing recovery, data formatting, etc., have to be integrated on the same chip, the computational hardware is time shared between the three steps mentioned

previously. The branch metric computation may be simplified by using a piecewise linear approach to be described in Chapter 7. Conventional implementation of the path memory updating involves double buffering which doubles the memory required for the storage of the survivor symbols. A new circuit design using master-slave dynamic register cell is proposed to avoid the doubling of the path memory.

As mentioned above the M-algorithm implementation of the reduced complexity sequence estimation involves sorting of the state metric. As the number of clock cycles needed to sort a list of N elements, using a single computational element is of the order of N^2 , a parallel architecture is proposed to reduce the time required for sorting. This necessarily increases the hardware complexity, but the time available for decoding symbols in the HDSL system necessitates the use of a parallel architecture. The silicon area required to implement the sequence estimator in 1μ CMOS technology is estimated.

5.2 M-algorithm for sequence estimation on ISI channels

A brief review of sequence estimation is given below. A sequence estimator consists of a number of states. With each state is associated a path metric and a path memory. The path metric represents a measure of the closeness of the signal based on a hypothesized transmitted (decoded) sequence to the received signal while the path memory stores the data symbols associated with the hypothesized decoded sequence. In an M-algorithm based sequence estimator there are M retained states at each baud interval. For example, in Figure 5.1 there are two retained states. In a binary scheme these two states are extended at time ' n ' to four paths. For an ISI channel the branch

Figure 5.1: Paths considered in the M-algorithm

metric associated with each of those four branches is computed by the equation

$$b_n = (r_n - \sum_{i=0}^L f_i \hat{a}_{n-i})^2 \quad (5.1)$$

where f_i , $i = 0, 1, \dots, L$ are the impulse response coefficients of the channel

$\{\hat{a}_n\}$ are the hypothesized sequences associated with that path

and r_n is the received signal at time t_n .

The path metrics of those four paths are calculated by adding the respective branch metrics to the path metric up to the previous baud interval. In a M-algorithm receiver the two (or M) survivors are selected by first eliminating the less likely paths (*i.e* the paths with larger path metrics) among merging paths, and then choosing the

best two paths (or M paths) from the remaining paths. This involves elimination of duplicate paths and then sorting the remaining paths according to the path metric. Once the surviving paths are selected the path memory is updated by extending the survivor sequence with the symbol in the selected branch.

As the memory is limited in any physical hardware, the oldest symbol in the path memory is discarded and the new symbol is stored by shifting all the symbols in the path memory by one position. The oldest symbol in the path memory of the surviving path with the smallest metric is declared as the decoded symbol. The decoder delay (*i.e.*, the number of baud intervals that the receiver waits before making an irrevocable decision on the received symbol) is equal to the length of the path memory. Efficient VLSI architectures for each of these preceding steps of the M-algorithm receiver for an HDSL channel are described in the following sections.

5.3 Branch and Path metric computation

The branch metric computation involves two operations: one is the convolution of the estimated data sequence with the impulse response of the channel and the other is the squaring as indicated in equation 5.1. The multiplication involved in the convolution is a simple 3 bit by B bit multiplication where the wordlength of the impulse response coefficient is B bits long and the data elements are 3 bits for an 8 level PAM signal set assumed in this study. The hardware required for this multiplication is simple. As the hardware required for multibit multiplication is very large, the squaring operation is done in a serial manner with one operand taken in three bit groups at a time. Thus a B bit *times* B bit multiplication would take $B/3$ cycles. The same hardware that was used for the convolution is used in this case.

A schematic diagram of the branch metric computational unit is shown in Figure 5.2. In a M-algorithm receiver there are M such computational elements working

in parallel. The data symbols are read from the path memory as explained in Section 5.3.3. The impulse response coefficients are read from the register file where they are stored. For long impulse response channels, computation of the branch metric can be simplified by using a piecewise linear approach proposed in Chapter 7. The four registers store the branch metrics of the four paths (corresponding to 2 information bits/symbol) emerging from each state. Adaptation of the coefficients are done simultaneously along with the branch metric computation. The path metric of an extended path is computed by adding the branch metric of the path with the old path metric of the state. The controller generates the necessary control signals for the various multiplexors routing the proper signals to the adder and multiplier.

5.3.1 Duplicate state elimination

In the M-algorithm based sequence estimation, the extended paths from the retained states should be examined for duplicate states and all the paths with larger metric, among the merging paths, must be eliminated. The duplicate states are states with the same encoder and channel states. The encoder state of a path at time ' n ' is the state that the encoder at the transmitting end would have been in, at the corresponding time instant, to produce the hypothesized data sequence of the path under consideration. The channel state of each path is represented by the estimated data sequence $\{\hat{a}_n\}$ of that particular path. This is usually done by sorting the extended paths according to the states and comparing adjacent states for equality. This is practical only for short impulse response length channels and a small symbol alphabet, where all the possible states can be given unique numbers. With an equalized response length of 50 symbol periods and 2 bits/symbol, there are 4^{50} states possible. Therefore giving a unique number and comparing them is impractical. Thus the only way to compare for state equality is to compare the estimated symbols in the path history, symbol by symbol. As every extended path must be compared with every

Branch /Path metric computation

Figure 5.2: Branch metric computation

other path, this sorting time is of the order of $O(N^2)$, where N is the number of extended paths, if it is done sequentially.

A new approach is proposed to reduce the sorting time. Associated with each extended path is a 1 bit register which indicates whether this path is a duplicated path or not. In the coded system, the state comprises the encoder state and the channel state (represented by estimated symbols in the path history). The channel state comparison part can be done during the convolution operation involved in the branch metric computation. As only the branch symbol is different in each path extended from each state, the extended branch symbol and encoder state comparison is done by the duplicate state elimination circuit described below. The channel state comparison is done by having $(M - 1), (M - 2), \dots, 1$ registers for states 1 to M and resetting that particular register when the paths compared differ in any symbol interval. The register arrangement for a simple 4 state receiver is shown in Figure 5.3. In the transmission system considered here there are 4 branches from each retained state. This information along with the branch symbol, next coder state and the path metric are used in the duplicate state marking procedure.

The duplicate state marking circuit consists of an array of 4×4 comparator cells as shown in Figure 5.4. As the paths leaving a particular state will have different branch symbols they need not be compared between themselves as it will not lead to any merging of the paths. Only paths from the other states need to be compared. The state parameters (represented by the result of the channel state comparison (stored in a flip-flop), the encoder state, the branch symbol and the new path metric) from the reference state are input on the vertical bus and the state parameters from the compared state are input on the horizontal bus. The output from each comparator at each cross point is wired-OR column wise as well as row wise. The column outputs are used to set the duplicate state flag of the reference states and the row outputs are used to set the compared state flags. This process is repeated for all the other states by keeping state 1 as the reference state. This requires $(M - 1)$ cycles. Next, state 2 is

State comparison

f_{ij} Flag for comparing path symbols of state 'i' and state 'j'

Figure 5.3: Channel state comparison

used as a reference state and states 3 to M are input as compared states sequentially. This would take $(M - 2)$ cycles. This procedure is repeated until state $(M - 1)$ is used as a reference state. Thus the total number of cycles needed for duplicate state eliminations is

$$(M - 1) + (M - 2) + \dots + 1 = M * (M - 1)/2 \quad (5.2)$$

Thus any merging paths are rejected except the one with the least path metric.

5.3.2 Metric sorting

After the duplicate states are marked, the remaining states should be sorted according to the metric and the best M states should be selected for path extension in the next baud interval. In this application, sorting of the whole list is not necessary because only the best M states are retained for further processing.

The most common method for sorting is called the selection sort. In this method, the lowest metric in the list is found first. Then, that element and the element in the first position are interchanged. The next lowest element is found by starting the search from the second position. When it is found, that element is interchanged with the element in the second position. Using this procedure we need

$$N + (N - 1) + (N - 2) + \dots + (N - (M - 1)) \quad (5.3)$$

cycles to select the best M paths. Here, N is the total number of extended paths, which is equal to $(m * M)$ in the coded system. The selection sort requires 228 cycles for choosing the best paths in an 8 state M -algorithm receiver.

A merge sorting procedure[SWP78] reduces the number of cycles required to sort a list of N elements. In this method, the list is divided into a number of smaller lists and each is sorted individually. Finally, the sorted lists are merged into a single list. The four paths from each state are initially sorted among themselves and then

Duplicate state elimination

Figure 5.4: Duplicate state comparator

merged with sorted lists from other states. The sorting of four paths in each state requires $4 * 3/2 = 6$ cycles. All the states sort their respective lists simultaneously in parallel using M sets of hardware as shown in Figure 5.5.

The selection of the best M paths from these sorted lists requires $M * (M - 1)$ cycles. As individual lists are sorted, only the top most elements in the lists need to be compared in this selection process. Thus the total number of cycles required for merge sorting is 62 (use of parallel hardware is accounted for here). The merge sorting procedure, implemented on a Batcher's network, is recommended for metric sort in [MS86] when the number of processing elements available is less than $N/2$.

In the merging of the sorted sublists *Comparator 0* is used to select the path with the minimum metric. In the first cycle, the top most elements of the lists of state 0 and state 1 are compared. The winning state number is stored in a register. In the next cycle, the winner of the previous cycle and the top most element of the list of state 3 are compared and the winning state number is registered again. This procedure is carried out till the top most element of the list of state 7 is compared. Once an element at the top of the list is selected as the best, the next element in the list becomes the top most element in the subsequent selection process.

Suppose that the top most element of state 2 had the lowest metric. Then the path history corresponding to this extended path is transferred to the master stage of the state 0 path history. The next element in the list of state 2 is now marked as the top of the list for further selection. To select the second best path, the above mentioned procedure is repeated starting the comparison of elements from state 0 and state 1. If all the elements of a state are selected before all the M paths are chosen, a maximum integer may be input from that state for the comparison. In each iteration the selected path is copied into the master stage of the path memory of the state corresponding to the iteration number. At the end, when all the M best paths are selected the path memory is updated as described in the next section.

State metric sorting

Figure 5.5: State metric sort

Figure 5.6: Path memory register Cell

5.3.3 Path memory updating

Once the best M paths are selected from the extended paths, the path memory must be updated by concatenating the branch symbols of the selected path with the path history symbols in the path memory. The memory cells storing the data symbols are assumed to be of a master-slave type registers as shown in Figure 5.6. The data symbols in these memory cells are output on the row bus during the convolution operation of the branch metric computation and on the column bus during the path memory updating.

As the length of the path memory is limited, the oldest symbols in the paths will be discarded. The oldest symbol in the path with the best metric, *i.e.*, of state 0 is declared as the decoder output. As any path history may be copied onto any other path history, some form of double buffering is necessary. This would double the mem

ory required for the path memory. Even if the memory complexity can be tolerated, the data transfer rate from one memory to the other may be a bottleneck in high speed transmission systems such as high bit rate DSL. Another method is to make all the symbol transfers simultaneously. This requires a large number of interconnection wires between the two columns of memory cells. To reduce the interconnection complexity a single bus type arrangement shown in Figure 5.7 is proposed.

The symbols of one path are copied into the master section of the receiving register cells initially during the best path selection procedure. As the survivor paths are selected sequentially, the transfer to the master section of the receiver occurs sequentially. When all the M paths are selected, a single pulse will transfer the information from the master section to the slave section of the register cells. This is necessary because the old data symbols in the path history of a lower order state may be needed while updating the path history of a higher order state. This approach would reduce the silicon area required for the path memory implementation by nearly 50%.

5.4 Silicon area estimate for the sequence estimator

The silicon area required for the implementation of the M-algorithm based sequence estimator is estimated by determining the total number of gates. The total number of gates is then multiplied by the unit gate area of a particular technology to obtain a rough estimate of the chip size. The major functional blocks of the sequence estimator is shown in Figure 5.8. The path memory is implemented by dynamic register cells as shown in Figure 5.6. A unit gate is considered to be a 2 input NAND gate with 4 transistors. The register cell with the associated control signal is assumed to be equivalent to 2 unit gates. As the path memory stores the channel symbols which are each one of 8 levels, three bits are required to represent a data symbol and hence

Figure 5.7: Path memory

3 register cells are needed. The path memory length is denoted by PML . The total number of gates required for path memory = $PML * 3 * 2$.

The branch metric and path metric computation is done using a 3 bit X B bit multiplier. A parallel array multiplier shown in Figure 5.9 is assumed here. The last row of the adder consists of a Manchester carry chain adder[WE85] with groups of 4 bits. The 1 bit full adder is made up of 6 equivalent gates and a 4 bit group Manchester carry chain adder is made up of 22 gates. The total number of gates for one 3 bit X B bit multiplier is

$$3 * CFB * 3/2 + CFB * 6 + CFB * 22/4 \quad (5.4)$$

where CFB is the wordlength of the impulse response coefficient.

