This book was bound by

Badminton Press

Brittania Works, 942 Melton Road, Thurmaston
Telephone: Leicester (0533) 696799

Print Finishing - Book binding - Loose Leaf Binding - Art Folders - Printing - Stationery

PREDICTION TECHNIQUES APPLIED TO DIFFERENTIAL PULSE CODE MODULATION SYSTEMS FOR ENCODING SPEECH SIGNALS

BY

CUMHUR CENGIZ EVCI, M.I.E.E.E.,

B.Sc. (Birmingham University, England)

M.Sc. (Loughborough Univ. of Technology, England)

A Doctoral Thesis submitted in partial fulfilment of the requirements for the award of Doctor of Philosophy of the Loughborough University of Technology.
March, 1982.

SUPERVISOR: Costas S. Xydeas, D.E., M.Sc., Ph.D., M.I.O.A.,
Department of Electronic and Electrical Engineering.

SYNOPSIS

Differential pulse-code modulation (DPCM) is an efficient digitization technique for encoding speech signals. The two principal components of a DPCM system are the quantizer and the predictor, either or both of which can be adaptive. This thesis describes the investigation of various differential pulse-code modulation systems. Initially, fixed, i.e. time-invariant, predictors using long-term signal statistics of the speech signal are examined. The performance of such a predictor in a DPCM system having a fixed quantizer is studied. Then by replacing the fixed predictor with one whose coefficients are calculated at fixed time intervals, the performance of the encoder, in terms of signal to noise ratio (SNR), is improved by 3-5 dB. A further improvement of 2-3 dB in SNR is obtained when an adaptive quantizer is used in the DPCM system. However, the block adaptive predictor requires the transmission of prediction coefficients to the receiving end at the expense of an increase in the channel bandwidth.

In general, gradient prediction techniques update their coefficients every sampling instant using preceeding speech samples and thus the transmission of side information is avoided. The Stochastic Approximation gradient predictor (SAP) technique is analysed and the adaptation rate of its prediction coefficients is shown to be inadequate to follow fast variations in the statistics of a speech signal.

In order to obtain faster convergence to the optimum coefficient values, a novel technique called the Sequential Gradient Estimation

Predictor (SGEP) has been devised and thoroughly investigated. The advantage of the SGEP over SAP is illustrated by computer simulation and it is shown that an improvement of approximately 3 dB in SNR is obtained. Also the ability of SGEP to work efficiently with fewer coefficients is demonstrated.

SGEP and SAP are subsequently employed in DPCM systems using both fixed and adaptive quantizers. Although, SGEP performs approximately 3-4 dB better than SAP for wide range of transmission bit rates, both provide substantial improvement compared with linear DPCM having a leaky integrator in the feedback loop.

The performance of the SGEP and the SAP algorithms is also examined when a forward block adaptive quantizer (AQF) is employed in the DPCM system and the advantage of SGEP over SAP is shown to be 2-3 dB. Furthermore, the encoding efficiency of the above DPCM systems is investigated in the presence of channel errors. For a wide range of transmission bit-error rates, the increased tolerance to noise of SGEP compared with SAP is demonstrated.

Finally, a correlation switched predictor (CSP) having two coefficients is described for use in a DPCM-AQF codec. This predictor divides the range of the correlation coefficient of the speech signal into zones, and as the correlation coefficient changes zones the predictor coefficients undergo a substantial modification. By this method the adaptation rate of the predictor is improved, particularly during transitions between unvoiced and voiced sounds.

ACKNOWLEDGEMENTS

It gives me great pleasure to express my sincere gratitude to my supervisor, Dr. Costas Xydeas, for his guidance and inspiration throughout this research. His invaluable knowledge in the field of speech digitization helped to initiate and develop the work presented here.

I would like to thank my former supervisor, Dr. Raymond Steele of Bell Laboratories, U.S.A., for his continuous encouragement and constructive criticisms which tremendously influenced the course of this study.

I am thankful to Professor J.W.R. Griffiths, and Professor I.R. Smith, former and current Heads of the Department of Electronic and Electrical Engineering, Loughborough University of Technology, for providing research facilities.

I owe special thanks to all my colleagues particularly Dr. H. Gharavi of Auckland University, New Zealand, P.J. Patrick and A.F. Erwood of Loughborough University of Technology, for the numerous fruitful discussions we have had.

Many thanks to my friends Dr. E.S. Tez, O.M. Abdel Gadir of Loughborough University of Technology and Miss N. Küçükkoca of the London School of Economics, for their moral assistance.

I am grateful to some members of staff at École Supérieure d'Ingénieurs, Chambre de Commerce et d'Industrie de Paris, France, where I worked as a conseiller d'étude (1980-1981), for their patience during

the preparation of the manuscript.

Many thanks are due to Mr. G.P. Gerrard of the Computer Centre, Loughborough University of Technology, for his kindly assistance in computer programming and Miss J.M. Briers for her forebearance in typing this thesis.

Finally, my deepest gratitude is due to my family for their unconditional love and financial support which has made my study possible. The pictures of my niece, *Melis*, were a source of joy when I was under extreme pressure.

To her I dedicate this thesis.

GLOSSARY OF ABBREVIATIONS AND SYMBOLS

TDM	:	Time Division Multiplexing
FDM	:	Frequency Division Multiplexing
LSI	:	Large Scale Integration
PCM	:	Pulse Code Modulation
DPCM	:	General Differential Pulse Code Modulation
DM	:	Delta Modulation
SNR	:	Signal-to-Noise Ratio
SAP	:	Stochastic Approximation Prediction
SGEP	:	Sequential Gradient Estimation Prediction
AQF	:	Adaptive Quantization with Forward Stepsize
CSP	:	Correlation Switched Predictor
SNRSEG	:	Segmented Signal-to-Noise Ratio
VUBS	:	Voiced/Unvoiced Band Switching
$f_1, f_2, \ldots$	:	Formant Frequencies
$x(t)$	:	Input Speech Waveform
$u(t)$	:	Impulse Response of the vocal tract
$e(t)$	:	Excitation Function
$U(\omega)$	:	Fourier Transform of u(t)
$E(\omega)$	:	Fourier Transform of e(t)
$X(\omega)$	:	Spectrum of the speech signal
$FA_1, FA_2, \ldots$	:	Amplitudes of the major formants
DFT	:	Discrete Fourier Transform

IDFT	:	Inverse DFT
c(t)	:	Ceptrum Function
CCD	:	Charged Coupled Devices
LPC	:	Linear Predictive Coding
ARMA	:	Autoregressive Moving Average
H(z)	:	Transfer Function of all-pole model
G_e	:	Amplitude of the excitation
a_k	:	kth coefficient of the linear predictor
δe_i	:	Excitation Pulse at ith instant
N	:	Order of Linear Predictor
$\hat{A}_{opt}$	:	Optimum Prediction Coefficient's vector
<(.)>	:	Time-average of (.)
Φ/C	:	Autocovariance/Autocorrelation matrix
Ψ/C_0	:	Autocovariance/Autocorrelation vector
$\{X_i\}$	:	Input Speech Sequence
$\{X_e\}$	:	Quantized Amplitude Sequence
$\{L_i\}$	:	Binary Output Sequence
$\{X_d\}$	:	Decoded Amplitude Sequence
CI	:	Information Capacity, bits/sec.
f_s	:	Sampling Rate of speech signals
f_c	:	Signal Bandwidth
b	:	Number of bits/sample
$\hat{x}(t)$	:	Recovered Speech
N_Q	:	Number of Quantizer levels
Δ	:	Fixed Quantizer Stepsize

$P_d(x)$	:	Probability Density Function of x
σ_q^2	:	Mean-square value of the quan. noise
V	:	Overload limit of the quantizer
d_L	:	Loading Factor
σ_x	:	r.m.s. value of the input
DR(dB)	:	Dynamic Range in dB
Log-Q	:	Logarithmic Quantizer
Opt-Quant	:	Optimum Quantizer
$H_A(x)$	:	Characteristic of A-law Compander
$\hat{x}_i$	:	Locally decoded sample at ith instant
y_i	:	Predicted sample at ith instant
NS	:	Total number of sampling intervals
σ_e^2	:	Mean-square value of input signal to the quantizer in DPCM
c_1	:	First Shift Correlation coefficient
Q	:	SNR of the quantizer
G	:	Signal-to-prediction error ratio
W	:	Block-size
FBA	:	Forward Block Adaptation
APC	:	Adaptive Predictive Coding
M_p	:	Typical pitch period
$P_1(z)$	:	Long-term Predictor
$P_2(z)$	:	Short-term Predictor
ADPCM	:	Adaptive DPCM in general
PSADPCM	:	Pitch Synchronous ADPCM
$a_{i+1,k}$	:	kth prediction coefficient at (i+1)th instant

AQ	:	Adaptive Quantizer
Δ_i	:	Stepsize of AQ at ith sampling instant
$\Delta_{max}, \Delta_{min}$	:	Maximum, minimum stepsize of the quantizer
β_q	:	"Leakage" constant for adaptive quantization
M(.)	:	Time-invariant stepsize multiplier
AQJ	:	Jayant's Adaptive Quantizer
PCQ	:	Pitch Compensating Quantizer
VLC	:	Variable Length Coding
NFC	:	Noise Feedback Coder
FR	:	Ratio of the sampling frequency, f_s to twice cut-off frequency f_c
SNRP	:	Peak SNR
DSM	:	Delta-Sigma Modulator
HIDM	:	High Information Delta Modulation
FCFDM	:	First Order Constant Factor DM
SCFDM	:	Second Order Constant Factor DM
SVDM	:	Song Voice DM
Δ_0	:	Minimum Stepsize of the DM
CVSD	:	Continuously Variable Slope DM
SBC	:	Sub-band Coding
TC	:	Transform Coding
VEV	:	Voice-Excited Vocoder
TES	:	Time Encoded Speech
wb_i	:	Window Function, $1<i<W$
DPCM(N,b)	:	DPCM having Nth order fixed predictor and b bits fixed quantizer
NB	:	Total number of speech blocks

ADPCM(N,b)	:	DPCM having Nth order sequential predictor and b bits fixed quantizer
DPCM(N,b)-AQJ/AQF	:	DPCM having Nth order fixed predictor and AQJ or AQF
ADPCM(N,b)-AQJ/AQF	:	DPCM having Nth order sequential predictor and AQJ or AQF
FBADPCM(N,b)-AQJ/AQF	:	DPCM having Nth order forward block adaptive predictor and AQJ or AQF
MSB	:	Most Significant Bit
LSB	:	Least Significant Bit
BER	:	Bit Error Rate
FU	:	Error function to be minimized
g,h,c,A,B,D,M	:	Adaptation Constants of predictor
$\hat{W}_i^0$	:	N dimensional column vector of zero mean, while noise terms with unknown stationary variance
$G_{KAL}(i)$	:	Kalman Filter Gain
V_{i-1}	:	Predictor Coefficient Error variance matrix at (i-1)th instant
$\zeta(x_i,N)$	:	Function of x_i and N
$P_i(x)$	:	Variable adaptation parameter
K	:	Diagonal matrix whose elements are $1^{-\alpha},2^{-\alpha},\ldots,N^{-\alpha}$
$\hat{A}_i$	:	Vector controlling SGEP algorithm at ith instant
$s_{i,k}$	:	Amount by which each coefficient in SGEP is altered
$\tilde{S}$	:	Fixed vector whose elements are $s_{i,k}$, k=1,2,...,N
SBAP	:	Sliding Block Autocorrelation Predictor
$e_{i,s}$	:	Prediction error resulting from SBAP at ith instant
$\hat{A}_s$	:	Optimum Set of Coefficient vector due to SBAP

$\hat{\gamma}_i$: Difference vector

$||\hat{\gamma}_i||$: The Norm of vector $\hat{\gamma}_i$

Γ_i : Difference Matrix in SGEP Algorithm

A_i : Column Matrix

S_i : Fixed Diagonal Matrix in SGEP Algorithm

$\hat{\theta}_i$: Coefficient Vector in Cascaded-Predictor Structure

ANC : Adaptive Noise Cancelling

SNRSEGIM : SNRSEG IMprovement factor in ANC

FFOP : Fixed First-Order Predictor

FSOP : Fixed Second-Order Predictor

DPCM-AQJ-FFOP : DPCM codec employing AQJ and FFOP

DPCM-AQF-FFOP : DPCM codec employing AQF and FFOP

ADPCM-AQF-SAP/SGEP : DPCM codec employing AQF and SAP/SGEP

R_i : Quantization level number at ith instant

$\hat{e}_i$: Output Quantization level

σ_d : r.m.s. value of the difference between adjacent samples

α_q : Stepsize optimizing coefficient

w,v : Variables

h_e : Probability of getting at least 2 bits correct

H_b : Probability of getting any bit received in error

Z : Number of Thresholds

δc : Zone size

TR_j : Threshold levels for correlation coefficients

ADPCM-AQF-CSP(Z+1)-FSOP

: DPCM-AQF-FSOP with (Z+1) point CSP - Z-order CSP

ADPCM-AQF-CSP(Z+1)-SGEP	:	ADPCM-AQF-SGEP with (Z+1)-point CSP - Z-order CSP
SFP	:	Switched Fixed Predictor
SGEP-S-FSOP	:	Switched SGEP and FSOP
WBS	:	Wideband Signal
BLS	:	Bandlimited Signal
NPSS	:	Narrowband Processed Speech Signal

TABLE OF CONTENTS

PAGE

CHAPTER I

DIGITAL SPEECH COMMUNICATIONS - THESIS OVERVIEW

1.1 BACKGROUND

There are various forms of communication that appear to exist between animals but, only in human beings has this developed into "*speech*" which is possibly the most valuable attribute that man possesses. Although, manual signals, i.e., movements of the features and the limbs play an important role in the process of communication, these are considered secondary to speech. Manual signals alone are insufficient to illustrate abstract notions hence, only through speech we can convey intelligible arguments and information that is not easily reproducible by other forms of communication.

A principle of "*speaking clearly*"[1] exists in all sorts of speech communication. It involves, a) choosing words which convey the message without the need for further explanation, b) presenting information in a logical order so that the listener may follow what is being said without ambiguity and c) making the sound of speech loud enough so that words are audible to the listener.

There is a limit to the volume of sound waves the human vocal apparatus can produce and correspondingly, the distance is restricted to that over which the acoustic transmission is audible. Even ancient man had supplemented acoustic transmission of messages by using fire, smoke and flags.[2]

During the man's technological development of communication, he learnt about the phenomenon of electric current. This led to the invention of the telephone, the first electrical system for the trans-

mission of speech signals over a long distance. The telephone employs electro-acoustical transducers which can convert an acoustic signal to an electrical one or vice versa. Thus, the acoustic speech signal, after being converted into an electrical form, is transmitted to the receiver where it is converted back to its acoustical form. The goal to be achieved by such a system is the accurate reproduction of the speech signal, at the output of the receiver, in a cost effective way.

Several modulation techniques have been developed in an effort to produce the "*efficient*" communication system. In general, the communication systems are divided into two groups, that is, analogue and digital systems.[3] Analogue transmission systems require to be linear since any non-linearity causes distortion. A digital system does not need to be linear as the signal only consists of a number of discrete levels. In addition, the recent evolution of the semi-conductor technology from transistors to microprocessors has made digital systems much more preferable to analogue ones. Digital communication systems are extensively used today by several countries around the world in commercial telephony and in military and law enforcement applications.

Examples of modern digital communication systems include, a) Digital transmission of telephone speech signals where 30 different speech conversations are simultaneously transmitted using time division multiplexing (TDM) and Pulse Code Modulation (PCM) methods, at a transmission rate of 2.048 Mbits/sec, b) Digital transmission of television signals where PCM is used for the analogue to digital conversion and the transmission bit rate is approximately 120 Mbits/sec.[4]

1.2 DIGITAL SPEECH COMMUNICATION

Before presenting organization of this thesis and the summary of the main results, it is worthwhile to pause and recap the main factors in favour of digital speech systems over analogue ones. They are as follows:[5-9]

1. In digital speech communication, the information is transmitted over long distances without degradation of the speech quality. This occurs because digital signals can be regenerated, i.e., retimed and reshaped along the transmission path. In contrast, analogue transmission systems tend to accumulate noise and other impairments with distance.
2. Digital terminals are cheaper than analogue ones, i.e., cheaper filters and time-shared digital circuitry.
3. An easy way of multiplexing and demultiplexing signals is always required by a flexible speech communication system. Digital methods allow the TDM process to be applied in a simple and economical way to telephone transmission lines. This is in contrast with the frequency division multiplexing (FDM) method in analogue transmission systems, where complex and expensive filters are required.
4. Various types of signals which are encoded in a digital format can be transmitted over the same channel. Therefore, signals such as video, computer data and facsimile data can be handled together with speech.
5. Digital signals are well-favoured by today's device technology. Until recently, the transistor made digital transmission viable,

today Large Scale Integration, LSI, technique makes digital switching attractive and results in compact and economic equipment. In addition the signal processing techniques, such as bit-rate reduction methods and encryption are amenable to digital device technology evolution.

6. Due to item (5), maintenance is simplified. In service performances, monitoring is easy, protection switching and off-line fault isolation are affordable.
7. Digital speech provides the possibility of voiced communication with computers. Specifically, using speech recognition and speech synthesis procedures, digitization terminals could enable speech communication between the user and the computer.

All these factors recommend the digitization of speech and lead to the investigation of new digitization methods. An efficient speech digitizer is required to possess:[10]

a) A good speech quality at a low transmission bit rate.
b) A simple and therefore economical encoder and decoder design.
c) Robustness to the transmission errors.

However, there is at present no way of satisfying the users with all of these points, and in general, there must be a compromise between the three conflicting requirements.

The relative importance of these attributes depends on the application. In telephony, for example, quality and cost are the major factors in the choice of digitizer while in military applications intelligible speech quality at low-bit rates is often essential.

There are several methods for digitization of speech signals and can be categorized into two main groups, namely: waveform digitizers and parametric digitizers as shown in Figure 1.1.

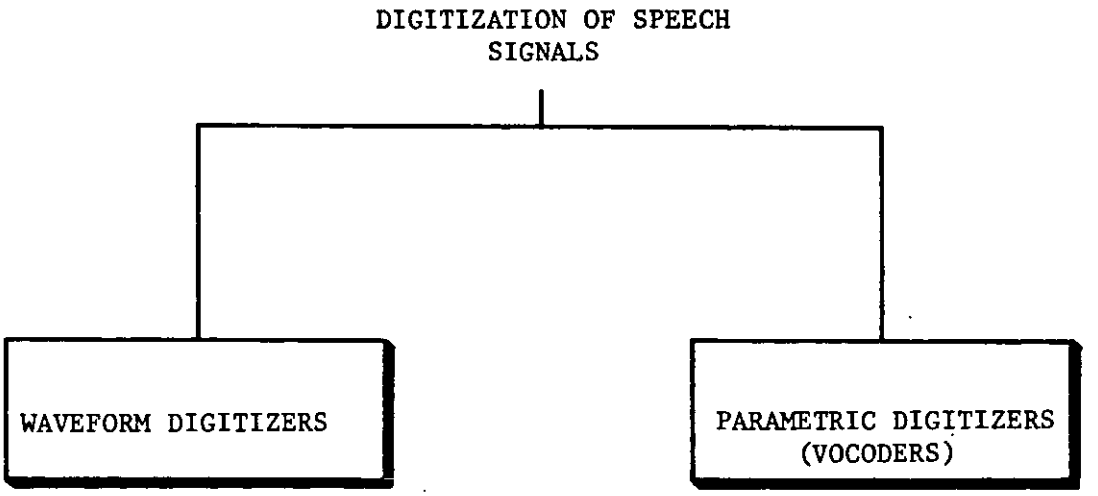


FIGURE 1.1: Methods for Speech Digitization

The concepts used in waveform and parametric digitization are very different. The parametric representation of speech signals, known as analysis-synthesis (vocoder) coding,[11] exploits certain properties of the speech production mechanism. Such systems extract the perceptually important features from the input speech and transmit them to the receiver where a speech production model is used to synthesize the speech

signal. Consequently, any redundancy not affecting the perception is removed. This leads to a dramatic reduction in transmission bit-rate, although vocoders are highly complex and expensive systems.

On the other hand, waveform coding techniques attempt to preserve the waveshape of the original signal. In this case, the speech signal is sampled and each sample is encoded and transmitted. In contrast to vocoders, waveform encoders are simple and inexpensive.

1.3 ORGANIZATION OF THESIS

The remaining chapters in this thesis are summarized as follows:

Chapter II is a brief review of various digital speech coding techniques. It is felt necessary to include this chapter in order to:

a) make the reader familiar both with the terminology and the most important techniques in the field of digital speech coding.

b) establish the direction for the investigations presented in the following chapters.

Firstly, the basic principles of analysis-synthesis techniques are discussed briefly. Then, the attention is focussed on the waveform coders where the main goal is to reproduce, at the output of the decoder, the original analogue signal waveform as accurately as possible. Pulse Code Modulation, PCM, Differential Pulse Code Modulation, DPCM, and Delta Modulation, DM, are reviewed in some detail.

DPCM is the central theme in this thesis and the aim is to design a relatively simple but, efficient DPCM speech digitizer. The performance of DPCM coders depend upon a) the estimation efficiency of the prediction and b) the accuracy of the quantizer used in the system. Adaptive quantization, as used in differential types of speech digitizers, have been extensively studied and several algorithms have been proposed. The importance of the prediction process however, has received less attention compared to quantization and it was felt that our research efforts should be directed towards the development of novel

and efficient speech prediction algorithms.

Chapter III, establishes, in a simple way, the theory of prediction and presents the existing estimation algorithms for the design of time-invariant and block adaptive predictors whose coefficients are calculated at fixed time intervals. The behaviour of these predictors, incorporating a DPCM having both time-invariant and adaptive quantizers, for transmission bit-rates of 16-40 Kb/s, is observed. Further, the effect of the channel errors on signal-to-noise ratio (SNR) values is examined.

Chapter IV introduces the concept of sequentially adaptive linear predictors for speech signals and examines the performance of the Stochastic Approximation Prediction, SAP, algorithm. Then a novel sequentially adaptive algorithm called the "Sequential Gradient Estimation Predictor, SGEP", is proposed. The superiority of SGEP over the SAP is illustrated by waveforms and SNR performance curves. Then, the mathematical analysis of the convergence of the prediction coefficients is examined for SAP and SGEP. The convergence rate of SGEP is proved, experimentally, to be faster than that of the SAP algorithm.

Also, adaptive combinations of SGEP-SAP and SGEP-SGEP predictors are introduced. Finally, the use of the SGEP algorithm is extended to the case where additive acoustic noise in speech signals can be reduced by adaptive noise cancellation, based on a reference noise source.

Chapter V examines the performance of DPCM systems employing the

predictors discussed in Chapter IV and presents computer simulation results. The observation of SNR values reveals that DPCM coders with both adaptive prediction and Jayant's adaptive quantization ADPCM-AQJ, achieve significant improvement over the same coders having fixed predictors, DPCM-AQJ. Further, sequentially adaptive prediction schemes are used in DPCM coders, employing an (AQF) adaptive quantizer with forward transmission of the step size, ADPCM-AQF. In order to reduce the coder complexity only second order predictors are considered. The segmented SNR, SNRSEG, is used as a performance measure. In addition, the encoding efficiency of the above DPCM coder is investigated in the presence of channel errors. For a wide range of transmission bit-error rates, i.e., 16-40 Kb/s, the increased tolerance to noise of SGEP compared with SAP is demonstrated.

Chapter VI starts with the concept of switched predictors, having two coefficients, and describes their use in a ADPCM-AQF coder. These predictors divide the range of the first correlation coefficients, c_1, of the speech signal into zones, and as the correlation coefficient changes zones the predictor coefficients undergo a substantial modification. The use of correlation switched predictors, CSP, in ADPCM-AQF, improves the performance, particularly when speech is transgressing from unvoiced to voiced sounds. Experiments are carried out when the range of correlation coefficient, c_1, is divided into 4 zones, i.e., a 3rd-order CSP, is examined and then, the same idea is extended to an 7 zones, i.e., 7th-order CSP system. Similar techniques are used for SGEP, where the initial values of the adaptive prediction coefficients are modified, every W samples, according to the value of

correlation coefficient, c_1, thereby facilitating a faster coefficient convergence rate. The switched predictors are then used in ADPCM-AQF and the SNRSEG results are presented and compared to DPCM having AQF, but either fixed or block adaptive predictor.

Finally, the proposed DPCM coders are used for the encoding of signals, obtained at the output of the voiced/unvoiced band switching, VUBS, bandwidth compression system.(12)

In Chapter VII, the main results reported in the preceeding sections are discussed. In addition, suggestions for future work are given. Specifically, the SGEP algorithm can be used for pitch extraction since the value of the prediction coefficient of a first order SGEP increases sharply at the onset of each vocal excitation pulse.

Also, the modelling of vocal tract, incorporating pole-zero recursive filter, can be achieved by using a modified SGEP algorithm to update the filter coefficients, such a pole-zero filter will accurately model any possible coupling between the vocal tract and the nasal cavity.

Furthermore, the algorithm of the second-order switched predictor, presented in Chapter VI, can be extended to implement a higher-order switched predictor.

The overall lay-out of the thesis is shown schematically in Figure 1.2.

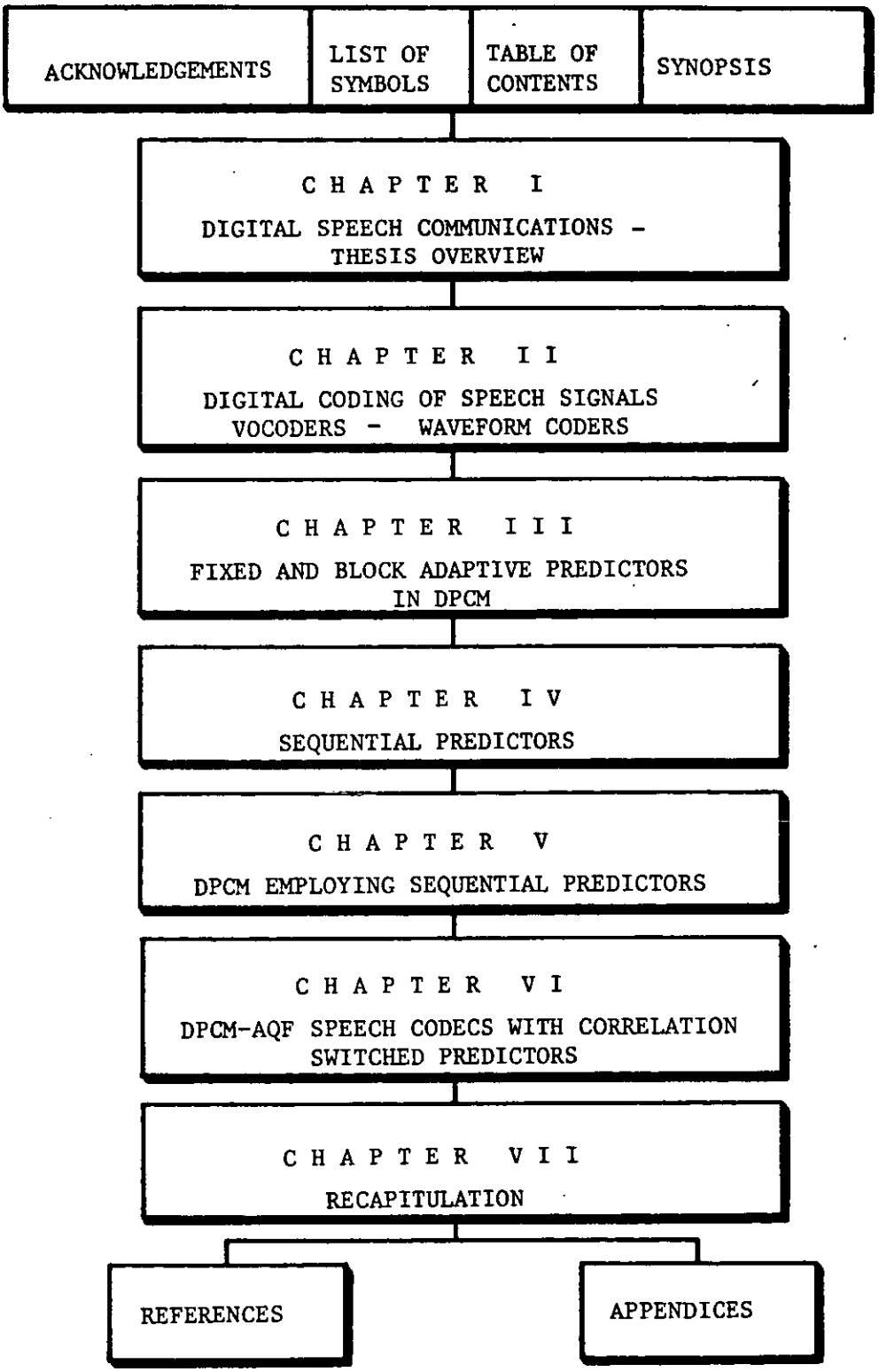


FIGURE 1.2: Thesis Layout

1.4 SUMMARY OF THE MAIN RESULTS

The results obtained in this thesis are summarized as follows:

In Chapter III, initially, fixed, i.e., time-invariant, predictors using long-term speech statistics are examined. The performance of such a predictor in a DPCM system having a fixed quantizer is studied. When, the fixed predictor is replaced with a block adaptive predictor, the SNR performance of the encoder, FBADPCM, is improved by 3-5 dB. A further improvement of 2-3 dB in SNR, is obtained when an adaptive quantizer is used in the DPCM system.

In Chapter IV, the concept of sequentially adaptive linear prediction is introduced. In contrast to block adaptive schemes, these predictors update their coefficients sequentially using preceeding samples and thus the transmission of prediction coefficients as a side information is avoided. The SNR performance of the SAP algorithm is compared with that of the proposed SGEP technique. The advantage of the SGEP over SAP is illustrated by computer simulation and it is shown that an improvement of approximately 3 dB in SNR is obtained. This is attributed to faster convergence of the SGEP towards the "*optimum*" coefficients which are obtained from the sliding-block autocorrelation predictor (SBAP). The SBAP coefficients are derived using the autocorrelation method computed over a length of W samples with the important exception that the analysis window is shifted only by one sample every time the coefficients are recalculated.

In Chapter V, SGEP and SAP subsequently employed in DPCM systems

using either fixed or adaptive quantizers, ADPCM or ADPCM-AQJ, while the input speech signal is band-limited to 3.4 kHz and sampled at 10 kHz. The system with SGEP performs approximately 3.4 dB better than that with SAP for a transmission bit-rates of 16-40 Kb/s, and both systems provide substantial improvement compared to encoder having leaky integrator.

Then a DPCM coder having block adaptive quantization, AQF, with forward transmission of step size and a two coefficients adaptive predictor is examined. For a transmission rate of 40 Kb/s and a block size of 256 speech samples, the ADPCM-AQF system using the SGEP algorithm has SNRSEG gains of 3 and 9 dB compared to the same encoder, but with the SAP and the leaky integrator, respectively. The dynamic range of the ADPCM-AQF using SGEP for a SNRSEG of 35 dB is 30 dB. ADPCM-AQF-SGEP has weaker dependence on block size than ADPCM-AQF-SAP and has a higher SNRSEG over for bit error rates (BER) less than 0.1%.

Also, the informal listening tests of the coders described here, at transmission bit rates of 40 Kb/s and 30 Kb/s, reveal the subjective quality of the speech signal produced by DPCM-AQF-SGEP is superior compared to that obtained from DPCM-AQF employing SAP, ADPCM-AQF-SAP, or a leaky integrator, DPCM-AQF-FFOP.

Chapter VI is concerned with a novel DPCM-AQF system where the predictors, both fixed and SGEP, are switched according to a simple statistic of the speech signal to yield an improved performance. The speech signal is band-limited to 3.4 kHz and sampled at 8 kHz. The SNRSEG for the systems reveal the ADPCM-AQF-SGEP using 3rd-order CSP,

i.e., ADPCM-AQF-CSP(4)-SGEP, has the highest SNRSEG irrespective of the bit rate, but more important it has the best SNR during unvoiced/voiced transitions. Typical gains are reflected in a 2 to 3.8 dB improvement in SNRSEG for ADPCM-AQF when CSP associated with SGEP is used, ADPCM-AQF-CSP(4)-SGEP, to compare with the second order fixed predictor, DPCM-AQF-FSOP.