The accumulator is made up of another Manchester carry chain adder and a register. The register in the accumulator is made up of D flip flop which requires 5 equivalent gates. Hence the total number of gates for the accumulator is equal to

$$BMB * 22/4 + BMB * 5 \quad (5.5)$$

where BMB is the wordlength of the branch metric.

As there are M such branch metric calculation circuits in the M - algorithm receiver, similar circuits are used. The total number of gates in one branch metric calculation unit is

$$(3 * CFB * 3/2 + CFB * 6 + CFB * 22/4) + (BMB * 22/4 + BMB * 5) \quad (5.6)$$

There are four registers for storing the branch metrics and four registers for storing the path metric in each state. The number of gates required are

$$2 * PMB * m * M * 3/2 \quad (5.7)$$

where PMB is the wordlength of the path metric.

The duplicate state elimination circuit consists of an array of 4 X 4 comparators. The comparator circuit is shown in Figure 5.10. The number of gates required in the

Figure 5.8: M-algorithm based sequence estimator

Legend : One bit Full Adder

Figure 5.9: Parallel multiplier

CO : Carry out

Figure 5.10: Digital comparator

comparator is equal to $PMB * 28/4$. Therefore the total number of gates required for the duplicate state elimination circuit is

$$4 * 4 * PMB * 28/4 \quad (5.8)$$

The state sorting circuit consists of M comparators and register cells for storing the path metric. These register cells are the same as the ones mentioned before. Hence the number of gates counted in the circuit will be that of the comparator. The total number of gates for M comparators is equal to

$$M * PMB * 28/4 \quad (5.9)$$

The impulse response coefficients are stored in register cells. Assuming that the ISI length is denoted by $ISIL$, the total number of gates required for this storage is

$$ISIL * CFAb * 3/2 \quad (5.10)$$

Here $CFAb$ is used because the impulse response coefficients are adapted recursively using LMS algorithm, which requires more bits to be kept in the register file than that are used in the branch metric computation. The adaptation is assumed to be done by a sign algorithm. This requires an adder which requires $CFAb * 22/4$ gates.

There are a number of multiplexers in the sequence estimator to route proper signal values to the multipliers and the adders. The number of gates required for the multiplexors is equal to

$$PMB * \text{number of Muxes} * M * 1 \quad (5.11)$$

where a unit gate is required for 1 bit multiplexing.

The control logic needed to generate the control signal for the functional units is assumed to require about 2000 gates. The routing of the signals requires a considerable amount of silicon area. To a first approximation this is assumed to be equal to 1/3 of the total gate area. The testing of the circuit requires additional logic which is assumed to require 20% of the total number of gates. Adding all these, we arrive at the total number of gates for the M- algorithm sequence estimator. Total number of gates for an 8 state M-algorithm sequence estimator with the following values for the parameters is equal to ≈ 27000 .

Parameter : values

coefficient wordlength(CFb) : 11

coefficient wordlength

in the adaptation($CFAb$) : 18

received signal wordlength : 13

Path metric wordlength(PMB) : 24

Branch metric wordlength(BMB) : 20

path memory length(PML) : 100

number of states(M) : 8

number of information levels(m) : 4

number of channel symbol levels(L) : 8

The wordlengths of these variables are determined from an analysis of their dynamic range and the SNR required in the detector. Let the noise introduced in the system due to quantization of the coefficients be less than 40 dB. The wordlength of the equalized end-to-end response coefficients is determined by estimating the mean-squared error in the ISI estimate due to quantization of the coefficients. For an equalized response length of N taps, the error is given by

$$N \cdot \frac{q^2}{12} \cdot P_s \quad (5.12)$$

where P_s is the average power of the data symbols. This error should be 40 dB below the decision distance, which is unity in the worst case. Therefore

$$N \cdot \frac{q^2}{12} \cdot P_s = \frac{1}{10^4} \quad (5.13)$$

For an equalized response length of 50 baud intervals (see Figure 3.2), and an average power $P_s = 21$ for the coded system considered

$$\begin{aligned} q &= \left(\frac{1}{10^4} \cdot \frac{12}{50 \cdot 21} \right)^{1/2} \\ &= \frac{1}{2^{10}} \end{aligned} \quad (5.14)$$

To represent both positive and negative values of the coefficient we need 11 bits.

The wordlength determined above is the number of bits to be used in the convolution of the branch metric. As the equalized response coefficients have to be found adaptively in a real system, we determine the wordlength of the tap coefficients for adaptation. The convergence of the adaptation is guaranteed only if the step size α

MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS
STANDARD REFERENCE MATERIAL 1010a
(ANSI and ISO TEST CHART No. 2)

is bound by

$$0 < \alpha < \frac{1}{N \cdot P_s} \quad (5.15)$$

For the values of N and P_s mentioned above $\alpha < \frac{1}{2\pi}$. To reduce the hardware required for the adaptation, the sign algorithm[VdESG79] version of the LMS-algorithm is used. The coefficient update equation is

$$C_{k+1}^i = C_k^i - \alpha \cdot \text{sign}(\epsilon_k) \cdot x_{k-i} \quad (5.16)$$

To obtain an error of 1/100 (corresponding to 40 dB SNR), the step size α in the sign-algorithm should be scaled (from 5.15) by this amount. Hence, the wordlength of the tap coefficients is equal to 18 bits.

The dynamic range of the branch metric can be estimated from the maximum value of the equalized received signal. The expression for the equalized received signal is

$$r_k = \sum_{i=0}^L f_i \cdot a_{n-i} + n_i \quad (5.17)$$

where L is the length of the response and n_i is the noise sample.

Assuming that the noise is small, the maximum value of r_k is

$$|r_k|_{\max} = |a_{n-i}|_{\max} \sum_{i=0}^L |f_i| \quad (5.18)$$

For the equalized responses obtained in Chapter 3 for the worst case loops, $|r_k|_{\max}$ is equal to 42. For a SNR of 40 dB, the ISI free signal should be represented by 8 bits (including the sign bit). This corresponds to a quantization level of $q = \frac{1}{2^8}$. Therefore, to represent the maximum range of r_k , the wordlength of the equalized received signal should be 13 bits. Theoretically, the range of values for the branch metric would be square of two times the received signal (see equation 5.1), which would require 28 bits. As the probability of the received signal attaining the maximum value is very small, we predict that the branch metric could be represented by 20 bits.

The path metric is the summation of the individual branch metrics along the

Operation	Number of cycles
convolution	50
Branch metric	32
Path metric	4
Duplicate state elimination	28
State metric sorting	62
Path update	1
Total number of cycles	177

Table 5.1: Computational time of the processing steps in the M-algorithm

survivor path. With suitable periodic scaling of the path metric, the wordlength of the path metric could be 24 bits.

The unit gate area of a 2μ CMOS process was found[Tex] to be $2016 \mu^2$. Scaling this for a 1μ CMOS process we assume that the unit gate area $\approx 720\mu^2$. Therefore the Silicon area of an 8 state sequence estimator in 1μ CMOS process ≈ 30 sq. mm.

The speed of the processing elements is another important parameter in the realization of the circuit. We estimate the cycle time of the system clock by estimating the number of clock cycles required to perform all the operations described earlier. To reduce the hardware required in the branch metric and path metric evaluation the processing element consisting of 3 bit X B bit multiplier/adder is used repeatedly, thus spreading the computation in time. The number of cycles required for each operation is shown in Table 5.1.

The symbol period of the 800 Kbits/sec systems with 400 Kbaud line rate, is equal to 2500 nsec. As the number of clock cycles needed to perform the M-algorithm in each baud interval is equal to 177, a clock period of 14 nsec (which corresponds to a clock frequency of 71 MHz) is required. The critical operation that is likely to have

the longest delay is the 3 bit X B bit multiplier. As shown in Figure 5.9, this involves the delay of an AND gate, 1 bit full adder and the Manchester carry chain adder. A 24 bit Manchester carry chain adder with a propagation delay of 12.8 nsec, in a 2μ CMOS technology, has been reported in[RHTA89]. Scaling¹ this for a 1μ process, we believe that the cycle time requirement of 14 nsec for the processing element can be met.

5.5 Conclusion

We have presented an architecture for the VLSI realization of the M-algorithm based sequence estimator. We have proposed a new method for the path memory implementation. This reduces the silicon area needed for its VLSI implementation. The operations are performed in a serial/parallel fashion to reduce the hardware required, but at the sametime provide enough parallelism to complete all the operations within a baud period. A novel approach for state sorting/duplicate state elimination involved in the M-algorithm is proposed that reduces the number of cycles needed for this operation. The silicon area required for the implementation of the proposed architecture in 1μ CMOS technology is estimated to be about 30 sq. mm. The clock frequency of the processing element was determined to be 71 MHz.

¹Scaling of the minimum dimension of a CMOS technology by a factor α leads to a reduction in the propagation delay of the gates. The delay of the gates in the scaled process will be approximately $1/\alpha$ [WE85].

Chapter 6

Oversampled Echo Cancellation

6.1 Introduction

An echo canceller is an important functional unit of the digital subscriber loop transmission system. An oversampled A/D converter is commonly used at the receiver front end to digitize the received signal in the U-interface transceiver implementations [CGH⁺89] [AKea89][SRD⁺87]. The oversampled A/D consists of a sigma-delta modulator and a decimation filter[Can85]. In the basic rate access the far end signal after echo cancellation is either high pass filtered or pre-equalized[TTea89][SRD⁺87] to remove any precursor and to limit the length of the post cursor to a reasonable number of baud intervals. Usually only symbol rate processing is done. But in the high bit rate system it may be necessary to use a fractionally spaced forward filter. Implementation of the forward filter requires a multibit multiplier which occupies large silicon area. As the VLSI realization of the transceiver is one of the major considerations, methods for reducing the complexity of the forward filter are needed.

A possible approach to complexity reduction is to combine the fractionally spaced forward filter with the decimation filter. This may reduce the silicon area needed

for the implementation of the two separate FIR filters. However adaptive echo cancellation following adaptive forward filtering introduces severe joint adaptation problems[FM79]. Thus for such an approach to be feasible echo cancellation must be done ahead of or combined with oversampled A/D conversion.

In this chapter we explore the possibility of the echo cancellation in the oversampled domain. As the echo canceller has to be adaptive, we propose an adaptation procedure for the new scheme. The performance in terms of the degree of echo cancellation and the convergence behaviour of the oversampled scheme are studied. The silicon area required for the combined decimation and forward filter is estimated and compared to that of the conventional implementation with two FIR filters.

6.2 Conventional echo cancellation

The high resolution required for ISDN transmission scheme necessitates the use of an oversampled A/D converter at the receiver front end to digitize the received signal. The received signal consists of the near end echo and the weak far end signal. An adaptive digital echo canceller is used to remove the near end echo before the far end signal is processed to detect the far end data sequence. The receiver signal processing with a fractionally spaced forward filter is shown in Figure 6.1.

The decimation filter of the oversampled A/D removes the high frequency quantization noise of the 1 bit quantizer in the sigma-delta modulator. The decimation filter is typically an FIR filter with input at the oversampled rate (1 bit stream), and an output (a multibit representation of the received signal) at the Nyquist rate. The echo estimate formed by the echo canceller is subtracted digitally and the residual far end signal plus any residual echo is input to the forward filter. The forward filter is again an FIR filter with a multibit input. The forward filter requires a multibit multiplier to implement the FIR filtering operation. The output of the forward filter

Figure 6.1: Conventional echo cancellation

is sampled at the baud rate to be input to the sequence estimator or a decision feedback equalizer. The adaptation of the echo canceller is done using the error signal at the subtractor output.

6.3 Oversampled echo cancellation

The echo can be cancelled in the sigma-delta modulator if the echo estimate is converted into a 1 bit stream at the oversampled rate. The structure of a 2nd order oversampled sigma-delta modulator is shown in Figure 6.2. The difference between the input signal and the quantizer output is integrated by integrator 1. The second integrator is used to integrate the difference between the output of the first integrator and the quantizer output. The quantizer output is sampled at the over sampling rate to obtain the 1 bit stream. The feedback signal applied to the subtractor is either $\pm V_{ref}$ depending on the output bit of the quantizer.

The output signal of the sigma-delta modulator consists of the input signal and the quantization noise at the high frequency. The output signal can be represented by

$$Y(z) = X(z) + Q(z) \cdot (1 - z^{-1})^2 \quad (6.1)$$

where $X(z)$ is the input signal which consists of the echo $\{e(z)\}$ and the far end signal $\{s(z)\}$. $X(z) = e(z) + s(z)$ and $Q(z)$ is the quantization noise. The quantization noise is shaped by a high pass transfer function as indicated by $(1 - z^{-1})^2$. The echo estimate $\hat{e}$, that is converted into a 1 bit stream at the oversampling rate by a digital sigma-delta modulator can be represented by

$$\hat{Y}_e(z) = \hat{e}(z) + Q_e(z) \cdot (1 - z^{-1})^2 \quad (6.2)$$

where $Q_e(z)$ is the quantization noise in the sigma-delta modulated echo estimate. The one bit signal can be subtracted in the same way as the feedback signal of Figure 6.2 at the input subtractor as shown in Figure 6.3.

Figure 6.2: Sigma-delta modulator

Figure 6.3: Oversampled echo cancellation

Now the total input signal can be considered to be the difference of $X(z)$ and $\hat{Y}_e(z)$. Thus the new output signal of the modified sigma-delta modulator can be written as

$$P(z) = X(z) - \hat{Y}_e(z) + Q(z) \cdot (1 - z^{-1})^2 \quad (6.3)$$

$$= [e(z) - \hat{e}(z)] + s(z) + Q(z) \cdot (1 - z^{-1})^2 + Q_e(z) \cdot (1 - z^{-1})^2 \quad (6.4)$$

Thus we have shown that the echo can be subtracted in the oversampled domain. The output $P(z)$ of the modified sigma-delta modulator can be lowpass filtered and decimated to obtain the far end signal. As the echo estimate is computed at the baud rate or twice the baud rate, the input to the digital oversampled sigma-delta in the echo estimate path is a sampled and held signal. Assuming that the received signal is bandlimited to half the nyquist frequency, the received signal should also be sampled and held at the same rate as the echo estimate.

To illustrate that the echo can be cancelled in the oversampled domain, the received signal is assumed to consist of two sinusoids at different levels. The echo is a sinusoid at one of the frequencies (100 kHz) and the far end signal is a sinusoid at the other frequency (50 kHz). Figure 6.4 shows the spectrum of the sigma-delta modulator output with the received signal as the input (echo is not subtracted). The echo is 20 dB above the far end signal. Figure 6.5 shows the spectrum of the sigma-delta modulator output with the echo cancelled in the oversampled domain. It can be observed that the echo is suppressed by about 80 dB by the modified sigma-delta modulator. The output of the sigma-delta modulator contains the high frequency noise which can be filtered by a lowpass filter.

Figure 6.4: Spectrum of the received signal at the Sigma-delta modulator

Figure 6.5: Spectrum with the echo cancelled

6.4 Adaptation of the oversampled echo canceller

The echo canceller has to be adapted to different line configurations. In the conventional scheme the output of the subtractor is used to adapt the echo canceller coefficients. But in the oversampled scheme the error signal is in the form of the 1 bit stream at the oversampling rate. We use a counter which acts like a crude lowpass filter to remove the high frequency noise in the error signal. The frequency response of the lowpass filter, implemented by the counter, is given by $\text{Sin}(\omega N\tau)/\text{Sin}(\omega\tau)$ [Can85][RG75], where $1/\tau$ is the oversampling frequency and 'N' is the oversampling ratio. The output of the counter is used to adapt the echo canceller.

Another possible source of the error signal for the adaptation is the output of the combined decimation and forward filter. But in a real system the forward filter has to be adapted as well. The adaptation of the forward filter would influence the adaptation of the echo canceller [FM79]. Hence the convergence could be a problem if the error signal is taken at the output of the combined filter. A higher order filter compared to the simple counter could be used at the sigma-delta modulator output to obtain the error signal, but this would defeat the purpose of this method as it would involve more hardware.

The adaptation of the echo canceller tap coefficients is done using the LMS algorithm, described by

$$h_n(k+1) = h_n(k) + \alpha \cdot \epsilon(k) \cdot x(n-k) \quad (6.5)$$

where 'n' is the tap coefficient index; $n = 1, 2, \dots, L$ and $\epsilon(k)$ is the error at time 'k', obtained by lowpass filtering (by a counter as shown in Figure 6.3) the sigma-delta modulator output. In the conventional scheme the mean squared residual echo level is defined by

$$\xi = \sum_{i=1}^L (e_i - \hat{e}_i)^2 \quad (6.6)$$

where e_i is the echo path impulse response coefficient,
 $\hat{e}_i$ is the echo canceller coefficient,
 and L is the length of the echo path response.

The adaptation behaviour of the oversampled scheme, with the mean squared error calculated using equation 6.6, is shown in Figure 6.6. The echo path is assumed to consist of 500 coefficients at $T/2$ interval. The echo level at the input of the sigma-delta modulator is taken to be 12 dB down from the full scale input of the modulator. The LMS adaptation constant is decreased from 10^{-3} to 10^{-5} as follows 10^{-3} , $5 \cdot 10^{-4}$, 10^{-4} , $5 \cdot 10^{-5}$ and 10^{-5} .

In the oversampled scheme, as the echo estimate is converted into a 1 bit stream by a digital sigma-delta modulator which has a quantizer (a nonlinear element) there was some doubt about the validity of using equation 6.6 in calculating the residual error. Hence residual error at the output of the decimation filter (which would be combined with the forward filter in the actual implementation) was obtained with the echo cancelled in the oversampled domain.

Figure 6.7 shows the residual echo at the output of the decimation filter. The convergence characteristics is observed to be very similar to that shown in Figure 6.6. In these simulations the far end signal was assumed to be zero. In the presence of far end signal the adaptation will be slower. It can be observed that the echo is suppressed by only 44 dB in the over sampled scheme. Beyond that the slope of the error curve is very small. This is due to the inadequate filtering of the quantization noise in the error signal used for the adaptation.

6.5 Combined decimation and forward filtering

As the echo is subtracted in the sigma-delta modulator, the decimation filter and the forward filter can be combined together into a single filter. The input data to

Figure 6.6: Residual echo by conventional definition

Figure 6.7: Residual echo after decimation filter

this filter is a single bit stream at the oversampling rate. The output of the filter needs to be computed only at the baud rate and is fed to the DFE or a sequence estimator. As the input data is only 1 bit the convolution involved in the FIR filtering reduces to only an accumulation of the filter coefficients. This obviates the need for a multibit multiplier in the conventional implementation. This method for reducing the complexity of the adaptive transversal filter has been reported in [WC88][TS85]. The impulse response of the combined filter is obtained by convolving the impulse responses of the decimation filter and the forward filter. The optimum fractionally spaced forward filter response for the worst case CSA loop (9 Kft, 26 Gauge) is convolved with the impulse response of the decimation filter to obtain the equivalent filter response shown in Figure 6.8. As the combined filter has to operate at the oversampled rate, there are a large number of coefficients in the filter.

6.6 Silicon area estimate

A rough estimate of the silicon area required for the implementation of the decimation filter and the forward filter in the conventional scheme and the oversampled scheme is obtained. The architecture of the forward filter in the conventional scheme is shown in Figure 6.9. As the sigma-delta modulator is common to both schemes, this is not included in the comparison.

The decimation filter is assumed to be implemented by a ROM based FIR filter. The silicon area required for this is ≈ 6 sq. mm. [Ada86]. The wordlength of the forward filter coefficients and the received signal that is input to this filter are required to estimate the silicon area of the register files used to store these values. The wordlength of the received signal has been determined to be 13 bits in Chapter 5 (see Section 5.4).

The wordlength of the forward filter coefficient is determined by the noise intro-

Figure 6.8: Combined decimation and forward filter response

Figure 6.9: Architecture of the forward filter in the conventional scheme

duced in the filter output due to quantization of the coefficients. The mean squared noise power is given by

$$N_f \cdot \frac{q^2}{12} \cdot P_r \quad (6.7)$$

where N_f is the number of forward taps, P_r is the mean squared value of the received signal and q is the quantization step size of the coefficients. The number of coefficients in the forward filter is equal to 21 as determined in Chapter 3. Assuming that an adaptive gain control circuit is used to compensate for the line loss, the mean squared value of the received signal in the coded system is given by

$$P_s \cdot \sum_{i=1}^L |h(i)|^2 \quad (6.8)$$

where P_s is the average power of the signal set (21 in the coded case) and $\{h(i)\}$ are the impulse response coefficients with the cursor equal to unity. The sum of squares of impulse response coefficients for the worst case CSA loop (9 Kft, 26 Gauge) was calculated to be 2.4.

Substituting the appropriate values in the above equation, the value of ' q ' to obtain a 40 dB SNR is given by

$$q \leq \frac{1}{2^{10}} \quad (6.9)$$

The maximum absolute value of the optimum forward filter coefficient for the loop mentioned above was found to be ≈ 35 . Thus the wordlength of the filter coefficient to represent the above range of values (both positive and negative) is 17 bits. As the forward filter coefficients have to be determined adaptively, the wordlength of the coefficients for adaptation was estimated to be 24 bits using the method described in Chapter 5. Therefore a 13 $\times$ 17 bit multiplier is required in the forward filter. The adder is a 32 bit adder. In the first half of the clock cycle the data from the input register file is multiplied with the coefficient and the product is accumulated in the accumulator. The second half of the cycle is used for adaptation of the coefficient. During this half of the clock cycle the error signal is multiplexed onto the right leg of the multiplier and the product is added to the coefficient to obtain the updated

Functional Unit	Area in sq.mm
Multiplier	8
Register, adder, Mux and routing	2.4
Decimation filter	6

Table 6.1: Silicon area estimate for the conventional scheme

coefficient for the next baud interval. This is stored back in the coefficient register file. The approximate silicon area for each of the functional blocks is obtained by multiplying the total number of gates for the function by the unit gate area of a 2μ CMOS process. The silicon area estimates are shown in Table 6.1. The area of the multiplier is obtained from [KBHH89].

The total area for the implementation of the decimation filter and the forward filter by the conventional method is 16.4 sq.mm.

The impulse response of the combined filter is approximately 1855 taps long. The over sampling ratio is 128. The architecture for the VLSI implementation of the combined filter is shown in Figure 6.10. The convolution is split into 15 sections that are working in parallel, due to the speed limitations of the adders. In each section an accumulation operation is done within the oversampling clock period. The input data to the shift register is a 1 bit stream at the oversampled rate. The output of the filter is needed only at the baud rate.

The coefficients are adapted simultaneously along with the convolution. The outputs from the section registers are added together by another adder to obtain the output of the combined filter. The wordlength of the coefficients would be the same as for the conventional method. The register files are used to store the coefficients. The total area for this implementation was calculated in a similar way as for the

Figure 6.10: Architecture of the combined decimation and forward filter

conventional implementation. The total area was estimated to be 60 sq.mm in 2μ CMOS technology. The large area is mainly due to the coefficient storage. The area estimates are obtained by using data provided in the standard cell data book of the Texas Instruments. There may be other memory implementation techniques that would result in a reduced area for the coefficient storage. However, from the echo cancellation point of view as well as the silicon area requirement the oversampling approach does not seem to be favourable.

6.7 Conclusion

A new approach was proposed for the echo cancellation in the oversampled domain. This would enable one to combine the decimation filter and the forward filter of a DFE or a sequence estimator. The adaptation behaviour of the new scheme was studied by simulation methods. An echo suppression of about 44 dB was shown to be feasible. This may not be sufficient for the ISDN application. The low level of echo suppression is attributed to the high frequency noise present in the error signal used for the adaptation of the echo canceller. The silicon area required for the implementation of the echo canceller and forward filter using conventional and oversampling schemes were estimated. The oversampling scheme was found to require significantly larger area due to the memory required for the coefficient storage. The use of dynamic RAM may reduce the area significantly. This may be pursued in the future.

Chapter 7

Reduced Complexity Transversal Filter Implementation

7.1 Introduction

In this chapter we will describe a new approach to reduce the complexity of the transversal filter that is commonly used in echo cancellation, or the feedback filter of a DFE, or in the branch metric calculation of the sequence estimator. The complexity is due to the long impulse response of the echo path or the subscriber loop channel. Traditionally, this problem is solved by using a $(1 - D)$ or other high pass pre-filter to reduce the length of the impulse response. This method may not be suitable for digital subscriber loops since the crosstalk which has a power spectrum of $f^{3/2}$ will be further accentuated.

There have been several methods proposed to reduce the complexity of the transversal filter with an impulse response having a slowly decaying tail. Use of an infinite impulse response (IIR) filter with fixed poles to approximate the tail was suggested in[MNP86]. The first few baud intervals of the response are taken care of by a FIR fil-

ter This approach is commonly used in the basic rate access system[TTea89][AKea89]. However performance may be sensitive to the choice of pole locations.

An interpolated FIR filter has been shown to be useful for reducing the complexity of the transversal filter in[Abo90]. In this method an FIR filter with tap spacing of LT (L is an integer > 1) is followed by an interpolating fixed IIR filter. Significant reduction in the number of multiplications and additions was reported. The adaptation behaviour of the IFIR filter was also studied in[Abo90].

Davidson and Cook[Dav89][Coo89] considered the approximation of the tail of the response by a linear combination of orthonormal functions. These functions were chosen such that they can be implemented by simple recursive filters with adaptive gains. Reduction in the complexity was found to be substantial for high bit rate system.

While these approaches use IIR filters in one form or the other, we propose a FIR filter structure for the approximation of the tail[MJFK89]. The solution proposed here involves dividing the long impulse response into suitable number of segments and approximating them by piecewise linear segments in the mean squared sense. A similar approach with piecewise linear approximation of the long tail, by taking the end points of the segments equal to the actual response at that baud interval, developed independently, was reported in[EA90]. We illustrate the application of this technique for the ISI estimation in a DFE. Each segment is represented by a single parameter in the computation of the ISI. Computational complexity, defined in terms of the number of multiplications and additions, of the piecewise linear filter approach and the conventional transversal filter method is compared. The performance improvement obtained by this technique is illustrated with the impulse response of a typical digital subscriber loop.