CHAPTER II

DIGITAL CODING OF SPEECH SIGNALS

2.1 TRANSMISSION BIT-RATES IN SPEECH CODING

A continuum of transmission bit-rates for digitally encoded speech, together with four zones of speech quality, is shown in Table 2.1. These qualities are designated, broadcast (high fidelity)- commentary, telephone toll, communications and synthetic.[13] Also shown are the two broad categories of speech digitizers: vocoders and waveform encoders.

For bit-rates $\geq$ 16 Kb/s, some waveform coders can produce toll-quality speech, namely a quality associated with analogue speech having a bandwidth of 200-3400 Hz, SNR$\geq$30 dB and a distortion <2%. For bit rates above 64 Kb/s, it is possible to have SNR and harmonic distortion characteristics of toll quality speech for wider bandwidth, typically 0-7 kHz, of input signal. When the bit-rate is below 16 Kb/s, the coders provide communication quality speech. The recovered speech is intelligible although there is a noticeable reduction in quality compared to toll quality speech. Coders, specifically vocoders, at 4.8 Kb/s and below, produce an output speech which has lost its naturalness. The speech has a tendency to sound machine-like, and speaker identification may be difficult. This quality of speech is referred to as synthetic quality.

The complexity of speech encoding systems tends to be a function of the transmitted bit-rate. Consequently, waveform coders which usually operate at higher bit-rates, tend to be less complex and less expensive. Vocoders, on the other hand tend to be more complex and costly.

TYPE OF CODING	Kb/s	QUALITY	COMPLEXITY
WAVEFORM CODING	200	BROADCAST COMMENTARY	LEAST
	64 32 24	TELEPHONE TOLL	MODERATE
	16 9.6 8	COMMUNICATION	GREAT
VOCODERS	7.2 4.8 2.4 1.0	SYNTHETIC	GREAT

TABLE 2.1: Spectrum of Bit-Rates and Qualities for Speech Digitizers

2.2 VOCODERS

Digital coding of speech signals using parametric representation techniques is referred to as vocoders. In these techniques, parameters based on a model of the vocal mechanism and a simplest representation of the auditory system are extracted from the speech signal, digitized, and transmitted. Vocoders, except phase vocoders and voice-excited vocoders, require excitation parameters; namely voiced/unvoiced decision, and pitch information (i.e., fundamental glottal frequency).[10] Excitation information, together with the vocal tract model parameters are the essential ingredients used in synthesising the speech signal. Before quantifying the parameters, we pause briefly to consider the physiology of the vocal mechanism.

A schematic diagram of the vocal tract is shown in Figure 2.1,[2,14] where the lungs, trachea, larynx, throat, nose and mouth contribute to the production of speech. The vocal tract may be considered as an approximation, to be an acoustical tube, between 15-17 cm in length, having a uniform cross-sectional area that extends from the lips to the vocal chords. However, the motion of the lips, jaw, tongue and velum, known as articulators, affect the structure of vocal tract. The nasal tract and vocal tract couplings are controlled by the size of the opening at the velum. The voiced sounds are associated with the vibration of the vocal chords, and unvoiced sounds result from turbulent air flow through a constriction of the vocal tract. Both of these sound sources have significant power over a wide range of audio frequencies, but the spectrum of the radiated sound results from

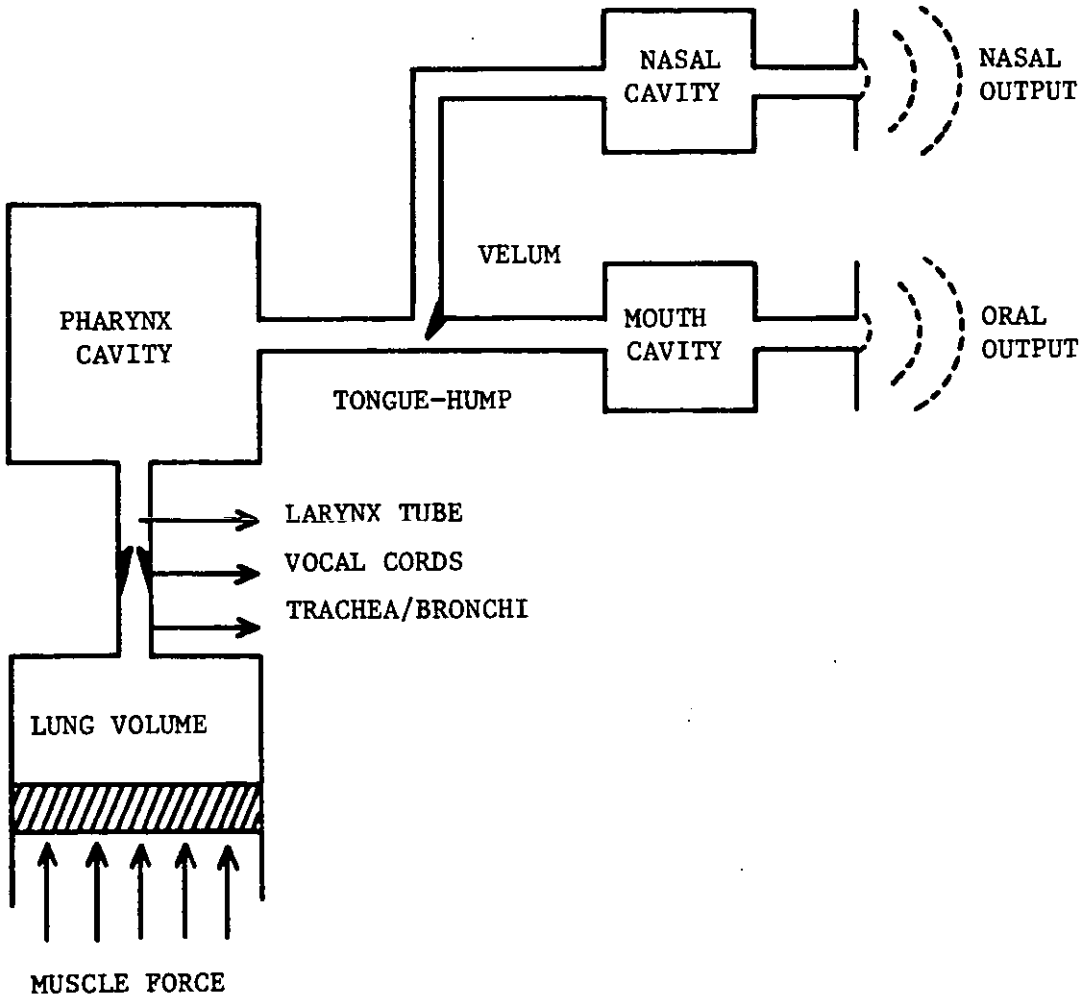


FIGURE 2.1: Schematic Diagram of The Vocal Tract[2]

spectral shaping of these sources by the acoustic resonant system of the vocal tract. The resulting sound, analysed over a time window of a few tens of milliseconds, has either a periodic or random structure, corresponding to voiced or unvoiced speech, respectively.

An important concept used in the parametric representation of speech is the formant which corresponds to frequency values of major spectral resonances associated with the peaks in the power spectral density function. Telephonic speech, band-limited to 3.4 kHz has typically three formant frequencies, f_1, f_2 and f_3. However, the location of such frequencies depends on the variation of cross-sectional area of the vocal tract resulting from the movement of the tongue and the position of jaw along the vocal tract. For example, vowel /a/, as in *"father"*, is produced by opening the lips, moving the tongue etc., resulting in an increase of the first formant frequency, f_1. As another example, the vowel /e/, as in *"eve"*, is formed by moving the tongue forward and this causes a reduction in f_1. For the same vowel utterance, the formants change from speaker to speaker and this has been studied in detail by Peterson and Barney.[15]

Approximating the voice production mechanism by a linear system,[16] as shown in Figure 2.2, enables the sound to be represented by the time convolution of the excitation function e(t), and the impulse response, u (t), of the vocal tract filter, viz:

$$x(t) = u(t) * e(t) \tag{2.1}$$

where * implies convolution.

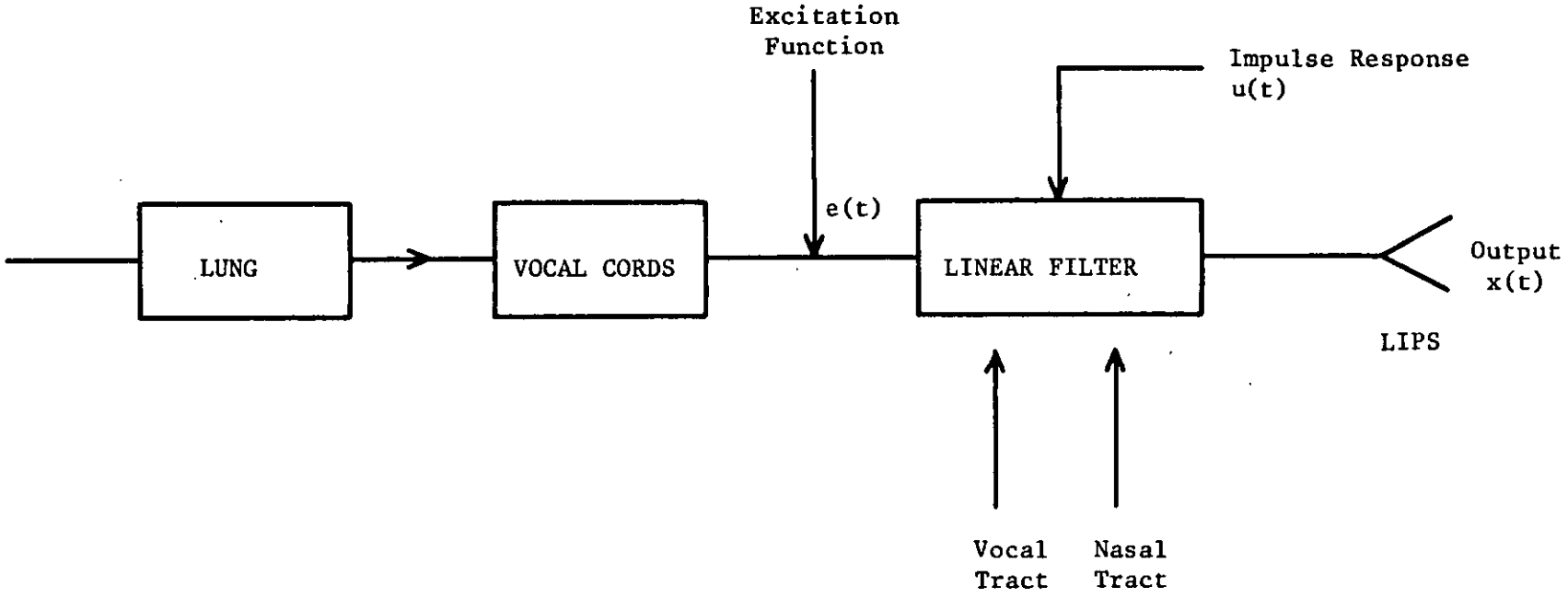


FIGURE 2.2: Voice Production Mechanism by a Linear System

In the frequency domain, this convolution is equivalent to a multiplication of the Fourier transforms of u (t) and e(t), such that the spectrum of the speech signal can be represented as

$$X(\omega) = U\ (\omega).E(\omega) \tag{2.2}$$

and whose magnitude is

$$|X(\omega)| = |U\ (\omega)|.|E(\omega)| \tag{2.3}$$

Therefore, in the frequency domain, the term $E(\omega)$ manifests as the fine structure of $X(\omega)$ while $U\ (\omega)$ corresponds to the envelope of $X(\omega)$. When voiced speech occurs $E(\omega)$ is a fine line structure, and $U\ (\omega)$ has a succession of peaks (typically 3 or 4 for telephone speech) whose frequencies are called formants. Unvoiced speech has $E(\omega)$ that is noise-like, as the vocal cords are not excited and e(t) is the result of air turbulence in the vocal tract. The spectral envelope $U\ (\omega)$ has usually one or two formants which often reside above 3.5 kHz, e.g., /s/ has a single formant at about 5 kHz.

Synthesizers in vocoders consequently employ an excitation source that is either a periodic pulse generator when voiced speech is present, or a random noise generator when unvoiced sounds occur.

The basic elements of a vocoder are shown in Figure 2.3. The box labelled *"vocal tract parameters"*, essentially provides a measure in parameter form of the spectral envelope of the short-time speech signal, information known to be required by the hearing mechanism. The vocoder analyser determines the vocal tract parameters, and the type of

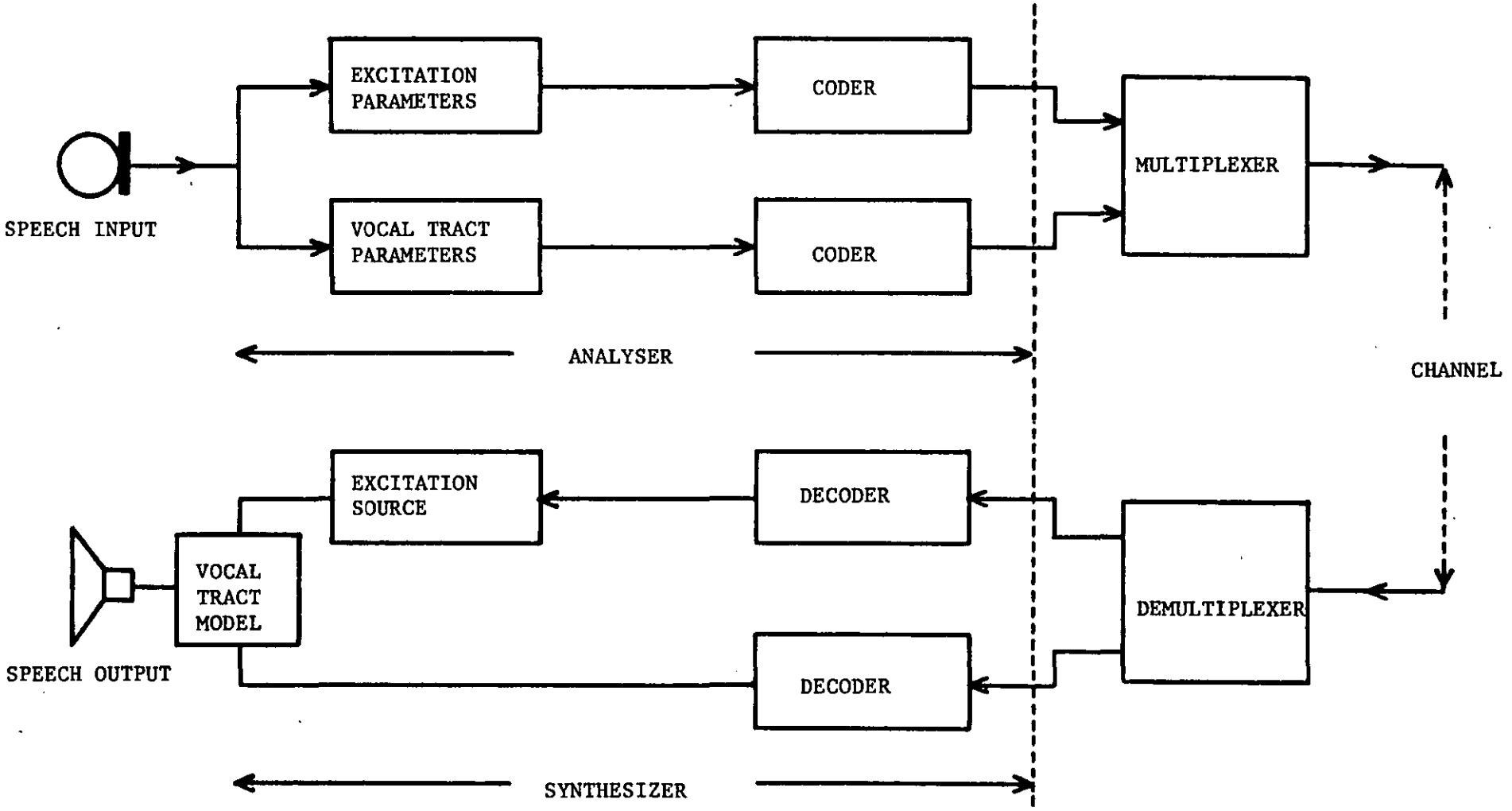


FIGURE 2.3: Generalised Block Diagram of Vocoders

excitation. In doing so significant reductions in signal bandwidth is achieved. For example, speech band-limited to 3 kHz can be represented in parameter form by signal whose bandwidth is of the order of 300 Hz. Usually these parameters generated by the vocoder analyser are digitized and transmitted. The synthesizer at the receiver decodes the digital signal and recreates the speech using either Equations (2.1) or (2.2).