As the impulse response of different loops may differ in shape, the segment boundaries may need to be adapted. Adaptive determination of segment lengths involves

additional complexity. Computations performed with a fixed partitioning for a set of loops indicates that there is no significant performance loss.

As the channel impulse response changes with time, some form of adaptation is required. A modified form of the LMS algorithm is proposed for the adaptation of the piecewise linear filter coefficients. To reduce the complexity of the adaptation a sign algorithm based LMS adaptation was proposed. The performance was observed to be essentially the same as for the stochastic iterative adaptation. The wordlength of the variables involved in the sign-algorithm implementation of the adaptive piecewise linear filter were determined. A rough estimate of the silicon area required for the VLSI implementation of this method is calculated and compared to that of the transversal filter.

7.2 Piecewise linear approximation

The impulse response of a typical digital subscriber loop is shown in Figure 7.1. The line response spans about 1024 baud intervals (*i.e.*, 51 dB down at baud period 1024). The long tail of the impulse response can be approximated by piecewise linear segments as shown in Figure 7.2.

Figure 7.3 shows the details of the piecewise linear approximation for a particular segment. Mathematically the approximated response $\hat{g}(m)$ can be written as

$$\hat{g}(N_i + j) = \frac{(K_i - j)}{K_i} h(N_i) + \frac{j}{K_i} h(N_{i+1}) \quad (7.1)$$

for $1 \leq j \leq K_i$ and $i = 0, 1, \dots, (I-1)$. In this expression, the N_i , $\{i = 0, 1, \dots, I\}$ are the end points of (I) segments in which $\hat{g}(\bullet)$ varies linearly with ' j ' and $\{K_i = N_{i+1} - N_i\}$ are the segment lengths ($K_i \geq 1, i = 0, 1, \dots, (I-1)$). Thus a span of $(N_I + 1)$ symbol intervals is represented by $(I + 1)$ samples $\{h(N_i), i = 0, 1, \dots, I\}$.

Figure 7.1: Sample impulse response of the loop

Figure 7.2: Piecewise linear approximation of the impulse response

Figure 7.3: Piecewise linear approximation of a segment.

It is assumed that $h(N_0) = 0$. If $K_i = 1$, then

$$h(N_{i+1}) = \hat{g}(N_{i+1}) = g(N_{i+1}) \quad (7.2)$$

and there is no approximation. The initial portion of the impulse response tends to vary rapidly. Hence a segment length of unity (*i.e.*, no approximation) may be used for the first few baud intervals. The tail of the impulse response can be approximated by larger segment lengths as the tail decays very slowly.

For a given set of ' I ' end points defined by $\{N_i = N_1, N_2, \dots, N_I\}$ it is straight forward to express and minimize the sum of squares of error

$$e(N_i + j) = \hat{g}(N_i + j) - g(N_i + j) \quad (7.3)$$

for $0 \leq i \leq (I-1)$ and $1 \leq j \leq K_i$ with respect to the end points $\{h(N_i)\}$. The result is a set of ' I ' linear equations for the $\{h(N_i)\}$, giving a least squares piecewise linear fit to the given response from $m = 1$ to $m = N_I$. Appendix A describes the mathematics involved in the formulation and solution of the system of linear equations.

Figure 7.4 shows the error signal $\{e_i\}$ for two sets of segment lengths, designated case I and case II. The tail of the impulse response is approximated by segment lengths (8,8,8,8,64,128) in case I and (8,8,8,8,32,32,32,32,32,32) in case II. Note that in both cases the error is zero for the first 32 samples for which $K_i = 1$.

It can be seen that the error signal is large in case I because the segment length is large compared to case II. The 'approximation error' can be defined as

$$\text{Approximation error} = \frac{\text{minimum sum of squares of error}}{\text{energy in the original response}} \quad (7.4)$$

This was calculated to be -37.8 dB in case I and -48.0 dB in case II. The corresponding 'SNR', defined by

$$\text{SNR} = \frac{\text{square of the maximum of the impulse response}}{\text{sum of squared errors}} \quad (7.5)$$

is 30.1 dB in case I and 40.3 dB in case II.

Figure 7.4: Approximation error

The SNR obtained above may be adequate for the DFE, but better performance may be needed for echo cancellation. This may be obtained by choosing the segment lengths appropriately.

7.3 Implementation and complexity

The convolution of the data symbols and the impulse response in a piecewise linear fit filter is implemented as follows. The ISI estimate is equal to the sum of the ISI s from each segment. Hence the PLF filter output can be written as

$$y_n = \sum_{i=1}^I Q_n(i) \quad (7.6)$$

where $Q_n(i)$ is the estimated ISI from the segment 'i'. The ISI from an impulse response of length N_I at time 'n' can be written as

$$y_n = \sum_{k=1}^{N_I} h(k) a_{n-k} \quad (7.7)$$

where the $\{h(k)\}$ are the impulse response coefficients and $\{a_k\}$ are the data symbols. In a piecewise linear approximation the coefficient at the index $k = (N_i + j)$ is given by

$$h(N_i + j) = \frac{(K_i - j)}{K_i} h(N_i) + \frac{j}{K_i} h(N_{i+1}) \quad (7.8)$$

Substituting 7.8 in 7.7 and splitting the summation over the segments we obtain

$$y_n = \sum_{i=0}^{I-1} \sum_{j=1}^{K_i} h(N_i + j) a_{n-(N_i+j)} \quad (7.9)$$

$$= \sum_{i=0}^{I-1} \sum_{j=1}^{K_i} \left\{ \frac{(K_i - j)}{K_i} h(N_i) + \frac{j}{K_i} h(N_{i+1}) \right\} a_{n-(N_i+j)} \quad (7.10)$$

$$= \sum_{i=0}^{I-1} \sum_{j=1}^{K_i} \left\{ \frac{(K_i - j)}{K_i} a_{n-(N_i+j)} \right\} h(N_i) + \sum_{i=0}^{I-1} \sum_{j=1}^{K_i} \left\{ \frac{j}{K_i} a_{n-(N_i+j)} \right\} h(N_{i+1}) \quad (7.11)$$

$$= \sum_{i=0}^{I-1} Y_n(i) h(N_i) + X_n(i) h(N_{i+1}) \quad (7.12)$$

$$= \sum_{i=1}^I h(N_i)[X_n(i-1) + Y_n(i)] \quad (7.13)$$

$$= \sum_{i=1}^I h(N_i)S_n(i) \quad (7.14)$$

where

$$\begin{aligned} X_n(i) &= \sum_{j=1}^{K_i} \frac{j}{K_i} a_{n-j-N_i}, & 0 \leq i \leq I-1; \\ &= 0, & i < 0; \end{aligned} \quad (7.15)$$

$$\begin{aligned} Y_n(i) &= \sum_{j=1}^{K_i} \frac{(K_i-j)}{K_i} a_{n-j-N_i}, & 0 \leq i \leq I-1 \\ &= 0, & i = I; \end{aligned} \quad (7.16)$$

and

$$S_n(i) = X_n(i-1) + Y_n(i) \quad (7.17)$$

The quantities $X_n(i)$ and $Y_n(i)$ are FIR filtered data which can be computed recursively as follows. Let $Z_n(i)$ be defined as

$$Z_n(i) = a_{n-1-N_i} + a_{n-2-N_i} + \dots + a_{n-K_i-N_i} \quad (7.18)$$

then

$$\begin{aligned} Z_{n-1}(i) &= a_{n-2-N_i} + a_{n-3-N_i} + \dots + a_{n-K_i-N_i-1} \\ &= a_{n-2-N_i} + a_{n-3-N_i} + \dots + a_{n-N_{i+1}-1} \end{aligned} \quad (7.19)$$

subtracting (7.19) from (7.18) and rearranging we get

$$Z_n(i) = Z_{n-1}(i) + a_{n-1-N_i} - a_{n-N_{i+1}-1} \quad (7.20)$$

From equation (7.15)

$$\begin{aligned} X_n(i) &= \frac{1}{K_i} [1a_{n-1-N_i} + 2a_{n-2-N_i} + \dots + \\ &\quad (K_i-1)a_{n-N_{i+1}+1} + K_i a_{n-N_{i+1}}] \end{aligned} \quad (7.21)$$

$$\begin{aligned} X_{n-1}(i) &= \frac{1}{K_i} [1a_{n-2-N_i} + 2a_{n-3-N_i} + \dots + \\ &\quad (K_i-1)a_{n-N_{i+1}} + K_i a_{n-N_{i+1}-1}] \end{aligned} \quad (7.22)$$

subtracting (7.22) from (7.21) and collecting terms corresponding to $Z_{n-1}(i)$

$$X_n(i) = X_{n-1}(i) + \frac{1}{K_i}Z_{n-1}(i) + \frac{1}{K_i}a_{n-1-N_i} - (1 + \frac{1}{K_i})a_{n-1-N_{i+1}} \quad (7.23)$$

Similarly

$$Y_n(i) = \frac{K_i - 1}{K_i}a_{n-1-N_i} + \frac{K_i - 2}{K_i}a_{n-2-N_i} + \dots + \frac{1}{K_i}a_{n-K_i-N_{i+1}} \quad (7.24)$$

$$Y_{n-1}(i) = \frac{K_i - 1}{K_i}a_{n-2-N_i} + \frac{K_i - 2}{K_i}a_{n-3-N_i} \dots + \frac{1}{K_i}a_{n-K_i-N_i} \quad (7.25)$$

subtracting 7.25 from 7.24 and collecting terms corresponding to $Z_n(i)$

$$Y_n(i) = Y_{n-1}(i) - \frac{1}{K_i}Z_n(i) + a_{n-1-N_i} \quad (7.26)$$

The variables $Z_n(i)$, $X_n(i)$ and $Y_n(i)$ are initialized to zero for $n \leq 0$.

The complexity of the piecewise linear fit method and the conventional method can be compared in terms of the number of multiplications and additions. For segments of length unity, the complexity in the PLF method is the same as for the conventional scheme. If segment length $K_i > 1$, then the computations involved in the estimation of ISI from that segment are embodied in equations (7.13), (7.20), (7.23) and (7.26). There are eight additions and one multiplication required per segment. The segment lengths K_i are chosen to be integer powers of two so that the division by K_i in (7.23) and (7.26) can be implemented by bit shifts. As the dynamic range of the variables X_n and Y_n are from $1/K_i$ to $K_i * |a_n|_{max}$, the wordlength of these variables are $(1 + \log_2 K_i)$ times that of the 2-bit data symbols (as in 2B1Q coding). Thus the multibit multiplication required in (7.13) is equivalent to $(1 + \log_2 K_i)$ multiplications in the conventional implementation.

Defining a multibit addition or a 2 bit X B bit multiplication (which can be done by a shift and add/subtract in the 2B1Q coding) as a unit operation, the total number

Segment length K_i	number of operations/segment	
	PLF	Conventional
8	12	15
16	13	31
32	14	63
64	15	127
128	16	255

Table 7.1: Computational complexity

of operations needed to compute the ISI from segments of various lengths are shown in Table 7.1. The number of unit operations required in the conventional method is equal to $2 \cdot K_i - 1$, where K_i is the segment length. It can be observed that the complexity increases logarithmically in the PLF method while it increases linearly in the conventional scheme.

7.4 Performance comparison

For a given number of operations, the conventional method would span fewer number of taps compared to the PLF method and hence may yield a lower SNR because of the uncanceled ISI beyond its span. In the PLF method the complexity can be varied by starting the approximation procedure at different tap indices from the main tap or by choosing different sets of values for the segment lengths. Using the definition of unit operation mentioned previously, the SNR achievable by these two methods for different complexity of operations was calculated. Figure 7.5 illustrates the PLF's more favourable SNR for a given complexity. The first 256 samples of the impulse response shown in Figure 2.2 were considered in these calculations. Each isolated point for the PLF corresponds to a different choice of the $\{K_i\}$ set. The

Set number	partition scheme
1	ones(1,16) 8 8 8 8 16 32 32 64 64
2	ones(1,24) 8 8 8 16 32 32 64 64
3	ones(1,32) 8 8 16 32 32 64 64
4	ones(1,32) 8 8 8 8 64 128
5	ones(1,32) 8 8 8 8 32 32 64 64
6	ones(1,32) 8 8 8 8 32 32 32 32 32 32

Table 7.2: Partitioning schemes with different complexity

set of segment lengths used to obtain these results are shown in table 7.2. In the table, 'ones(1,N)' means that the first 'N' segments have unit length (*i.e.*, there is no approximation). The total number of operations for the set 3 is 160. The conventional implementation with this many operations would only span 80 taps. The ISI from the remaining taps is considered to be the error. The SNR is obtained using equation 7.5. SNR improvement of the order of 20 dB seems possible for the same complexity.