The vocal tract parameters take a variety of forms, e.g. the amplitude spectrum of the signal at various frequencies (channel vocoder), prediction coefficients that define the spectral envelope, called Linear Predictive Coders, LPC, the frequencies of major resonances (formant vocoder). Details of speech production mechanism and vocoder designs are described by Flanagan, Moye,[17] Holmes[18] and Rabiner at al.[11,19] Widely used vocoders are the Channel Vocoder and the LPC vocoder, while the more complex formant vocoder with its greater complexity may be preferred in the future when technology makes its realisation at a competitive cost.

2.2.1 Channel Vocoders

In channel vocoders,[18,20] see Figure 2.4, in order to preserve the shape of the short-term amplitude spectrum at specific frequencies, the signal spectrum is divided into frequency bands, called channels, by using a bank of contiguous variable-gain bandpass-filters. The total number of channels are typically of the order of 10-20 with bandwidths

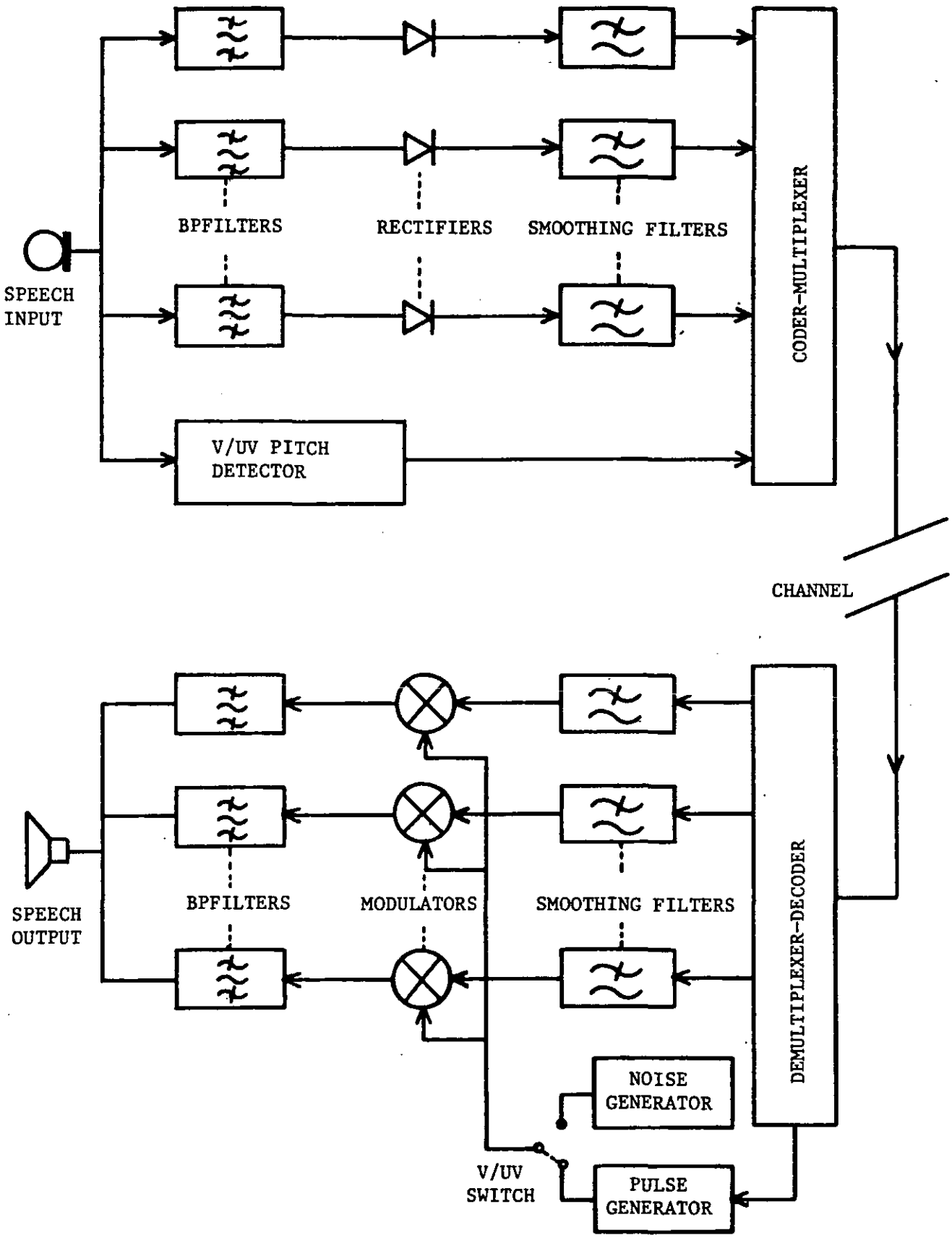


FIGURE 2.4: Block Diagram of Channel Vocoder

of 300 Hz to 150 Hz, respectively. By the use of rectifiers and low pass filters, the output of each channel produces the discrete power spectrum of the speech signal for the corresponding frequency band. The voiced/unvoiced decisions, V/UV, the pitch (quasi-periodic frequency of the vocal cords) if voiced sound is present, are measured in the analyser. Their values are multiplexed with the vocal tract parameters, and transmitted. The function of the synthesizer at the receiver is to produce a perceptually accepted reproduction of the original speech signal without attempted waveform replication. Voiced sounds are synthesized by using periodic pulse generator as the excitation source. Whereas for unvoiced sounds, a random noise generator serves as a substitute for the excitation source. Consequently in accordance with the vocal tract information produced by the analyser, either voiced or unvoiced sounds can be generated at the synthesizer. The channel vocoder achieves significant bandwidth reduction since the spectral envelope of the short-term spectrum has a lower bandwidth than that of the original speech signal. Typical transmission bit rates are between 2.4-4.8 Kb/s. However at 2.4 Kb/s, there is a degradation in naturalness despite the use of complex equipment. This is because:

a) Errors in voiced/unvoiced decisions,

b) the pulse generator being a poor replica of the vocal cord excitation,

c) the choice of bandwidth and filter spacings,

d) but most important, degradation derives from the inaccuracies of the basic model of the vocal tract and the excitation processes. Only by deriving more precise models of the speech will the synthetic quality of vocoder speech be removed.[21]

2.2.2 Formant Vocoders

Here the vocal tract parameters are the major formants f_1, f_2 and f_3 and their amplitude FA_1, FA_2 and FA_3. The analyser attempts to locate the formants by dividing the speech spectrum into frequency bands.[13,16,17,22] For each spectral band, the average frequency, f, and the rectified, low-pass filtered amplitude FA are measured. Signals proportional to f and FA are then transmitted into a digital form. Figure 2.5 shows a block diagram of formant vocoder. The synthesizer of such vocoder is related to human speech production, because its resonators specifically correspond to the major formants of the input speech signal. Hence, the synthesizing can be achieved by using the known frequency range and the specification of each formant. All these factors result in significant reduction in bit-rate to as low as 1.2 Kb/s. However, at this low rate, the reproduction of good quality speech depends on the accuracy in locating the formants at the analyser. In modern formant vocoders, digital computer techniques are used for calculating formant frequencies and peak values are determined using discrete Fourier Transform, DFT, methods.

2.2.3 Pattern-Matching Vocoder

This vocoder[10,22,23] achieves further reductions in bit-rates, i.e., typically 0.6-0.8 Kb/s. It operates by comparing the short-time speech spectrum with a set of stored spectral patterns where each pattern is specified by a binary code. In this way, one of the stored

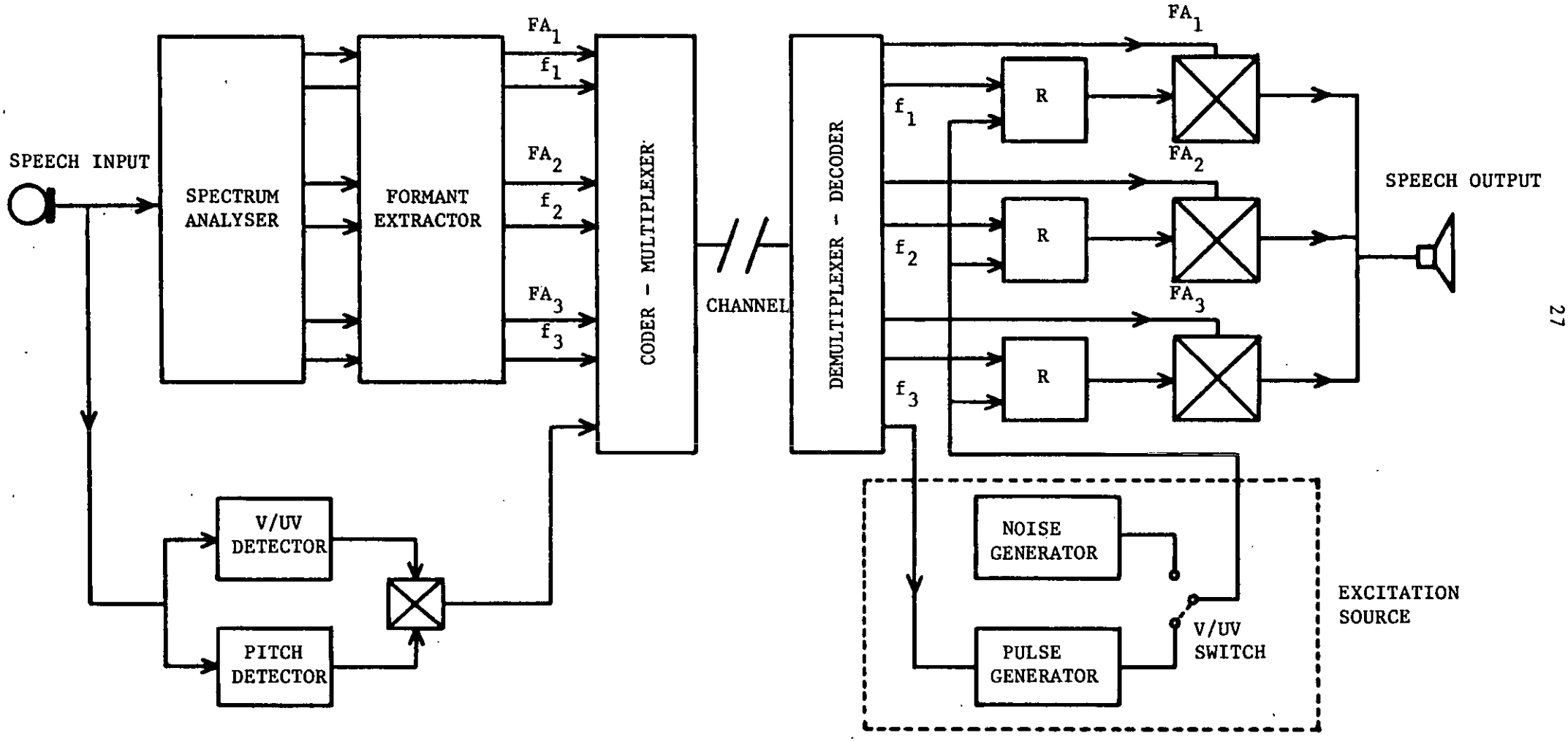


FIGURE 2.5: Block Diagram of Formant Vocoder[16]

patterns which corresponds closest to the speech spectral is identified, and its code is transmitted, together with the pitch and voiced/unvoiced information. The received code is used to synthesize the speech signal.

2.2.4 Ceptrum-Homomorphic Vocoder

The ceptrum of the speech signal is the basis of another type of vocoder, known as the ceptrum-homomorphic vocoder. Before describing such a vocoder, we pause briefly to define the concept of ceptrum.[11,19,22,24] The term, ceptrum, results from the logarithm of Equation (2.3),

$$\log|X(\omega)| = \log|U(\omega)| + \log|E(\omega)| \tag{2.4}$$

Equation (2.4) reveals that the excitation function, $E(\omega)$, and vocal tract function, $U(\omega)$, become additive, hence $e(t)$ and $u(t)$ can be separated by a filtering process. The inverse DFT, IDFT, of Equation (2.4) is,

$$\text{IDFT}\{\log|X(\omega)|\} = \text{IDFT}\{\log|U(\omega)|\} + \text{IDFT}\{\log|E(\omega)|\} \tag{2.5}$$

which gives the ceptrum, $c(t)$. Figures 2.6 and 2.7 represent the time waveform, amplitude spectrum, log-spectrum and ceptrum for voiced and unvoiced sound respectively. For voiced speech, the region around the origin, see Figure 2.6(d), is due to the impulse response of vocal tract, while the remaining portion of the waveform is due to the excitation source. Now, for a periodic excitation, the ceptrum of the excitation is also a train of impulses with the same spacing as the impulse train. With voiced speech, the quasi-periodic nature

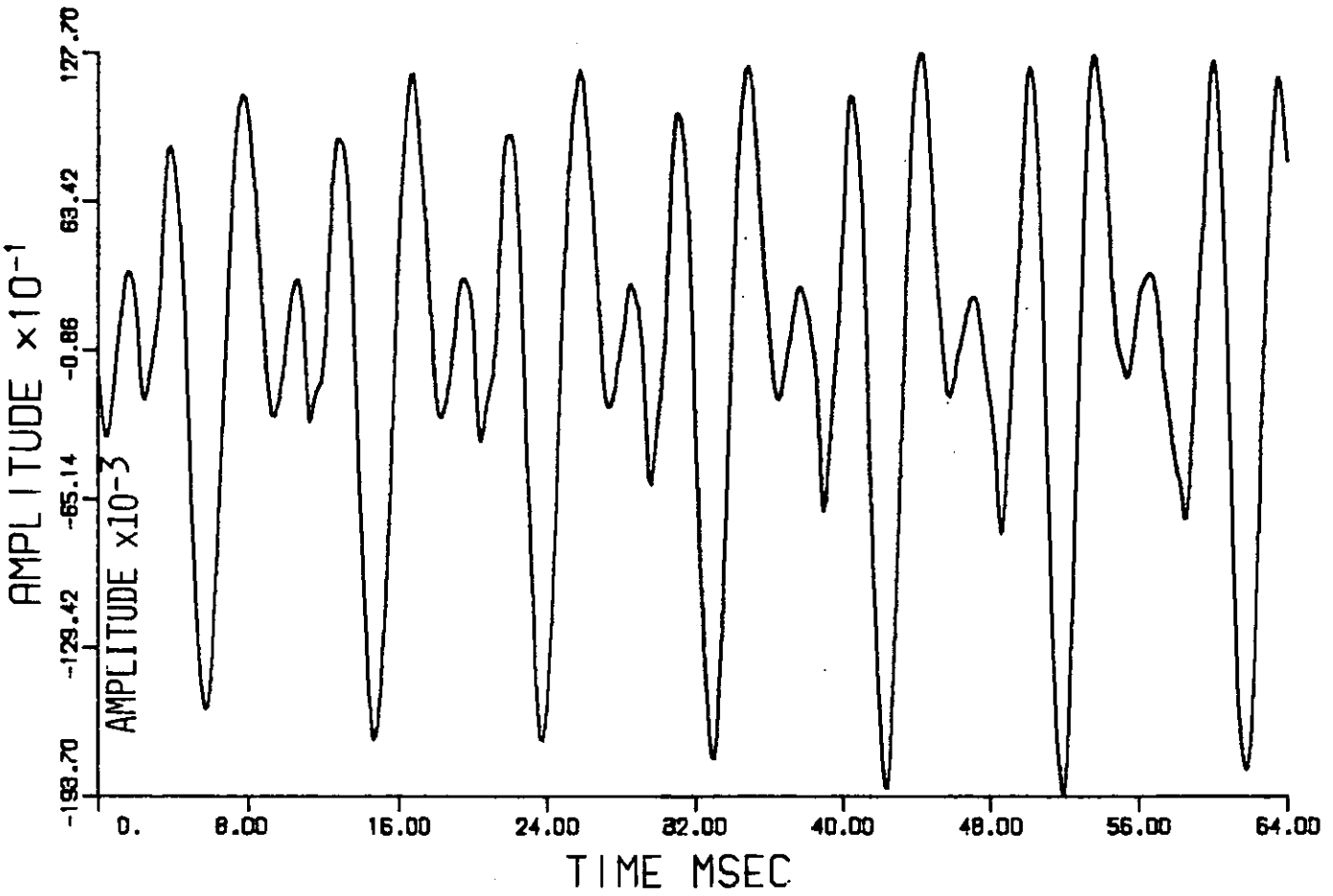
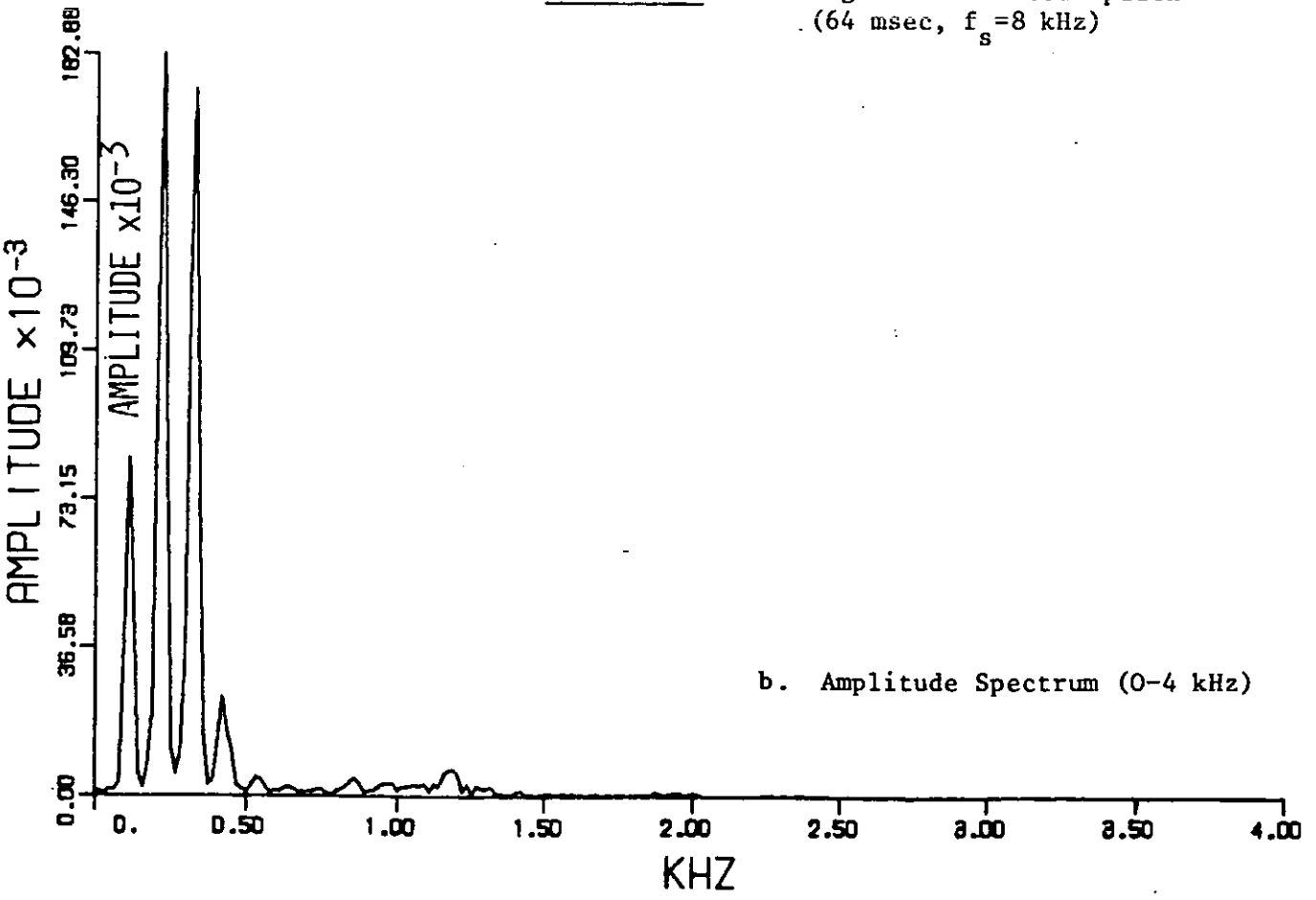
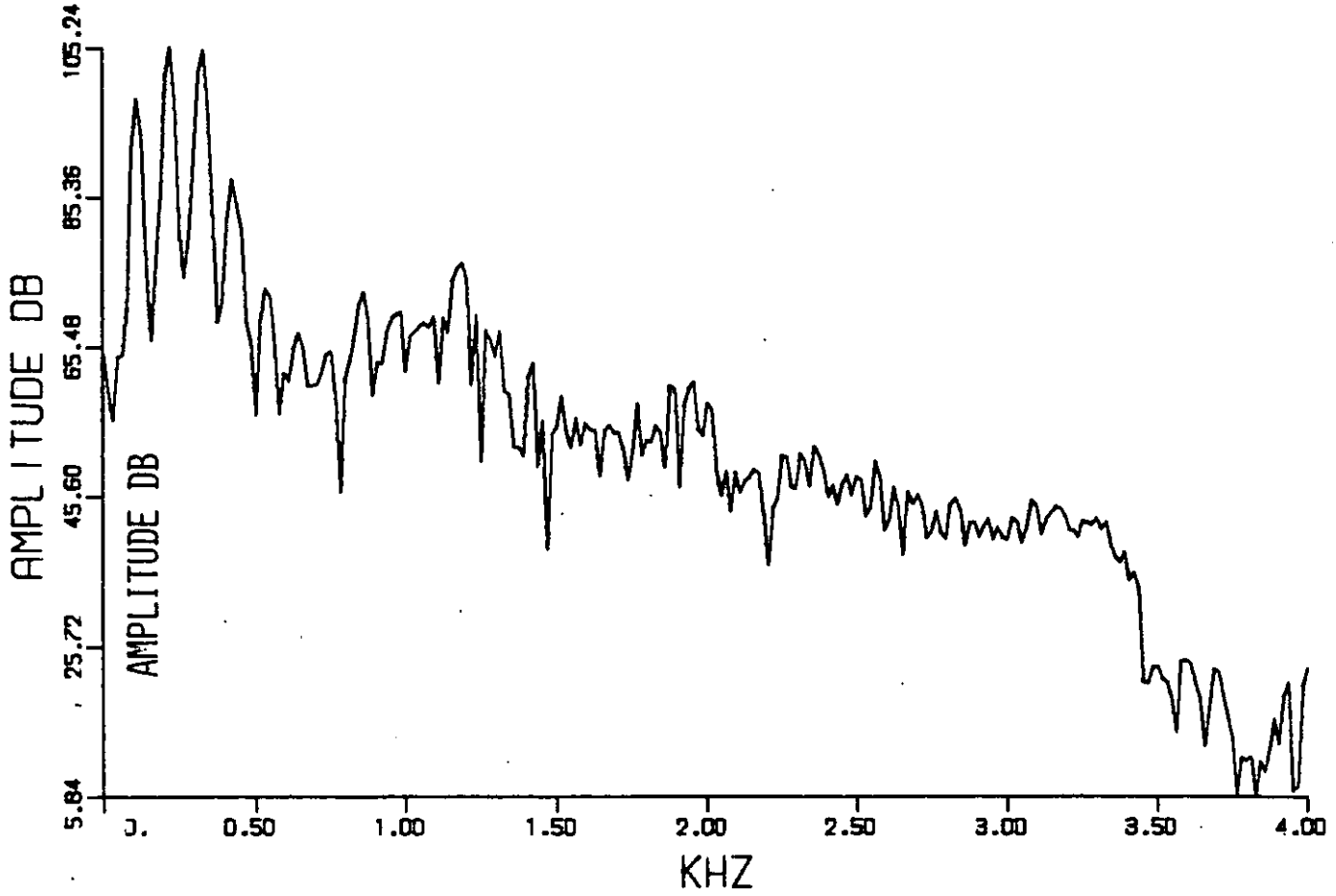


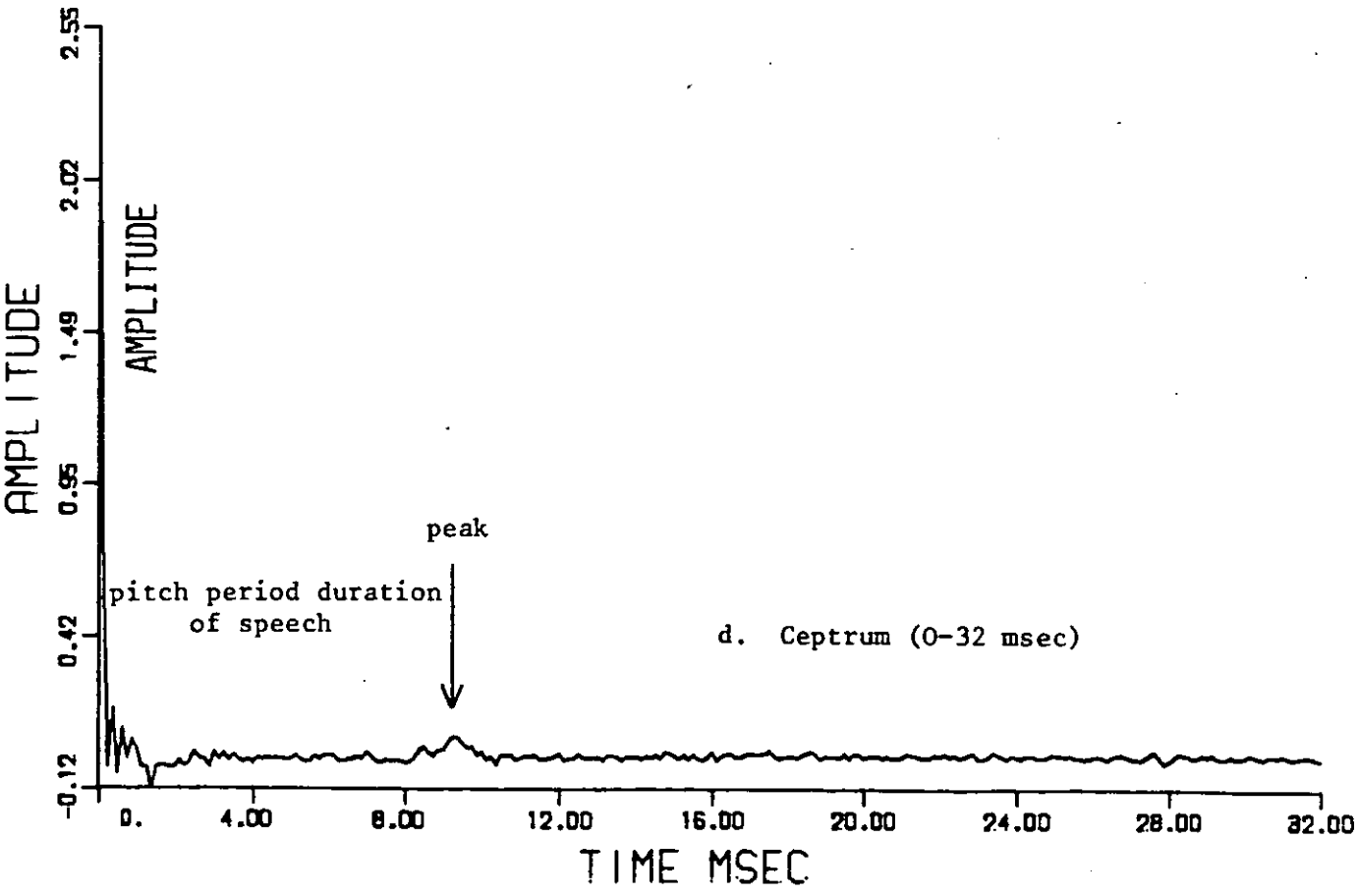
FIGURE 2.6 a. Segment of Voiced Speech (64 msec, f_s=8 kHz)



b. Amplitude Spectrum (0-4 kHz)



c. Log (amplitude spectrum)



d. Ceptrum (0-32 msec)