7.5 Adaptation behaviour

The transversal filter is adapted using the LMS algorithm. This is a gradient search algorithm where the tap coefficients of the filter are adjusted by estimating the slope of the error surface. A schematic diagram of the adaptive filter is shown in Figure 7.6. The filter coefficients $\{h_n\}$ are adapted at time ' k ' using the error signal $\epsilon(k)$ which is the difference between the desired signal $d(k)$ and the filter output $y(k)$. When the filter is adapted to the correct coefficients the error $\epsilon(k)$ would be zero, if

Figure 7.5: Performance (SNR) vs computational complexity

there is no noise. The adaptation of the tap coefficients is given by the expression

$$h_n(k+1) = h_n(k) + \alpha \cdot \epsilon(k) \cdot x(n-k) \quad (7.27)$$

where 'n' is the tap coefficient index; $n = 1, 2, \dots, L$.

In a conventional transversal filter adaptation each coefficient or a group of coefficients is updated in each baud interval 'k'. But in the piecewise linear filter, as a segment of impulse response is approximated by one coefficient only that coefficient is adapted. The adaptation of the i^{th} segment is given by

$$h_{N_i}(k+1) = h_{N_i}(k) + \beta_i \cdot \epsilon(k) \cdot S_i(k) \quad (7.28)$$

where $\epsilon(k)$ is again the error signal,

$S_i(k)$ is the averaged signal value of the segment 'i' as defined in (7.15), (7.16) and (7.17), and β_i is the adaptation constant for segment 'i'.

The theoretical convergence behaviour of the piecewise linear filter is discussed in Appendix B. The step size β_i is made inversely proportional to the segment length. This was found to give faster convergence compared to the constant step size for all the segments. Simulations were done to obtain the convergence behaviour of the PLF method for the loop described in Section 7.2. The mean squared error curve for the case $\alpha_i = \alpha/K_i$; $\alpha = 0.0025$, the $\{K_i\}$ are those of case II, and the PLF filter approximating the first 256 samples of the channel response are shown in Figure 7.7. Also shown in the figure is the theoretical convergence behaviour. The steady state mean squared error for both cases can be observed to be the same.

7.6 Adaptive segment length determination

The results obtained in Section 7.2 indicate that the performance of the piecewise linear fit method depends on the way the segment lengths are chosen. For example, the SNR's for the two partitionings illustrated as cases I and II in Figure 7.4 differ

Coefficient adaptation

Figure 7.6: Adaptive filter

Figure 7.7: Mean squared error performance

by a large value. Hence it may be necessary to adaptively determine the partitioning for each individual loop. A set of six loops were chosen to study the sensitivity of the partitioning. The loop characteristics are as shown in Figure 7.8. The tail of the impulse responses for these loops are plotted in Figure 7.9. To meet a certain SNR criterion, with a constraint on the number of segments an adaptive partitioning algorithm was developed.

The algorithm allocates larger lengths for the segments till the error due to that segment is more than the budgeted error available for that segment. As the segment lengths are constrained to be integer powers of two, any leftover segment error is added to other segments. The results of the adaptive segmentation algorithm applied to the test loops are shown in Table 7.3. It must be noticed that the segment lengths are not monotonically increasing. From an implementation point of view it may be desirable to keep the partitioning fixed for all the loops. To determine the performance degradation due to a fixed partitioning, the partitioning for the loop 4 was applied to the test loops. The results of individually optimized partitioning and fixed partitioning are shown in Table 7.4. It can be observed that the degradation is very small and that the fixed partitioning still gives an SNR greater than 45 dB for all the loops. Another approach to determine the partitioning may be to find a mean impulse response for a set of loops and use the optimized partitioning for the mean impulse response for all the loops. The results of this method for the set of loops is also shown in Table 7.4. From this it can be concluded that adaptive segment length determination is not necessary to obtain good performance as it involves additional complexity.

Figure 7.8: Test loops

Figure 7.9: Impulse responses of the test loops

Loop number	optimal partition
1	ones[1,32] 8 8 16 32 32 32 32 64 64 64 128
2	ones[1,32] 16 16 32 32 32 32 64 64 64 128
3	ones[1,32] 8 8 16 32 32 32 32 32 32 64 64 128
4	ones[1,32] 8 8 16 8 8 16 32 64 64 64 128
5	ones[1,32] 16 16 8 8 16 16 16 64 64 64 64 128
6	ones[1,32] 8 8 16 16 16 32 32 32 32 32 64 64 128

Table 7.3: Optimal partitioning of the test loops

Loop number	SNR in dB		
	using optimal partition	using fixed partition	using optimal partition of mean response
1	49.48	48.02	48.12
2	48.86	48.71	48.89
3	47.96	47.56	47.89
4	49.29	49.29	49.72
5	48.41	46.32	46.74
6	49.05	47.80	48.27

Table 7.4: Performance under various partitioning schemes

7.7 Adaptation of PLF filter using sign algorithm

The LMS adaptation algorithm described in Section 7.5 uses the actual value of the error signal in the adaptation process (*i.e.*, in evaluating the coefficient update term). This requires a real multiplier which is very expensive in terms of silicon area. To minimize the hardware required for adaptation the sign of the error can be used to calculate the coefficient update term. The structure of the adaptation unit for the sign algorithm is similar to that shown in Figure 7.6 with $e(k)$ replaced by $\text{sgn}[e(k)]$. The LMS adaptation based on sign algorithm is described by the equation

$$C_{n+1}^k = C_n^k - \alpha \cdot \text{sgn}(e_n) \cdot x(n - k) \quad (7.29)$$

The adaptation behaviour of the PLF filter using the sign algorithm is shown in Figure 7.10. The step size of the adaptation process is decreased in 3 steps (as shown in Figure 7.10) so that the speed of convergence is not degraded by the smaller step size required to obtain the minimum mean squared error that was achieved in conventional LMS algorithm of Section 7.5.

Figure 7.10: Convergence behaviour of the sign algorithm

7.8 Hardware complexity (wordlength determination)

The signal processing involved in the piecewise linear fit method is described by the following equations:

$$Z_n(i) = \sum_{j=1}^{K_i} a_{n-j-N_i} \quad (7.30)$$

$$X_n(i) = \frac{1}{K_i} \sum_{j=1}^{K_i} j a_{n-j-N_i} \quad (7.31)$$

$$Y_n(i) = \frac{1}{K_i} \sum_{j=1}^{K_i} (K_i - j) a_{n-j-N_i} \quad (7.32)$$

$$S_n(i) = X_n(i-1) + Y_n(i) \quad (7.33)$$

where $K_i, i = 0, 1, \dots, (I-1)$ is the length of the segment 'i' and the suffix 'n' is the time index. The ISI estimate is given by

$$y_n = \sum_{i=1}^I h_n(N_i) S_n(i) \quad (7.34)$$

The hardware implementation of the PLF filter requires determination of the wordlength of the variables mentioned above. The wordlength of a variable depends on the dynamic range of the variable concerned. The dynamic range of the variables Z_n, X_n, Y_n and S_n depend on the maximum value of segment length K_i . Assuming a maximum value of $K_i = 128$ and data symbols encoded with 2B1Q code (i.e., $\pm 1, \pm 3$ symbols), the dynamic range and hence the wordlength of these variables are calculated as shown below.

Dynamic range of Z_n :

$$|Z_n|_{max} = \sum_{j=1}^{128} |a_j|_{max} = 384 \quad (7.35)$$

$$|Z_n|_{min} = 0 \quad (7.36)$$

Therefore the number of bits required to represent $Z_n = 10$ bits.

Dynamic range of X_n :

$$|X_n|_{max} = |a_j|_{max} \frac{1}{128} \sum_{j=1}^{128} j \quad (7.37)$$

$$= 192 \quad (7.38)$$

$$|X_n|_{min} = \frac{1}{128} = 2^{-7} \quad (7.39)$$

$$(7.40)$$

The number of bits required for $X_n = 16$ bits. Similarly the number of bits required for Y_n is 16 bits.

Dynamic range of S_n :

$$|S_n(i)| = |X_n(i-1)| + |Y_n(i)| \quad (7.41)$$

Therefore the number of bits required for $S_n(i)$ is 17 bits.

The ISI estimate needs to be represented to the same accuracy as the digital representation (A/D converted) of the far end signal. Assuming a 14 bit representation for the received signal the SNR for a 1V transmitted signal can be calculated as follows. Let the attenuation of the channel be 40 dB. Hence the received signal amplitude will be 10mV. Let the supply voltage of the front end A/D be 5V. With a 14 bit representation of the full scale voltage of 5V, the SNR achievable for the 10mV received signal is given by

$$\begin{aligned} SNR &= 20 \log_{10} \left(\frac{10 \cdot 10^{-3}}{\frac{5}{2^{14}} \cdot \left(\frac{1}{12}\right)^{0.5}} \right) \\ &= 41 \text{ dB} \end{aligned} \quad (7.42)$$

The feedback filter coefficients are adapted by the modified LMS algorithm given below

$$h_{n+1}(i) = h_n(i) + \alpha \cdot \text{sgn}(e_n) \cdot S_n(i) / K_i \quad (7.43)$$

where 'i' is the segment index and 'n' is the time index.

The wordlengths of the variables Z_n, X_n, Y_n, S_n, h_n and y_n are shown in Figure 7.11 for an adaptation constant $\alpha = 10^{-8}$ of the sign algorithm and a maximum segment length $K_i = 128$. This bar chart representation for the wordlengths of the variables was proposed in[VdESG79], and we find it very useful. Even though the wordlengths were obtained for the case where there is no adaptive gain control(AGC) to compensate for the line loss, necessary changes to the wordlengths can be made from a study of the dynamic range of the variables if an AGC is used.

Quantization of the filter coefficients introduces noise in the ISI estimate. Let the quantization step size of the impulse response coefficients be equal to $1/2^{18}$. Assuming that the error introduced by the quantization of the coefficients are independent, the total error introduced in the ISI estimate due to the quantization of coefficients is given by

$$N \frac{q^2}{12} P_s \quad (7.44)$$

where N is the number of coefficients,

q is the quantization step size and

P_s is the average power of the data signal.

For the 2B1Q signal $P_s = 5$. Assuming $N = 512$, the signal-to-noise ratio of the ISI estimate due to coefficient quantization, relative to a received signal of 10mV, is given by

$$\begin{aligned} SNR &= 10 \log_{10} \left(\frac{(10 \cdot 10^{-3})^2}{512 \cdot (\frac{1}{2^{18}})^2 \cdot \frac{1}{12} \cdot 5} \right) \\ &= 45 \text{ dB} \end{aligned} \quad (7.45)$$

7.9 Silicon area estimation for the PLF and conventional methods

A functional block diagram of the PLF filter is shown in Figure 7.12.

Figure 7.11: Wordlength of the PLF filter variables

Figure 7.12: Architecture of the PLF filter

An estimate of the silicon area required for the implementation of this PLF filter in 2μ CMOS process is calculated under the following assumptions. The register files are assumed to be implemented by dynamic shift register cells consisting of six transistors[WE85]. The area of a 1 bit cell ($28\mu \times 72\mu$) is assumed to be equal to that of unit gate in Texas Instrument's 2μ CMOS Standard cell catalog[Tex]. The adders are assumed to be implemented by manchester carry chain adders. The area of the 1 bit full adder ($108\mu \times 86\mu$) was calculated by scaling the area of a 3μ CMOS implementation done by the author at Carleton University. The area for the 20 bit multiplier (11.25 sq. mm.) is obtained by scaling the area for a 16 bit multiplier implementation reported in[KBHH89]. The total area of the piecewise linear fit filter was estimated to be 20 sq. mm. as shown in Table 7.5.

The wordlength of the variables in a conventional transversal filter implementation of the feedback filter is shown in Figure 7.13. The transversal filter implementation requires a high speed adder with delay time $\approx 5\text{nsec}$ since 512 additions have to be completed in 2.5μ sec (baud period of the 400 Kbaud high bit rate system). High speed adders reported in[RHTA89] have a delay time of 10 nsec and a silicon area of ($295\mu \times 86\mu$) per bit. It is assumed that two such adders are used in parallel in the conventional implementation.

Under the assumptions mentioned above, the silicon area of the conventional transversal filter implementation is estimated to be 36.2 sq. mm. as shown in Table 7.6. It is observed that the PLF filter requires significantly less area compared to the transversal filter implementation.

7.10 Conclusion

A new approach called piecewise linear fit method was proposed to reduce the complexity of the transversal filter with a slowly decaying impulse response. The

Functional Unit	Area in sq. mm.
Z_n adder	0.0929
X_n adder	0.3715
Y_n adder	0.2786
S_n adder	0.1486
Adaptation	
adder	0.3994
y_n adder	0.2601
Z_n register	0.2419
X_n, Y_n registers	0.7741
y_n accumulator	0.0565
h_n register	3.8143
Shift register	2.0644
Delay register	0.0323
20 bit Multiplier	11.25
Total area	20 sq. mm.