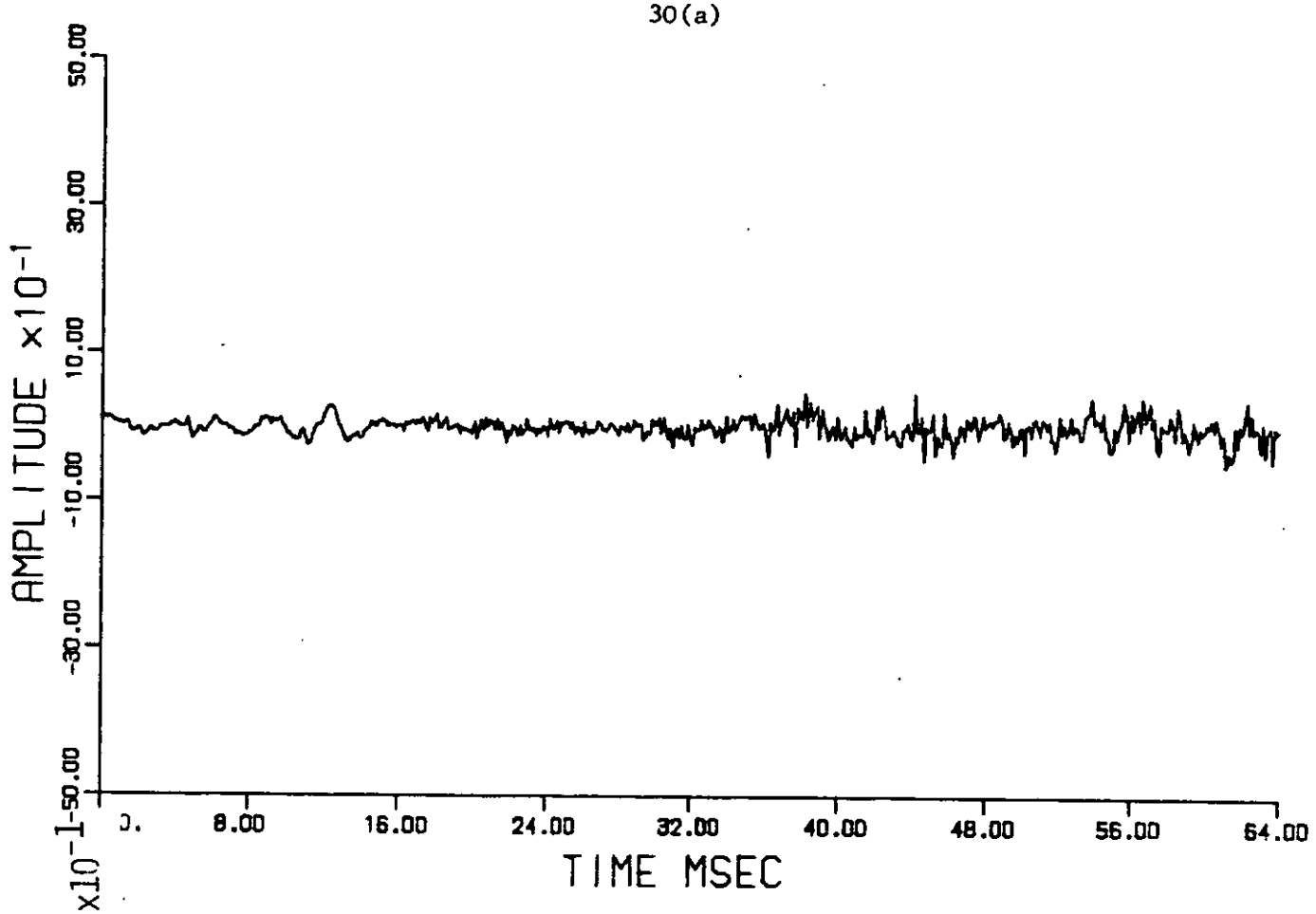
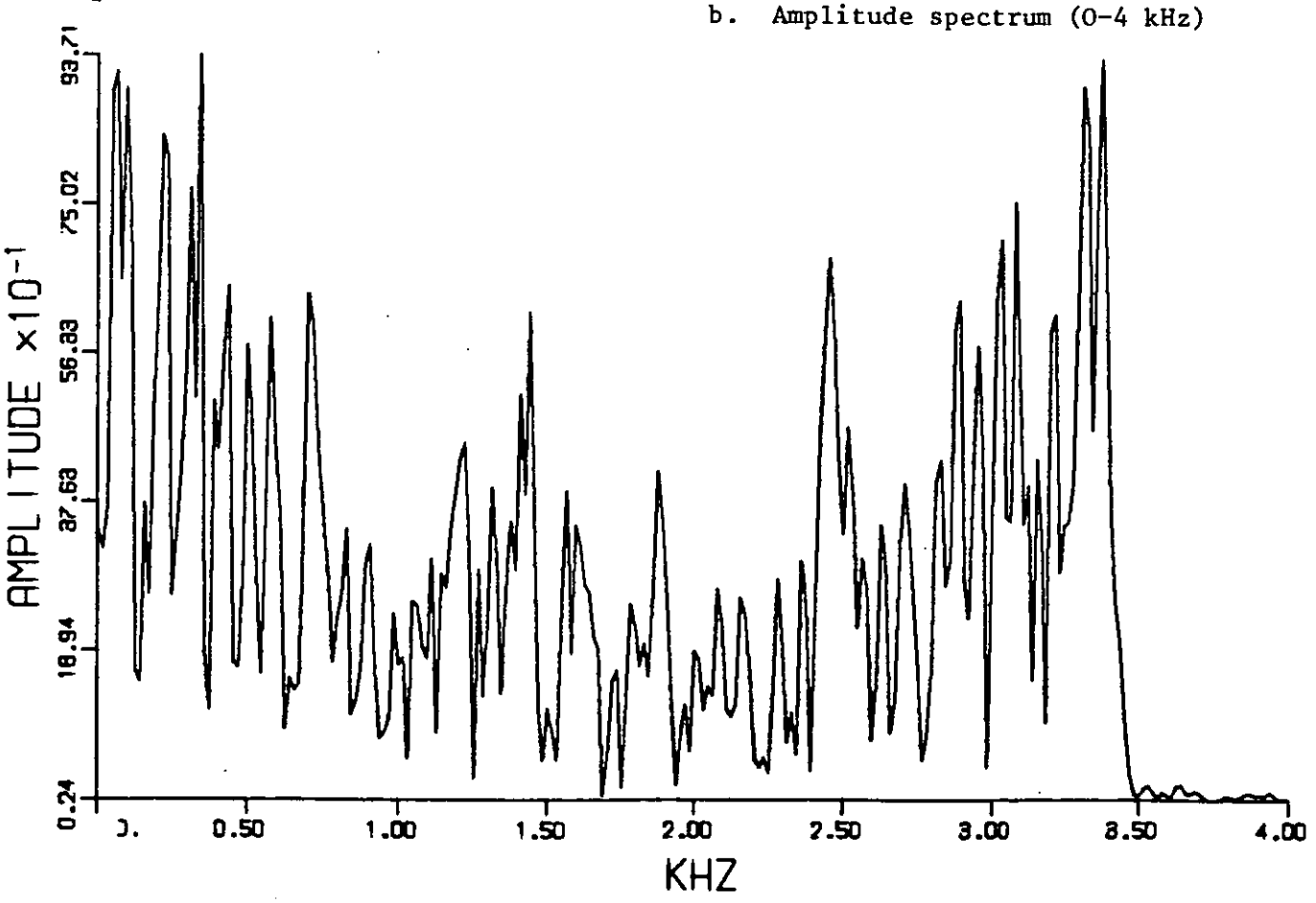
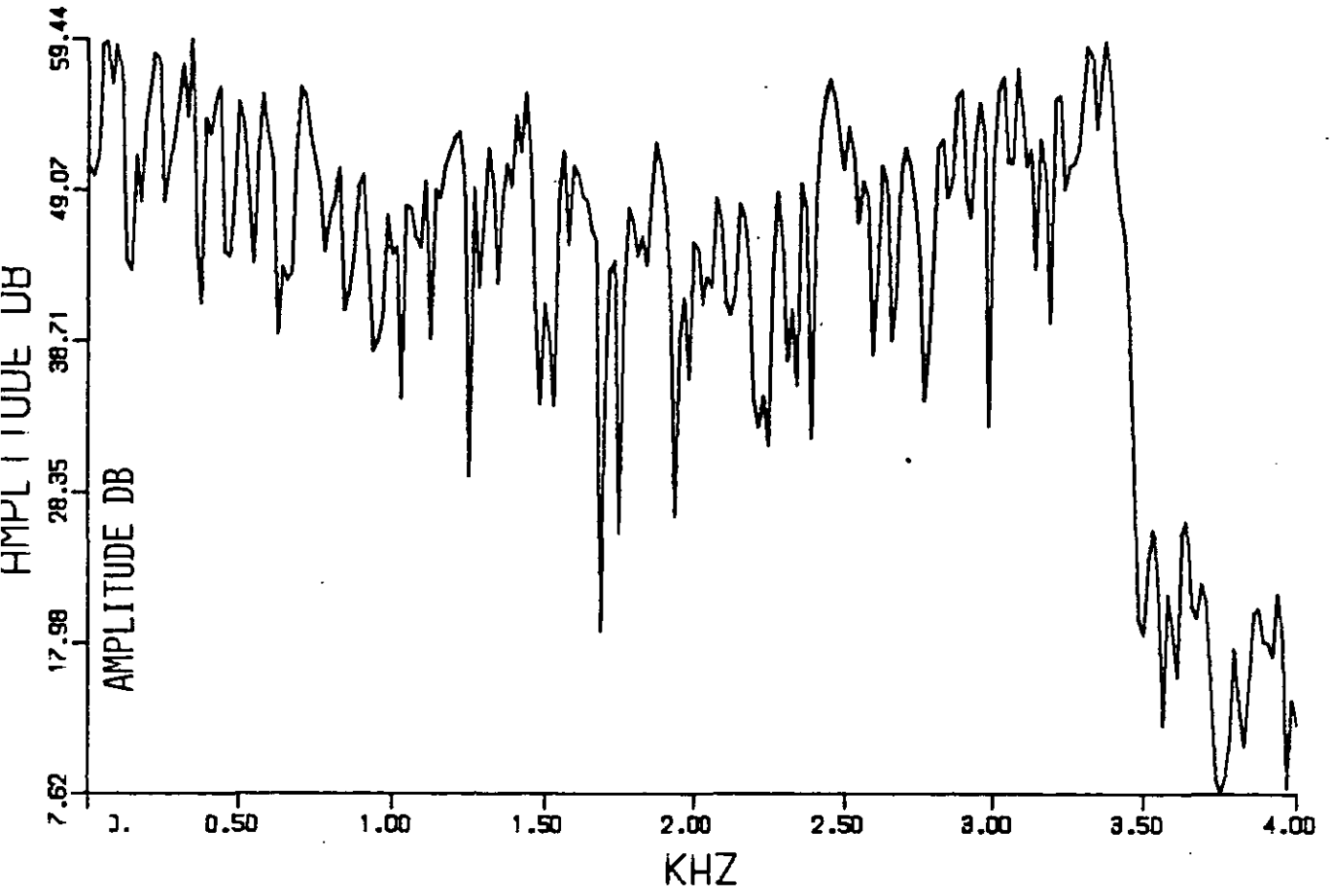


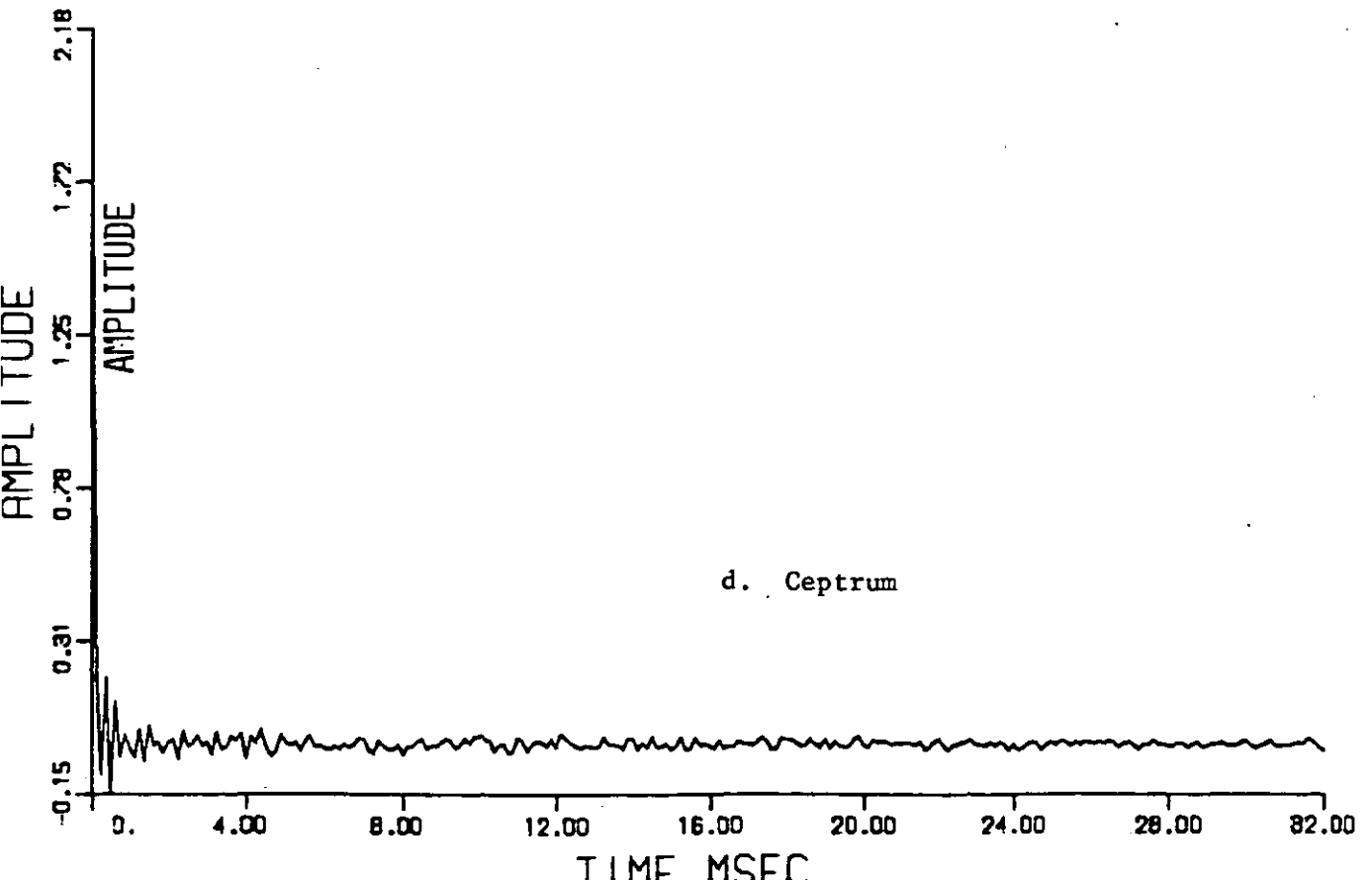
FIGURE 2.7 a. Segment of Unvoiced Speech (64 msec)

b. Amplitude spectrum (0-4 kHz)





c. Log (amplitude spectrum)



d. Ceptrum

of the excitation which consists of puffs of air that are asymmetrical in shape, yield a ceptrum that consists of a large spike, as shown in Figure 2.6(d). However, for unvoiced sound, the selection of a strong and isolated peak in the ceptrum is not feasible, see Figure 2.7(d), due to the random nature of the excitation. Thus, ceptrum can be used to make the decision as to whether speech is voiced or unvoiced.

The property of ceptral analysis is exploited in homomorphic vocoders, as shown in Figure 2.8. The speech is windowed using a Hamming function, typically 40 ms. duration, and then is Fourier-transformed. A rectangular window is avoided because of the *"spectral leakage"* that it produces.[25] On the other hand, ceptrum window shown in Figure 2.8 is used to remove the excitation information. From c(t), the pitch is deduced, and this is encoded along with the vocal tract information and both are transmitted. In the synthesizer, the DFT of c(t) produces $\log|U(\omega)|$, exponentiation ensues and the inverse transformation yields u(t). The excitation generator produces either pitch or random noise in accordance with V/UV information respectively. Finally, the discrete convolution process convolves u(t) and e(t) to produce the synthesized speech.

Such vocoders require only the computation of the ceptrum which is often available in the pitch identification process. At 7.8 Kb/s, homomorphic vocoder yields good synthetic quality speech, and its implementation has recently been eased by the invention of charged-coupled devices, CCD.

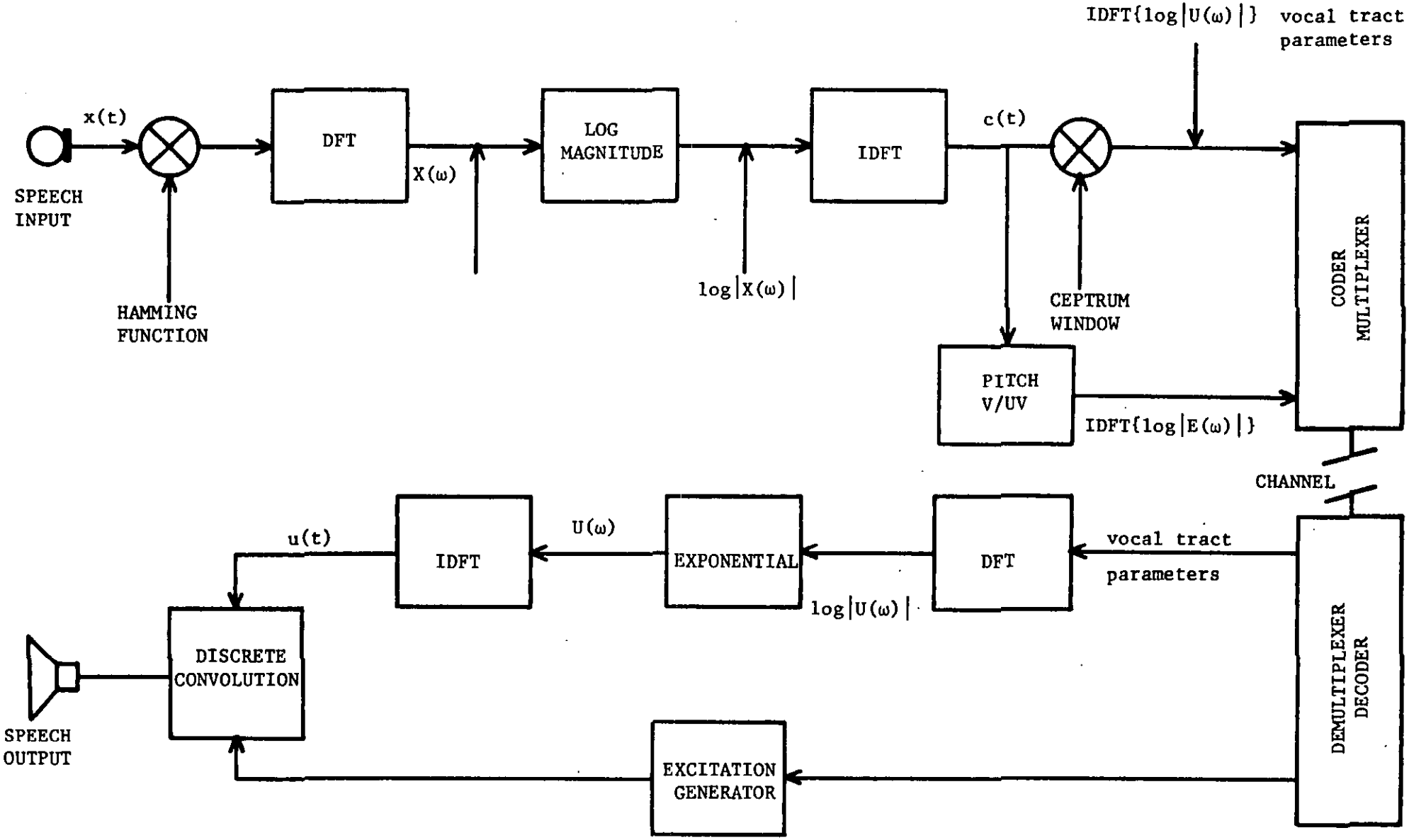


FIGURE 2.8: Block Diagram of Ceptrum Vocoders

2.2.5 Linear Predictive Coding (LPC) Vocoders

A different method of representing the spectrum of a speech sound is by means of the LPC vocoder in which the spectral approximation is given by the response of a sampled-data filter. In LPC, the modelling of the speech waveform is carried out in the time rather than the frequency domain,[11,13,26] thereby reducing the difficulties of locating formants which are inherent in frequency domain techniques. In recent years, much literature has appeared in favour of LPC vocoders (particularly in the U.S.A., more than the U.K.) due to its speed of computation and simple implementation.[27] Most of the research in LPC analysis has been focussed on all-pole models. However, the presence of unvoiced and nasal sounds suits a zero-pole model sometimes known as autoregressive moving average model, (ARMA), whose mathematical treatment is rather complicated and will not be mentioned here.[28]