Table 7.5: PLF filter area estimate

Figure 7.13: Wordlength of the conventional filter variables

Functional Unit	Area in sq. mm.
h_n register	30.9658
y_n accumulator	0.1129
y_n adder	1.4207
Adaptation	
adder	1.6237
Shift register	2.0644
Total area	36.2 sq. mm.

Table 7.6: Conventional transversal filter area estimate

performance, defined in terms of the mean squared error, of the PLF method was compared to the conventional transversal filter method of the same hardware complexity. Improvements of the order of 20 to 30 dB in SNR was shown to be possible with the new technique. A modified LMS algorithm with a step size that is inversely proportional to the segment length was found to have a faster convergence. Adaptive segment length determination for individual loops was found to be unnecessary as the performance degradation from using a fixed partitioning for a set of loops was less than 2 dB compared to the optimal partitioning.

To reduce the complexity of the adaptation a sign algorithm has been suggested for the LMS adaptation. Adaptation constant (α) for the sign algorithm is reduced gradually in three steps to obtain faster convergence. A VLSI architecture was proposed for the sign algorithm based implementation of the piecewise linear fit filter. The wordlengths of the filter variables were determined from an analysis of the dynamic ranges of the variables. Performance of the PLF filter with finite wordlength was shown to meet the filter requirements. Preliminary estimates of the silicon area required for the implementation of the PLF filter and the conventional transversal filter in 2μ CMOS technology were calculated to be 20 sq. mm. and 36.2 sq. mm.

respectively.

Chapter 8

Conclusion

The major contribution of this thesis is the determination of the transmission range of the high bit rate digital subscriber loop system, using trellis coding at the transmitter and combined trellis decoding and equalization at the receiver. The VLSI design issues of the transmitter and receiver functions are also studied in detail. The ultimate performance of a coded system on an ISI channel has been evaluated by determining the minimum distance bounds as well as by simulation. It was observed that a coded system with a relatively large number of coder states is needed to obtain reasonable coding gains on an ISI channel. Even though encoders with large numbers of states are required for obtaining the coding gain, receivers with fewer states implemented by M-algorithm were found to provide good performance. The advantage of using the M-algorithm receiver as compared to the DFSE receiver is that the number of states in the receiver can be less than that of the encoder. This fact seems to have been overlooked, until recently, by other studies on reduced state sequence estimation receiver.

There are other studies[EFBP90] that aim to increase the transmission rate in the DSL by performing the equalization at the transmitter, while this study considers only the receiver signal processing. The receiver considered here is a combined trellis

code/ISI decoder. This has been found to be necessary to avoid the error propagation due to DFE if the channel equalization and decoding are done separately.

As the distance bounds are indicative of the performance of a very complex receiver, simulation studies were carried out to determine the performance of practical receivers. We have determined the range (loop length) of the high bit rate DSL system under stationary and cyclostationary crosstalk interference. Symbol error probability results were obtained for various loop lengths using 8 state M-algorithm receivers. It was observed that under stationary crosstalk conditions even the trellis coded system does not provide the necessary loop coverage in the carrier serving area, with 12 dB of noise margin. But the range exceeds the worst case CSA loop if the crosstalk is cyclostationary. Even a simple DFE receiver has longer loop coverage under cyclostationary interference conditions.

A fractionally spaced equalizer(FSE) provides better performance (SNR) under cyclostationary interference compared to under stationary interference. We use these fractionally spaced forward filters at the front end of the sequence estimators considered above. In addition, the sensitivity of the performance under cyclostationary crosstalk to other system parameters were explored in this thesis. Specifically, the effect of the level of the background white noise and the bandwidth of the crosstalk interference process were studied.

The cyclostationarity of the crosstalk is found to depend on the bandwidth of the lowpass filter at the receiver front end. Hence, it was shown that the performance of the equalizer can be improved by increasing the bandwidth of the lowpass filter. Under stationary noise an increase in the bandwidth of the receive filter would increase the noise, and hence degrade the system noise margin; while the forward filter is able to suppress the cyclostationary interference more effectively by increasing the bandwidth of the lowpass filter. The level of the background white noise was found to have a significant effect on the performance under cyclostationary interference.

To be cost effective the transceiver must be implemented in VLSI. The most complex operations in the receiver are the transversal filter function of the echo canceller and the ISI estimation involved in a DFE or the branch metric calculation of the sequence estimator. A new method for calculating the echo estimate or the ISI was proposed in Chapter 7. This method approximates the long tail of the echo or the end-to-end impulse response by piecewise linear segments to reduce the number of multiplications and additions. Using this method, impulse response lengths of a few hundred baud periods can be covered. The number of multiplications increases only logarithmically in this approach while it increases linearly with the length of the response in a conventional transversal filter. Hence for a given number of operations up to 20 dB reduction in the mean squared error has been obtained. The adaptation behaviour of the piecewise linear filter is studied analytically as well as by simulation. The theoretical and simulation results of the convergence of the adaptive filter was presented. A detailed analysis of the wordlength requirements of the piecewise linear fit filter was also presented. The silicon area estimates of the PLF method and the conventional method for the implementation of the transversal filter were compared.

An efficient VLSI architecture for the implementation of the M-algorithm based sequence estimator is proposed in this thesis. Duplicate state elimination and state metric sorting have been identified as the most time consuming operations in the M-algorithm. Due to limitations on the time available for signal processing in the high bit rate system, a new approach is proposed for these functions. Also the path memory updating is an important functional unit in the sequence estimator. A new method is proposed for this function which eliminates the double buffering required in a straightforward implementation. The computational speed of the processing elements are calculated for an 8 state sequence estimator. An analysis of the wordlength requirements of the different variables in the M-algorithm is also given. The silicon area required for the VLSI implementation is estimated using 1μ CMOS technology. It is shown that an area of 30 sq. mm. may be required for the implementation of

the sequence estimator.

High bit rate transmission systems may need fractionally spaced forward filters to achieve the required performance. Conventional implementation of this filter requires a multibit multiplier. As the silicon area required for a parallel multiplier is large, methods for combining the forward filter with the decimation filter of the oversampled A/D are explored in Chapter 6. This requires cancellation of the echo at the input of the sigma-delta modulator. A new method was proposed for echo cancellation in the oversampled domain. The degree of cancellation was found to be about 44 dB which may not be sufficient for this application. The adaptation behaviour of the oversampled echo canceller was also presented. The combined decimation and forward filter implementation issues were studied. The VLSI implementation areas required for the conventional and oversampled schemes were estimated. The oversampled scheme requires more area due to the memory requirements of the coefficient storage. Further work is required in this area to study the implications of the adaptation of the combined filter.

The overall emphasis of this thesis has been to explore methods for improving the data rate or the transmission range of the digital subscriber loop system while taking into account the limitations of the complexity of their implementation in VLSI.

8.1 Future Work

The simulation results presented in Chapter 4 are obtained on a general purpose computer with floating point representation of the variables. The effect of the finite wordlength of the variables in real hardware must be assessed to get an accurate estimate of the error rate. Also the number of transmitted symbols used in the simulation is only 10^5 . To determine error rates of the order of 10^{-7} much longer sequences must be used. Simulation on a general purpose computer with longer

sequences would take an inordinately long time. Hence a special purpose hardware with multiple digital signal processing microprocessors may be needed to obtain more accurate results. The effect of the number of retained states in the reduced state sequence estimator must also be explored.

There is some concern among researchers about the 12 dB noise margin assumed in this study. The performance achievable with reduced margin must be studied. The performance of the coded system under impulse noise conditions must be studied as it is one of the major sources of impairment.

As the performance under cyclostationary crosstalk is significantly better than under stationary crosstalk, methods for synchronizing the transmitter clocks for all the lines in a cable must be explored.

VLSI architectures proposed in the Chapter 5 must be studied in detail to get an accurate estimate of the silicon area. Some assumptions about the area required for testing and routing of the signals have been made. Detailed design of this VLSI circuit at the logic and circuit level, must be pursued to validate these assumptions. As the path metric grows with the number of received symbols, issues related to scaling of the path metric must be explored.

Implementation of the combined decimation and forward filter using dynamic RAM may be studied in order to determine the reduction in the area requirements. Also the error signal used for the adaptation of the echo canceller may be obtained by filtering the high frequency noise in the sigma-delta modulator output more effectively instead of using a simple counter. This may further improve the echo cancellation.

The computational complexity of the piecewise linear fit method must be compared to other complexity reduction techniques[Abo90][Dav89] to determine their relative merits.

Bibliography

- [Abd89] M. Abdulrahman. Decision feedback equalization with cyclostationary interference for DSL. Master's thesis, Carleton University, 1989.
- [ABG81] S.V. Ahamed, P.R. Bohn, and N.L. Gottfried. A tutorial on two-wire digital transmission in the loop plant. *IEEE Transactions on Communications*, November 1981.
- [Abo90] A. Abousaada. Echo tail canceller based on AIFIR filtering. Master's thesis, University of Ottawa, 1990.
- [AC89] J.T. Aslanis and J.M. Cioffi. Capacity and cutoff rate of the digital subscriber loop with near end crosstalk noise. In *International Conference on Communications*, 1989.
- [Ada86] P.F. Adams. Die size for a high performance 2B1Q line code transceiver for ISDN basic access . *Contribution T1D1.3/86-029 to T1D1 committee*, January 1986.
- [AHM82] O. Aggazzi, D.A. Hodges, and D.G. Messerschmitt. Large scale integration of hybrid method digital subscriber loops. *IEEE Transactions on Communications*, November 1982.
- [AKea89] O.E. Agazzi, T. Koh, and S.S. Haider et al. A digital signal processor for an ANSI standard ISDN transceiver . *IEEE Journal of Solid State Circuits*, December 1989.
- [ANS88] ANSI T1-601-1988. *American National Standards Institute, 'Integrated Services Digital Network (ISDN) Basic Access Interface for use on metallic loops for application on the network side of the NT (Layer 1 Specification)'*, September 1988.

- [ATT89] ATT contribution to T1E1 committee, T1E1.4/89-151. *High rate DSL line code - proposal for coded QAM*, June 1989.
- [Bat68] K.E. Batchler. Sorting networks and their application. In *Proc. of AFIPS Spring joint Computer Conference*, AFIPS Press , 1968.
- [Ben58] W.R. Bennett. Statistics of regenerative digital transmission. *Bell Systems Technical Journal*, November 1958.
- [Ber87] J.W. Bergmans. On the design of smearing filters for ISDN transmission systems. *Philips Journal of Research*, pages 172-208, 1987.
- [Big84] Ezio Biglieri. High-level modulation and coding for nonlinear satellite channels. *IEEE Transactions on Communications*, May 1984.
- [CA85] S.A. Cox and P.F. Adams. An analysis of digital transmission techniques for the local network. *British Telecom Technical Journal* , July 1985.
- [Can85] J.C. Candy. A use of double integration in sigma delta modulation. *IEEE Transactions on Communications*, March 1985.
- [CD89] J. Cioffi and G.P. Dudevoir. Data transmission with mean-squared partial response. In *Conference record of Globecom*, 1989.
- [CE89] P.R. Chevillat and E. Eleftheriou. Decoding of trellis encoded signals in the presence of intersymbol interference and noise. *IEEE Transactions on Communications*, July 1989.
- [CGH⁺89] R.P. Colbeck, R.A. Gervais, G. Hunt, A.T. Ahdab, and C.W. Kurowski. A Single-Chip 2B1Q U-Interface transceiver . *IEEE Journal of Solid State Circuits*, December 1989.
- [CGS83] J.C. Campbell, A.J. Gibbs, and B.M. Smith. The cyclostationary nature of crosstalk interference from digital signals in multipair cable - Part I & II. *IEEE Transactions on Communications*, May 1983.
- [Cla85] B.R. Clarke. The time-domain response of minimum phase networks. *IEEE Transactions on Circuits and Systems*, November 1985.
- [CM84] A.R. Calderbank and J.E. Mazo. A new description of trellis codes. *IEEE Transactions on Information Theory*, November 1984.