The transfer function, H(z), of the all-pole model, sometimes referred to as autoregressive model, is given by,

$$H(z) = \frac{G_e}{1 - \sum_{k=1}^{N} a_k z^{-k}} \qquad (2.6)$$

where G_e is the amplitude of the input excitation. The coefficients, a_k's, specify the all-pole approximation of the short-time speech spectrum. Figure 2.9 shows the speech production process. The output of the linear filter at i^{th} sampling instant is,

$$\hat{x}_i = \sum_{k=1}^{N} a_k x_{i-k} + G_e \delta e_i \qquad (2.7)$$

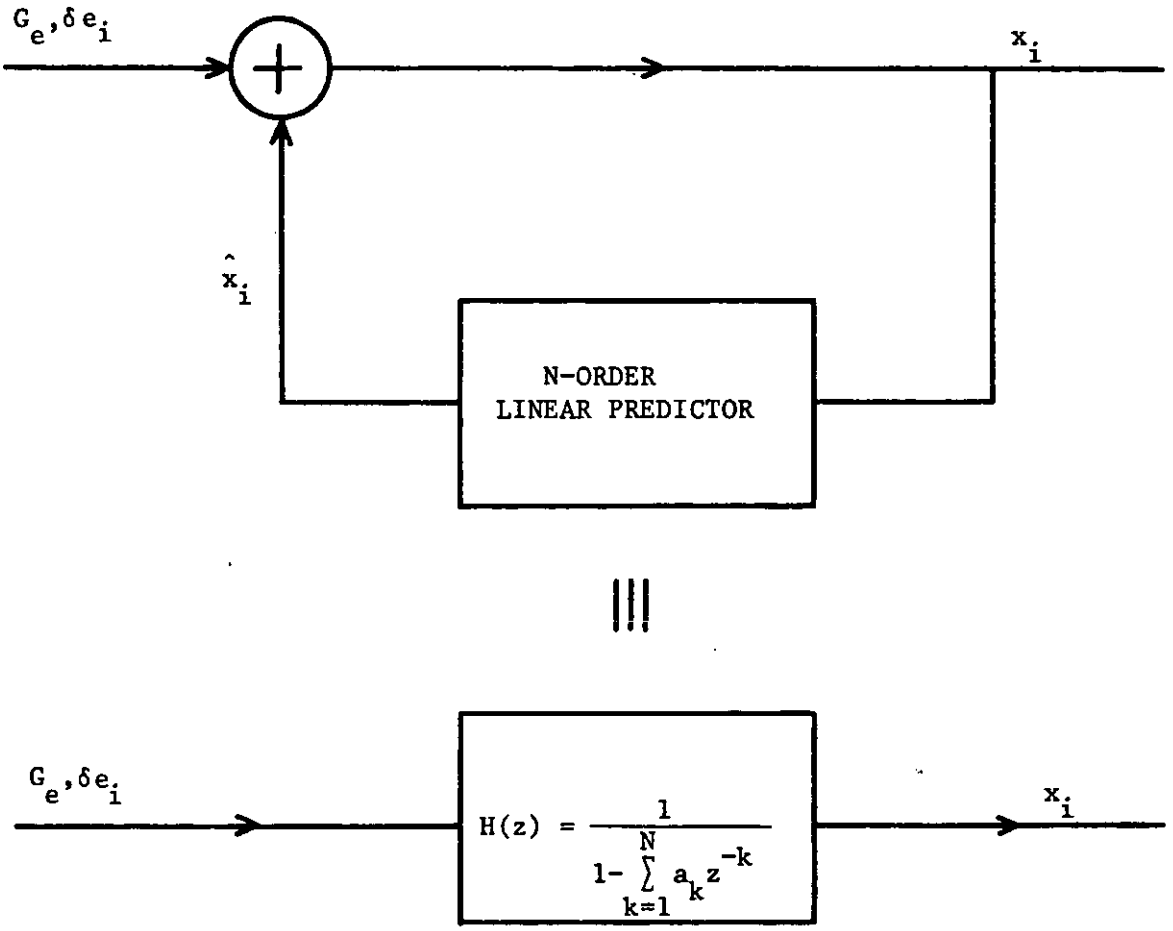


FIGURE 2.9: Block Diagram of Speech Production

where δe_i corresponds to the excitation pulses and G_e is due to the amplitude variations of the excitation source. The output speech sample $\hat{x}_i$ is $\sum_{k=1}^{N} a_k x_{i-k}$, except at the beginning of a pitch period. Thus generation of speech by this technique requires a knowledge of the pitch, V/UV information, a_k coefficients and the gain of filter. All these parameters are transmitted in digital form, as shown in Figure 2.10, to enable the synthesize of the speech signal.[29-30] These vocoders give good performance for bit-rates of the order of 2.5 to 4 Kb/s.

The sampled-data filter coefficients, a_k's, are calculated so that the error between the original sample and the predicted value becomes minimum, i.e.,

$$\langle e_i^2 \rangle \Big|_{\hat{A}=\hat{A}_{opt}} = \langle (x_i - \sum_{k=1}^{N} a_k x_{i-k})^2 \rangle \qquad (2.8)$$

where $\langle(.)\rangle$ means time average of (.). To achieve this, mean-square error criterion, the autocorrelation and autocovariance techniques are often used, updating the coefficients at a rate commensurate with significant changes in the vocal tract, i.e., the order of 5-30 ms. The number of prediction coefficients, N may be selected according to the number of formants in the speech signal, usually N is equal to twice the number of formants, i.e., typically 6 to 8.[31] The complex-roots of Equation (2.6) indicate the location of formants and their bandwidths. In correlation/covariance methods, Equation (2.8) is minimized so that,[26]

$$\hat{A}_{opt} = \Phi^{-1}\Psi \qquad (2.9)$$

where Φ is the autocorrelation/covariance matrix, Ψ is the autocorrelation/

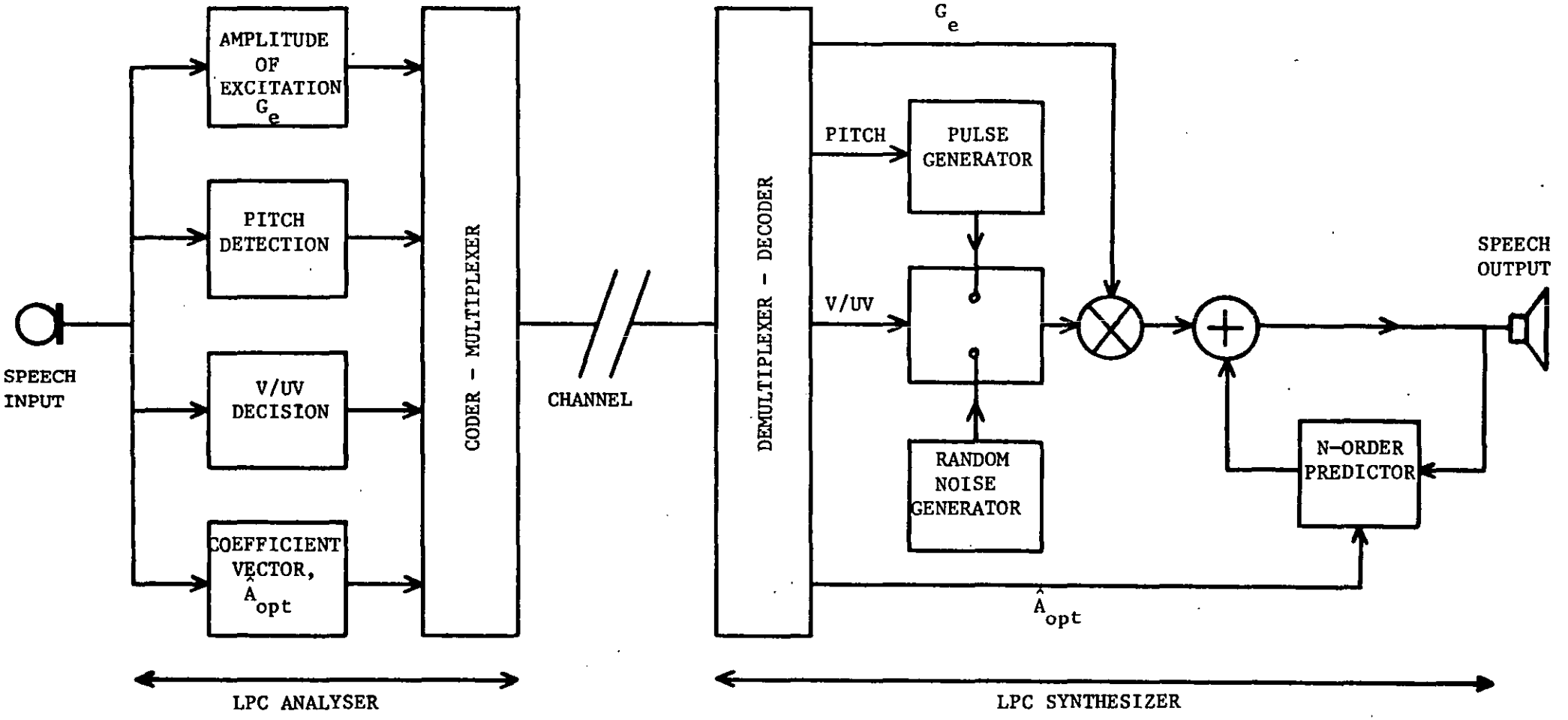


FIGURE 2.10: Block Diagram of LPC Vocoder

covariance vector and $\hat{A}_{opt}$ is the optimum vector set of prediction coefficients.

A different approach of estimating the vocal tract parameters involves the computation of PARCOR (PARtial CORrelation) coefficients which are found to be less sensitive to transmission errors.[24,27-28,32]

Further, in Chapter IV, it will be shown that the predictor coefficients can be modified sequentially, hence modelling of vocal tract is achieved in an adaptive manner.

2.2.6 Relative Merits of Vocoders

Table 2.2 shows the performance of various vocoders described here, in terms of speech quality, transmission bit-rate and complexity.[10]

VOCODER	Kb/s	QUALITY	COMPLEXITY
CHANNEL	2.4	FAIR	HIGH
FORMANT	1.2	GOOD	VERY HIGH
PATTERN MAT.VOCODER	0.8	POOR	VERY HIGH
CEPTRUM	7.8	GOOD	HIGH
LPC	2.4	FAIR	HIGH

TABLE 2.2: Relative Merits of Vocoders

2.3 WAVEFORM ENCODING OF SPEECH SIGNALS

The methods in this section are confined to the reproduction of an actual pressure waveform using discrete-time and amplitude representation of the speech signal. These techniques avoid the extraction of the vocal tract and excitation parameters. Waveform coders are generally simpler in implementation and therefore less expensive than vocoders.[21] They also have a more natural sounding quality. An increasing use of digital systems for transmission purposes requires the speech signal to be sampled (quantized in time), and quantized in amplitude to a set of finite values. Due to quantization process, the received signal differs from the original speech signal, and the difference is called distortion-quantization noise.[33]

A schematic block diagram of digital waveform coder is shown in Figure 2.11. A speech signal, x(t), is sampled at a rate exceeding twice its highest frequency component (Nyquist Rate), to produce the sequence of samples $\{X_i\}$, i=1,2,...,NS, where NS is the total number of samples. These samples are applied to the waveform encoder which generates a set of quantized amplitudes $\{X_e\}$ which are encoded for transmission. Binary coding is usually employed for this purpose, although higher order coding could be used, as $\{X_e\}$ is unsuitable for transmission through most practical channels. The binary output sequence $\{L_i\}$ is relatively robust to channel interference and the bits are relatively easy to regenerate at the receiver. After regeneration, the binary code words are decoded into the sample quantized amplitudes, $\{X_d\}$. In an error free channel the binary code

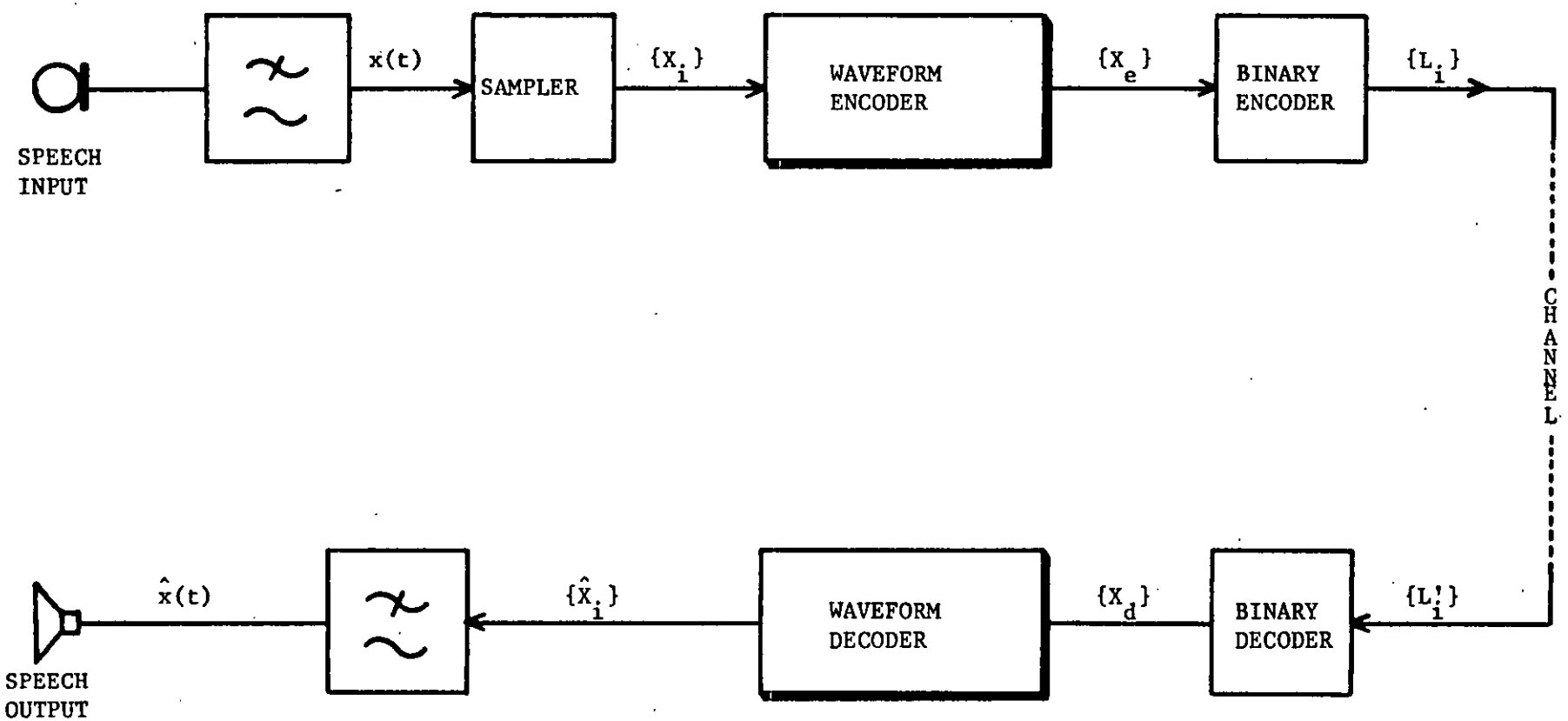


FIGURE 2.11: Schematic Diagram of Waveform Codec

sequences, $\{L_i\}$, $\{L_i'\}$ and the quantized sequences $\{X_e\}$ and $\{X_d\}$ are identical. The sequence, $\{\hat{X}_i\}$ at the output of the waveform decoder is low-pass filtered to yield the speech signal $\hat{x}(t)$. It is evident that even if $\{L_i\}=\{L_i'\}$, $x(t)$ differs from $\hat{x}(t)$ due to the quantization process at the transmitter. For a given bit-rate, b, the quantizer has 2^b discrete levels. The relationship between the sampling rate f_s and b yields the information capacity, CI,

$$CI = b\ f_s \quad \text{bits/sec.} \tag{2.10}$$

Equation (2.10) reveals that the main goal in the design of waveform coders is to reconstruct the analogue signal as accurately as possible with a minimum number of bits/sample, b, for a given f_s. The higher b, the smaller the noise or vice-versa. Thus, there is a compromise between the quality of the recovered speech and the bit-rate.[11] We now, consider the main types of waveform coders.