- [Cod] Codex. *Codex 2600 series high-speed modems application manual*.
- [Coo89] C.G. Cook. Two stage data echo cancellation utilizing orthonormal functions for echo tail cancellation. Master's thesis, Carleton University, February 1989.
- [CTC90] J.S. Chow, J.C. Tu, and J.M. Cioffi. A computationally efficient adaptive transceiver for high speed digital subscriber lines. In *International Conference on Communications*, 1990.
- [CY] C.Y. Chang and K. Yao. Systolic array architecture for the Viterbi decoding algorithm. *IEEE Press VLSI Signal Processing II*.
- [Dav89] G. Davidson. Reduced complexity echo cancellation using orthonormal functions. Technical report, Department of Systems and Computer Engineering, Carleton University, March 1989.
- [DHH89] A. Duel-Hallen and C. Heegard. Delayed decision feedback sequence estimation. *IEEE Transactions on Communications*, May 1989.
- [EA90] M. Elder and E. Arnon. Coefficient reduction in high rate DSL. In *IEEE HDSL Workshop '90*, March 1990.
- [EF89] M.V. Eyuboglu and G.D. Forney. Trellis precoding. Preprint of the paper submitted to the IEEE Symposium on Information Theory, 1989.
- [EF90] V.M. Eyuboglu and G.D. Forney. Trellis precoding: Combined coding, precoding and shaping for intersymbol interference channel. In *IEEE Int'l Symposium on Information Theory*, 1990.
- [EFBP90] M.V. Eyuboglu, G.D. Forney, L. Brown, and G. Pottie. Trellis precoding for high-rate transmission on subscriber loops. In *IEEE HDSL Workshop '90*, March 1990.
- [EMS79] R.K. Even, R.A. McDonald, and H. Seidel. Digital transmission capability of the loop plant. In *International Conference on Communications*, 1979.
- [EQ89] V.M. Eyuboglu and S.U.H. Qureshi. Reduced state sequence estimation for coded modulation on intersymbol interference channels. *IEEE Journal on Selected Areas in Communications*, August 1989.

- [Fal87] D.D. Falconer. Carrierless AM/PM . Course notes, 1987.
- [FC89] G.D. Forney and A.R. Calderbank. Coset codes for partial response channels; or coset codes with spectral nulls. *IEEE Transactions on Information Theory*, September 1989.
- [FLF89] A. Fung, L.S. Lee, and D.D. Falconer. A facility for near end crosstalk measurements on ISDN subscriber loops. In *Conference record of Globecom*, 1989.
- [FM73] D.D. Falconer and F.R. Magee. Adaptive channel memory truncation for maximum likelihood sequence estimation. *Bell Systems Technical Journal*, November 1973.
- [FM79] D.D. Falconer and K.H. Mueller. Adaptive echo cancellation/AGC structures for two-wire full-duplex data transmission. *Bell Systems Technical Journal*, September 1979.
- [FM89] G. Fettweis and H. Meyer. Parallel Viterbi algorithm implementation: Breaking the ACS-Bottleneck. *IEEE Transactions on Communications*, August 1989.
- [For72] G.D. Forney. Maximum likelihood sequence estimation of digital sequences in the presence of intersymbol interference. *IEEE Transactions on Information Theory*, May 1972.
- [For89] G.D. Forney. Trellis shaping. Preprint of the paper presented at the IEEE Information Theory Workshop, 1989.
- [Gal68] R.G. Gallager. *Information Theory and Reliable Communication* . John Wiley and Sons Inc., 1968.
- [GF75] W.A. Gardner and L.E. Franks. Characterization of cyclostationary random signal processes. *IEEE Transactions on Information Theory*, January 1975.
- [GS86] P.G. Gulak and E. Shwedyk. VLSI structures for Viterbi receivers: Part I - General theory and applications, Part II - Encoded MSK Modulation. *IEEE Journal on Selected Areas in Communications*, January 1986.

- [GW81] R.D. Gitlin and S.B. Weinstein. Fractionally-spaced equalization: An improved digital transversal equalizer. *Bell Systems Technical Journal*, February 1981.
- [HCKC89] R.G. Hunt, J.W. Cook, R.H. Kirkby, and N.G. Cole. The potential for high rate digital subscriber loops. In *International Conference on Communications*, 1989.
- [JF87] V. Joshi and D.D. Falconer. Channel capacity bounds for the subscriber loops. In *Pacific Rim Conference on Communication, Computer and Signal Processing*, 1987.
- [JF90] V. Joshi and D.D. Falconer. Sequence estimation techniques for digital subscriber loop transmission with crosstalk interference. *IEEE Transactions on Communications*, September 1990.
- [Jos88] V. Joshi. *Sequence estimation techniques for enhanced digital subscriber loop transmission capability*. PhD thesis, Carleton University, 1988.
- [Kap89] Vivek Kapil. Measurement and characterization of impulse noise on DSL for ISDN application. Master's thesis, Carleton University, 1989.
- [KBHH89] J. Kernhof, M.A. Beunder, B. Hoefflinger, and W. Hass. High speed CMOS adder and multiplier modules for digital signal processing in semi-custom environment. *IEEE Journal of Solid State Circuits*, June 1989.
- [Ket87] J.W. Ketchum. Performance of trellis codes for M-ary partial response. In *Conference Record of Globecom*, 1987.
- [KS90] I. Kalet and S. Shamai. On the capacity of a twisted-wire pair: Gaussian model. *IEEE Transactions on Communications*, March 1990.
- [Law89] V.B. Lawrence. HRDSL line code: Fundamental limits. *T1E1.4/89-103*, March 1989.
- [Lec88] J.W. Lechleider. Feasibility study of very high bit rate digital subscriber lines. *contribution T1E1/88-012 to T1E1 committee*, March 1988.
- [Lec89] J.W. Lechleider. Line codes for digital subscriber lines. *IEEE Communications Magazine*, September 1989.

- [Lec90] J.W. Lechleider. Exploiting crosstalk pair-to-pair correlation in two-pair digital subscriber lines. In *International Conference on Communications*, 1990.
- [LH77] W.U. Lee and F.S. Hill. A maximum likelihood sequence estimation with decision feedback equalization. *IEEE Transactions on Communications*, September 1977.
- [Lin90] D.W. Lin. Minimum mean-squared error echo cancellation and equalization for digital subscriber line transmission: Part I & II. *IEEE Transactions on Communications*, January 1990.
- [LM88] Edward A. Lee and David G. Messerschmitt. *Digital Communication*. Kluwer Academic, 1988.
- [LSW68] R.W. Lucky, J. Salz, and E.J. Weldon. *Principles of data communication*. McGraw-Hill Book Company, 1968.
- [Maz79] J.E. Mazo. On the independence theory of equalizer convergence. *Bell Systems Technical Journal*, May 1979.
- [Mes73] D.G. Messerschmitt. Geometric theory of intersymbol interference: Part I & II. *Bell Systems Technical Journal*, November 1973.
- [Mes86] D.G. Messerschmitt. Design issues in the ISDN U-interface transceiver. *IEEE Journal on Selected Areas in Communications*, November 1986.
- [MFK90] P. Mohanraj, D.D. Falconer, and T.A. Kwasniewski. Baseband trellis coded modulation with combined equalization/decoding for high bit rate digital subscriber loops. In *Conference record of Globecom*, 1990.
- [MJFK89] P. Mohanraj, V. Joshi, D.D. Falconer, and T.A. Kwasniewski. Reduced complexity coding/decoding for high bit rate digital subscriber loops. In *International Conference on Communications*, 1989.
- [MNP86] C. Mogavero, G. Nervo, and G. Paschetta. Mixed recursive echo canceller (MREC). In *Conference record of Globecom*, 1986.
- [MS86] S. Mohan and A. Sood. A multiprocessor architecture for (M,L) algorithm suitable for VLSI implementation. *IEEE Transactions on Communications*, December 1986.

- [MV89] R.A. McDonald and C.F. Valenti. Assumptions for Bellcore HDSL studies. *contribution T1E1.4/89-066 to T1E1 Committee*, March 1989.
- [PG88] J.D. Provence and S.C. Gupta. Systolic arrays for Viterbi processing in communication systems with a time-dispersive channel. *IEEE Transactions on Communications*, October 1988.
- [Pie86] S.B. Pierce. Crosstalk in twisted pair circuits. In *Intl wire and cable symposium*, 1986.
- [Pot90] G.J. Pottie. Comparison of baseband vs passband systems for HDSL. *contribution T1E1.4/90-164 to T1E1.4 Committee*, 1990.
- [QN73] S.U.H. Qureshi and E.E. Newhall. An adaptive receiver for data transmission over time-dispersive channels. *IEEE Transactions on Information Theory*, July 1973.
- [Qur85] S.U.H. Qureshi. Adaptive equalization. *Proceedings of IEEE*, September 1985.
- [RC87] A. Ruiz and J. Cioffi. A frequency domain approach to combined spectral shaping and coding. In *International Conference on Communications*, 1987.
- [RG75] L.R. Rabiner and B. Gold. *Theory and application of digital signal processing*. Prentice-Hall Inc., 1975.
- [RHTA89] A. Rothermel, B.J. Hosticka, G. Troster, and J. Arndt. Realization of transmission-gate conditional sum(TGCS) adders with low latency time. *IEEE Journal of Solid State Circuits*, June 1989.
- [She86] N. Sheshadri. *Error performance of trellis modulation codes on channels with severe intersymbol interference*. PhD thesis, Rensselaer Polytechnic Institute, 1986.
- [Sim90] S.J. Simmons. Breadth-First trellis decoding with adaptive effort. *IEEE Transactions on Communications*, January 1990.
- [Sis90] K. Sistanizadeh. Performance evaluation studies of QAM and PAM schemes on a sample of CSA loops with an ideal DFE. *contribution T1E1.4/90-180 to T1E1.4 Committee*, 1990.

- [SRD⁺87] D. Sallaerts, D.H. Rabaey, R.H. Dierckx, J. Sevenhans, D.R. Haspeslagh, and B.J. De Ceulaer. A Single-Chip U-Interface transceiver for ISDN . *IEEE Journal of Solid State Circuits*, December 1987.
- [SW85] S.D. Stearns and B.D. Widrow. *Adaptive signal processing*. Prentice-Hall Inc., 1985.
- [SWP78] Schneider, Weingart, and Perlman. *An introduction to programming and problem solving with Pascal*. John Wiley and Sons Inc., 1978.
- [SWZ90] M. Sorbara, J.J. Werner, and N.A. Zervos. Carrierless AM/PM. *contribution T1E1.4/90-154 to T1E1 Committee*, 1990.
- [Sze89] K. Szechenyi. On the NEXT and impulse noise properties. In *Conference record of Globecom*, 1989.
- [TCDC89] J.C. Tu, J.S. Chow, G.P. Dudevoir, and J.M. Cioffi. Crosstalk limited performance of a computationally efficient multichannel transceiver for high rate digital subscriber lines. In *Conference Record of Globecom*, 1989.
- [Tex] Texas Instruments. *Standard cell data catalog*.
- [TS85] T. Tjahjadi and W. Steenaart. Adaptive filter realization with a minimum number of multipliers. *IEEE Trans. on Circuits and Systems* , March 1985.
- [TTea89] Y. Takahashi, M. Takahara, and T. Makabe et al. An ISDN echo cancelling transceiver chip set for 2B1Q coded U-Interface. *IEEE Journal of Solid State Circuits*, December 1989.
- [Ung82] G. Ungerboeck. Channel coding with multilevel/phase signal. *IEEE Transactions on Information Theory*, January 1982.
- [V.3a] V.32 - CCITT Contrib. COM-XVII No. T30. *Rapporteur on 9600 bits/s duplex family modem, ' Draft recommendation V.32 for a family of 2-wire, duplex modems operating at data signalling rates of up to 9600 bits/s for use on the general switched telephone network and on leased telephone-type circuits'*.

- [V.3b] V.33 - CCITT Study Group XVII. *Rapporteur on 14 400 bits/s modem, 'Working draft recommending V.33 for a 14 400 bits/s modem standardized for use on point-to-point 4-wire leased telephone-type circuits'.*
- [VdESG79] N.A.M. Verhoeckx, H.C. Van den Elzen, F.A. Snijders, and P.J. Van Gerwin. Digital echo cancellation for baseband data transmission. *IEEE Transactions on Acoustic, Speech, and Signal Processing*, December 1979.
- [VO79] A.J. Viterbi and J.K. Omura. *Principles of Digital Communication and Coding*. McGraw-Hill Book Company, 1979.
- [WC88] C.H. Wei and N.C. Chen. Sigma-delta modulation adaptive digital filters. In *Conference record of International Symposium on Circuits and Systems*, 1988.
- [WE71] S.B. Weinstein and P.M. Ebert. Data transmission by frequency division multiplexing using Discrete Fourier Transform. *IEEE Transactions on Communications*, October 1971.
- [WE85] N. Weste and K. Eshragian. *Principles of CMOS VLSI Design - A systems perspective*. Addison-Wesley Publishing Company, 1985.
- [Wer89] J.J. Werner. Impulse noise on the loop plant. In *International Conference on Communications*, 1989.
- [YFS90] Yu-Dong Yao, D.D. Falconer, and A.U.H. Sheikh. PAM and QAM for high rate digital subscriber loop transmission. Technical report, Department of Systems and Computer Engineering, Carleton University, April 1990.

Appendix A

Optimal piecewise linear fit filter coefficients

In this appendix we describe the method for obtaining the optimal (in the mean squared sense) piecewise linear fit filter coefficients for a given impulse response. Let the impulse response of the transversal filter extend over $(N_I + 1)$ symbol intervals. Let this response be divided into 'I' segments as shown in Figure A.1.

The index of the end points of the segments are $N_0, N_1, \dots, N_I$ with $N_0 = 0$. Hence, the segment lengths $K_i, i = 0, 1, 2, \dots, (I - 1)$ are given by

$$K_i = N_{i+1} - N_i \quad (\text{A.1})$$

A piecewise linear approximation to the segment 'i' can be written as

$$\begin{aligned} h(N_i + j) &= \frac{(K_i - j)}{K_i} h(N_i) + \frac{j}{K_i} h(N_{i+1}) \\ 1 \leq j \leq K_i \\ i &= 0, 1, 2, \dots, (I - 1) \end{aligned} \quad (\text{A.2})$$

Let the actual response at index $(N_i + j)$ be represented by

$$g(N_i + j) = g_j(i) \quad (\text{A.3})$$

Figure A.1: Segmentation of the impulse response

The mean squared error between the actual and approximated response is then expressed by

$$mse = \sum_{i=0}^{I-1} \sum_{j=1}^{K_i} \{h(N_i + j) - g_j(i)\}^2 \quad (A.4)$$

The mean squared error can be represented conveniently in a matrix form as shown below. Let us define two quantities

$$a_{ij} = \frac{(K_i - j)}{K_i} \quad (A.5)$$

$$b_{ij} = \frac{j}{K_i} \quad (A.6)$$

Using these definitions equation A.2 can be written as

$$h(N_i + j) = a_{ij}h(N_i) + b_{ij}h(N_{i+1}) \quad (A.7)$$

Substituting A.7 in A.4 we obtain

$$mse = \sum_{i=0}^{I-1} \sum_{j=1}^{K_i} [a_{ij}^2 h(N_i)^2 + b_{ij}^2 h(N_{i+1})^2 + 2a_{ij}b_{ij}h(N_i)h(N_{i+1}) - 2a_{ij}h(N_i)g_j(i) - 2b_{ij}h(N_{i+1})g_j(i) + g_j(i)^2] \quad (A.8)$$

In the matrix form

$$mse = H^T R H - 2H^T B + |g|^2 \quad (A.9)$$

where the matrices H, R and B are obtained as follows.

Substituting $i = 0, 1, \dots, i, \dots, (I-1)$ in A.8 we obtain the following terms in the expansion

$i = 0 :$

$$\sum_{j=1}^{K_0} [b_{0j}^2 h(N_1)^2 - 2b_{0j} h(N_1) g_j(0) + g_j(0)^2]$$

since $h(N_0) = 0$.

$i = 1 :$

$$\begin{aligned} \sum_{j=1}^{K_1} [a_{1j}^2 h(N_1)^2 + b_{1j}^2 h(N_2)^2 + 2a_{1j} b_{1j} h(N_1) h(N_2) - 2a_{1j} h(N_1) g_j(1) \\ - 2b_{1j} h(N_2) g_j(1) + g_j(1)^2] \\ \vdots \end{aligned}$$

$1 < i < (I-1) :$

$$\begin{aligned} \sum_{j=1}^{K_i} [a_{ij}^2 h(N_i)^2 + b_{ij}^2 h(N_{i+1})^2 + 2a_{ij} b_{ij} h(N_i) h(N_{i+1}) - 2a_{ij} h(N_i) g_j(i) \\ - 2b_{ij} h(N_{i+1}) g_j(i) + g_j(i)^2] \\ \vdots \end{aligned}$$

$i = I-1 :$

$$\begin{aligned} \sum_{j=1}^{K_{I-1}} [a_{I-1,j}^2 h(N_{I-1})^2 + b_{I-1,j}^2 h(N_I)^2 + 2a_{I-1,j} b_{I-1,j} h(N_{I-1}) h(N_I) \\ - 2a_{I-1,j} h(N_{I-1}) g_j(I-1) - 2b_{I-1,j} h(N_I) g_j(I-1) + g_j(I-1)^2] \end{aligned}$$

Taking the constants $h(N_1), h(N_2)$ etc., outside of the summation in the above expressions and summing them all, we get

$$\begin{aligned}
mse = & h(N_1)^2 \left(\sum_{j=1}^{K_0} b_{0j}^2 + \sum_{j=1}^{K_1} a_{1j}^2 \right) + h(N_1)h(N_2) \sum_{j=1}^{K_1} a_{1j}b_{1j} \\
& + h(N_1)h(N_2) \sum_{j=1}^{K_1} a_{1j}b_{1j} + h(N_2)^2 \left(\sum_{j=1}^{K_1} b_{1j}^2 + \sum_{j=1}^{K_2} a_{2j}^2 \right) + h(N_2)h(N_3) \sum_{j=1}^{K_2} a_{2j}b_{2j} \\
& \cdot \\
& \cdot \\
& \cdot \\
& + h(N_{I-1})h(N_I) \sum_{j=1}^{K_{I-1}} a_{I-1,j}b_{I-1,j} + h(N_{I-1})^2 \left(\sum_{j=1}^{K_{I-2}} b_{I-2,j}^2 + \sum_{j=1}^{K_{I-1}} a_{I-1,j}^2 \right) \\
& + h(N_I)h(N_{I-1}) \sum_{j=1}^{K_{I-1}} a_{I-1,j}b_{I-1,j} \\
& + h(N_I)h(N_{I-1}) \sum_{j=1}^{K_{I-1}} a_{I-1,j}b_{I-1,j} + h(N_I)^2 \left(\sum_{j=1}^{K_{I-1}} b_{I-1,j}^2 \right) \\
& - 2h(N_1) \left(\sum_{j=1}^{K_0} b_{0j}g_j(0) + \sum_{j=1}^{K_1} a_{1j}g_j(1) \right) \\
& - 2h(N_2) \left(\sum_{j=1}^{K_1} b_{1j}g_j(1) + \sum_{j=1}^{K_2} a_{2j}g_j(2) \right) \\
& \cdot \\
& \cdot \\
& \cdot \\
& - 2h(N_{I-1}) \left(\sum_{j=1}^{K_{I-2}} b_{I-2,j}g_j(I_2) + \sum_{j=1}^{K_{I-1}} a_{I-1,j}g_j(I-1) \right) \\
& - 2h(N_I) \left(\sum_{j=1}^{K_{I-1}} b_{I-1,j}g_j(I-1) \right) \\
& + |\underline{g}|^2
\end{aligned} \tag{A.10}$$

Defining the matrices H, R and B as shown below

$$H = \begin{bmatrix} h(N_1) \\ h(N_2) \\ \cdot \\ \cdot \\ \cdot \\ h(N_I) \end{bmatrix} \quad (\text{A.11})$$

and

$$R = \begin{bmatrix} R_{11} & R_{12} & 0 & 0 & \cdot & \cdot & 0 \\ R_{21} & R_{22} & R_{23} & 0 & \cdot & \cdot & 0 \\ 0 & R_{32} & R_{33} & R_{34} & 0 & \cdot & 0 \\ & & \cdot & \cdot & \cdot & \cdot & \\ & & \cdot & \cdot & \cdot & \cdot & \\ & & & \cdot & R_{I-1,I-1} & R_{I-1,I} \\ & & & & R_{I,I-1} & R_{I,I} \end{bmatrix} \quad (\text{A.12})$$

and

$$B = \begin{bmatrix} B_1 \\ B_2 \\ \cdot \\ \cdot \\ \cdot \\ B_I \end{bmatrix} \quad (\text{A.13})$$

where

$$R_{ii} = \sum_{j=1}^{K_{i-1}} b_{i-1,j}^2 + \sum_{j=1}^{K_i} a_{ij}^2 \quad \text{for } 1 \leq i \leq I-1$$

$$R_{i,i+1} = \sum_{j=1}^{K_i} a_{ij} b_{ij} = R_{i+1,i}$$

and

$$B_i = \sum_{j=1}^{K_{i-1}} b_{i-1,j} g_j(i) + \sum_{j=1}^{K_i} a_{i,j} g_j(i)$$

$$B_I = \sum_{j=1}^{K_{I-1}} b_{I-1,j} g_j(I-1)$$

the equation A.10 can be written in a compact form as

$$mse = H^T R H - 2H^T B + |g|^2 \quad (\text{A.11})$$

The optimum coefficient matrix H_{opt} that minimizes the mean squared error can be obtained by solving the set of linear equations resulting from the differentiation of the R.H.S of the equation A.14 with respect to H. It has been shown[SW85] that H_{opt} is given by

$$H_{opt} = R^{-1} B \quad (\text{A.15})$$

Appendix B

Convergence behaviour of the piecewise linear fit filter

Appendix A described the determination of the optimal piecewise linear fit filter coefficients $\{h(i)\}$ under the assumption that the impulse response to be approximated is known *a priori* . In a real system, the impulse response coefficients must be determined adaptively from the received signal.

The most common algorithm for the adaptation of the transversal filter is the LMS algorithm. A modified form of the LMS algorithm was proposed for the adaptation of the piecewise linear fit(PLF) filter in Section 7.5. The theoretical convergence behaviour of the PLF filter is discussed here.

The LMS algorithm for the adaptation of the PLF filter coefficients is described by the equation

$$\underline{h}(n+1) = \underline{h}(n) - A \cdot \underline{S}(n) (y(n) - r(n)) \quad (\text{B.1})$$

where ' A ' is the diagonal matrix of step sizes $(\alpha_1, \alpha_2, \dots, \alpha_I)$
i.e.,

$$A = \begin{bmatrix} \alpha_1 & 0 & . & . & 0 & 0 \\ 0 & \alpha_2 & 0 & . & . & 0 \\ 0 & . & . & . & . & . \\ 0 & . & . & . & . & . \\ 0 & . & . & . & . & \alpha_I \end{bmatrix} \quad (B.2)$$

The step size α_i is chosen to be inversely proportional to the segment length K_i . This was found to give a faster convergence.

$y(n)$ is the piecewise linear filter output given by

$$y(n) = \underline{S}^T(n) \cdot \underline{h}(n) \quad (B.3)$$

where $\underline{S}(n)$ is the vector of averaged signal values of the segments as defined in 7.17 and $r(n)$ is the received signal given by

$$r(n) = \underline{g}^T \cdot \underline{a}_n + \nu_n \quad (B.4)$$

The tap coefficient error vector at time 'n' can be expressed as

$$\underline{E}(n) = \underline{h}(n) - \underline{h}_{opt} \quad (B.5)$$

where $\underline{h}_{opt} = R^{-1}B$ are the optimal coefficients as described in appendix A.

Using the matrix representation of the error vector[Qur85], the evolution of the error vector in time can be represented by the difference equation

$$\underline{E}(n+1) = (I - A\underline{S}(n)\underline{S}(n)^T)\underline{E}(n) - A\underline{S}(n)[\underline{S}(n)^T\underline{h}_{opt} - r(n)] \quad (B.6)$$

The excess mean squared error is given by

$$\xi = \langle \underline{E}(n)^T R \underline{E}(n) \rangle \quad (B.7)$$

where $R = \langle \underline{S}(n)\underline{S}(n)^T \rangle$. R is the correlation matrix, determined by the set of segment lengths K_i , $i = 0, 1, \dots, (I-1)$, described in appendix A.

Using the independence theory of equalizer convergence[Maz79], Falconer showed that the excess mean squared error at time ' n ' is given by

$$\text{excess mse}(n) = \sum_{i=1}^I x_{ii}(n) + MSE \quad (B.8)$$

where the x_{ii} are the diagonal elements of the matrix ' X ' given by

$$X(n+1) = X(n) - X(n) \Theta - \Theta X(n) - \Theta^2(\text{tr}(X(n)) + MSE) \quad (B.9)$$

In this equation, Θ is a diagonal matrix whose diagonal elements are the eigen values $\{\theta_i\}$ of the positive definite symmetric matrix $C = \sqrt{R} \cdot A \cdot \sqrt{R}$.

MSE is the minimum mean squared error obtained when the PLF filter coefficients are equal to the optimal coefficients $\underline{h}_{opt}(n)$.

Since we are only interested in the diagonal elements of ' X(n) ', we can rewrite the difference equation in the form

$$\underline{x}(n) = Q * \underline{x}(n-1) + \theta^2 * MSE \quad (B.10)$$

Here $\underline{x}(n) = [X_{11}, X_{22}, \dots, X_{II}]$,

$$Q = \begin{bmatrix} (1-\theta_1)^2 & \theta_1^2 & . & . & . & \theta_1^2 \\ \theta_2^2 & (1-\theta_2)^2 & \theta_2^2 & . & . & \theta_2^2 \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ \theta_I^2 & . & . & . & . & (1-\theta_I)^2 \end{bmatrix} \quad (B.11)$$

and θ is a vector whose elements are the eigen values $\{\theta_i\}$. The initial value of $\underline{x}(n)$ is given by the diagonal elements of the matrix

$$(V^T \cdot \underline{h}_{opt} \cdot \underline{h}_{opt}^T \cdot V) \quad (B.12)$$

where ' V ' is the unitary matrix whose columns are the eigen vectors of ' R '.

The theoretical convergence behaviour is obtained by solving the difference equation B.10 iteratively for the required number of time steps.

END

0/2·03·92

FIN