2.3.1 Pulse Code Modulation, PCM

The simplest form of waveform coding, known as PCM was suggested by Reeves[34] in 1938. It was the first method used for digital transmission of speech and is used extensively throughout the world in digital telephony.

The operations involved in a PCM coder is described in detail by Cattermole[5] and is as follows: The band-limited speech data, $x(t)$, sampled at the rate of $\geq 2f_c$, where f_c is the highest frequency in the

speech signal, is quantizer into the nearest 2^b levels, where b is the number of bits in each word assigned to the quantized samples. The binary words are then transmitted. At the receiver, bit regeneration followed by binary decoding occurs, and the resulting amplitude levels are low-pass filtered to give the recovered speech signal, $\hat{x}(t)$. High fidelity reproduction of speech is achieved by using a large number of quantization levels, as the quantization accuracy improves when the levels are closer together. Observe that the only source of noise in PCM is that generated in the quantizer. The choice of quantization step size, componding techniques,[5,35] namely signal compressing followed by expansion at the receiver, and the adaptive step size quantizers,[36,37] are all various means of reducing the PCM bit-rate for a given recovered speech quality.

2.3.2 Quantizers

The purpose of the quantizer is to replace each speech sample with one which is a close approximation of the original sample. The quantized sample is confined to one set of finite values. A b-bit binary word is associated with an output quantization level, so that each sample is transformed into a unique binary word. Output levels of the quantizers are sometimes called *"quantum levels"*, or *"quantization levels"*.[39]

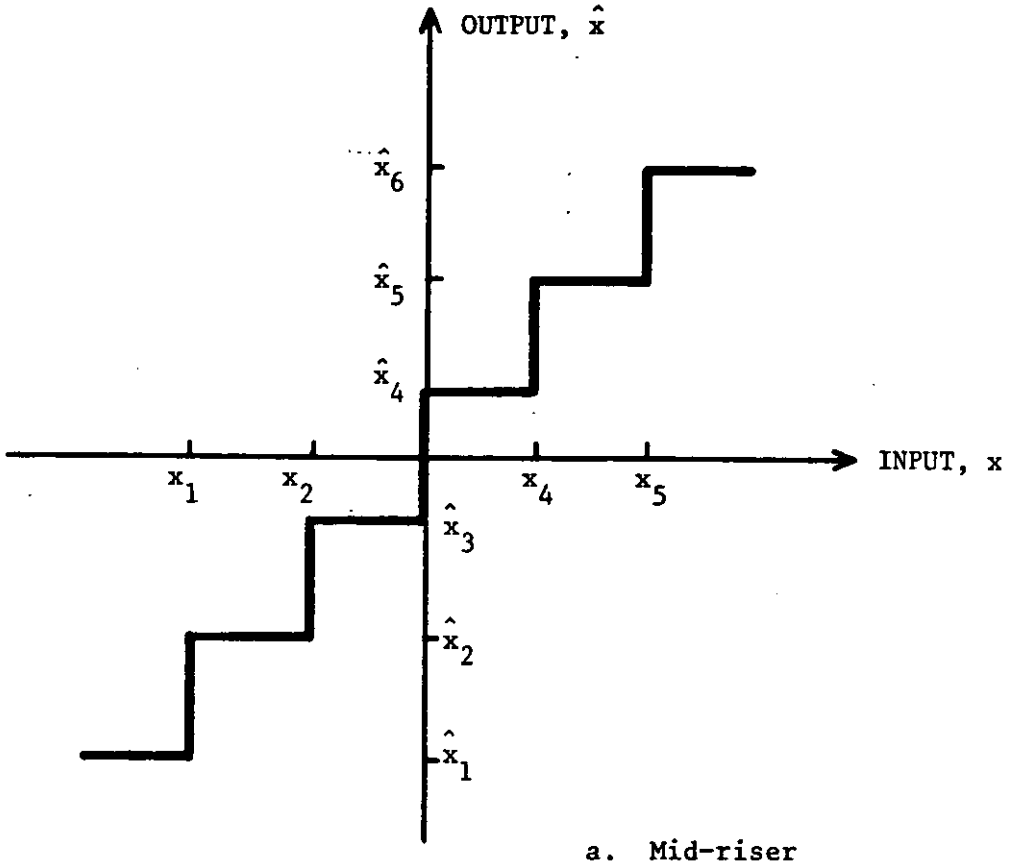
Numerous quantizers are employed in speech coding, and they are broadly categorized as either time-invariant, FQ, or time-variant

(adaptive) quantizers, AQ. The former is more common, simple and sometimes referred to as zero-memory quantizer, i.e., the quantized output is calculated from the corresponding input sample without the involvement of previous samples. If the input sample lies in the k^{th} interval of quantization,

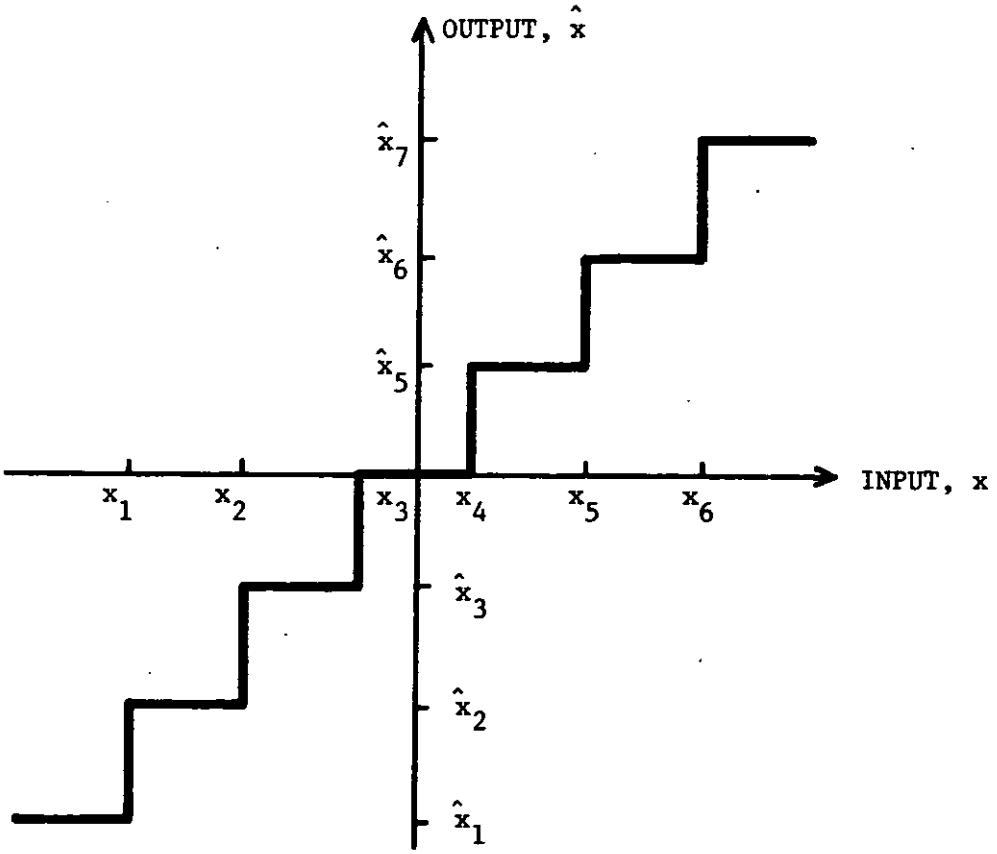
$$x_k < x < x_{k+1} \quad , \qquad 1 \leq k \leq N_Q+1$$

the output of quantizer is $\hat{x}_{k+1}$. For $1 \leq k \leq N_Q+1$, there are N_Q quantization levels, together with N_Q+1 input thresholds. The transfer function of typical quantizer is a stair-case characteristic. In practice, two versions of this characteristic, known as mid-riser and mid-tread are commonly used, see Figure 2.12. Mid-riser quantizer has its decision level at the origin, while the mid-tread has zero output level. Mid-riser characteristics are preferred due to their symmetrical properties and efficient use of $N_Q=2^b$ levels, despite the fact that low level signals, as in the silence section of speech, can not be expressed properly due to the existence of non-zero level. Hence, this sort of quantizer results in oscillations for low level signals. Crochiere[40] has suggested a kind of switch that exploits both mid-riser and mid-tread characteristics. The binary codes associated with both types of quantizer characteristics can be either natural-binary code, NBC, or folded-binary code, FBC.[41]

The output approximation of the quantizer introduces distortion-noise, known as *"granular noise"*, provided the input speech sample lies between $x_1 \leq x_k \leq x_{N_Q+1}$. If the input sample lies outside the specified range, the output is said to be overloaded and *"overload noise"* occurs.



a. Mid-riser



b. Mid-tread

FIGURE 2.12: Typical Uniform-Q Characteristics

The total noise introduced is the combination of these types of noise. A simple statistical model of the quantizer associated with its granular noise is shown in Figure 2.13. The model is based on the following assumptions:[11]

a) the noise is stationary, an assumption that is valid for small step size, Δ, and when number of levels, N_Q is large.

b) the noise is uncorrelated with the original input (implies no slope overloading)

c) the probability density function, p.d.f., of the noise is uniform and given by $p_d(q)=1/\Delta$ between $\pm\Delta/2$, and $p_d(q)=0$ outside the range, $\pm\Delta/2$.

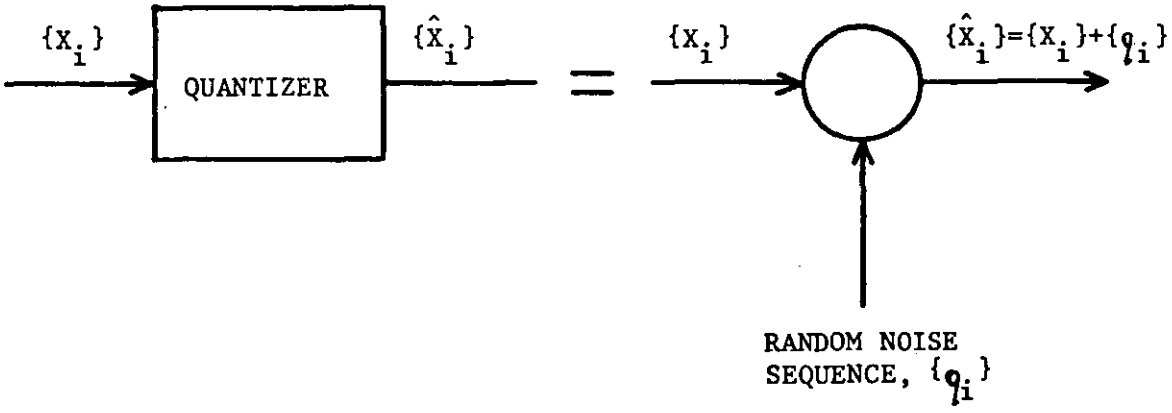


FIGURE 2.13: Quantization Model

Hence, the mean-square value of the quantization noise, σ_q^2, is

$$\sigma_q^2 = \frac{\Delta^2}{12} \quad . \tag{2.11}$$

Therefore, uniform quantizers are defined by 3 parameters, N_Q, Δ and overload limit, V respectively. An appropriate value of V for which the quantizer is not overloaded is selected in accordance with the ratio of V to the input r.m.s. value, σ_x. This ratio is known as *"loading factor"*, d_L. A common choice of d_L is the so called four-signal loading which ensures that almost all the input samples will lie inside the range $\mp 4\sigma_x$. For Gaussian and Laplacian p.d.f.'s with zero mean, only 0.01% and 0.35% of speech samples fall outside this range, $\mp 4\sigma_x$, respectively. Therefore, Δ, for uniform quantizer is

$$\Delta = \frac{2V}{N_Q} \tag{2.12}$$

or

$$\Delta = \frac{8\sigma_x}{2^b} \tag{2.13}$$

From Equations (2.11) and (2.13),

$$SNR = k.2^{2b}, \qquad k=3/16 \tag{2.14}$$

or in dB's,

$$SNR(dB) = 6b-7.3 \tag{2.15}$$

i.e., SNR(dB) increases linearly with b. Equation (2.15) is accurate for $b \lesssim 9$, above this value overload noise due to values outside the four-sigma loading becomes significant.

The quantizer described here has an error with a constant noise that is independent of the signal amplitude. Higher values of SNR

can be obtained by exploiting the p.d.f.'s of the signal being quantized. Quantizers, designed according to their p.d.f.'s of the input signal are known as optimum quantizers, Opt-Quant, and are well suited to speech signal characteristics. Opt-Quant's are basically non-uniform quantizers and the distribution of the levels is selected according to the p.d.f. of the signal. Alternatively, Opt-Quant's can be considered as uniform quantizers having a compressor which compresses the input samples prior to quantization. Compression is accomplished by means of a non-linear element, say H(x), in accordance with the p.d.f. so that more levels are introduced for low level signals due to their higher probability of occurrance. The output of the uniform quantizer is expanded through a function, $H^{-1}(x)$. This process of compression at the encoder and expansion at the receiver is known as companding.

For large values of N_Q (>128), the mean-square error power of Opt-Quant is given by,[4,42]

$$\sigma_e^2 = \frac{V^2}{3N_Q^2} \int_{-V}^{V} P_d(x) \left\{\frac{dx(t)}{d\hat{x}(t)}\right\}^2 dx(t) \qquad (2.16)$$

where $x(t)$, $\hat{x}(t)$, $\mp V$ and $P_d(x)$ are the quantizer input, output, amplitude range and p.d.f. of its input signal, respectively. Most of the research has been conducted on the assumption of N_Q being large. Panter et al[43] gives the relationship similar to Equation (2.16) for large N_Q,

$$\sigma_e^2 = \frac{2}{3N_Q^2} \left\{\int_0^V [P_d(x)]^{1/3} dx\right\}^3 \qquad (2.17)$$

where the p.d.f., $P_d(x)$ is even and valid between $\mp V$ and zero outside the interval. Equation (2.17) is the simplified version of Equation (2.16) with

$$\frac{d\hat{x}(t)}{dx(t)} = B[P_d(x)]^{1/3} \; ; \quad B = \frac{V}{2\int_{-V}^{V}[P_d(x)]^{1/3}} \tag{2.18}$$

The author also assumes that the quantization levels are very close together such that $P_d(x)$ is almost constant over the quantization intervals, and overload noise is neglected.

For non-uniform, i.e., *"Opt-Quant"* quantizers, a similar relation to that depicted in Equation (2.15) holds, namely,

$$SNR(dB) = 6b-K \tag{2.19}$$

where K is the function of the normalized p.d.f., $P_d(x)$, (the ratio of $P_d(x)$ to σ_x^2) that makes SNR independent of signal power. K is found to be 4.3 dB for signals with Gaussian p.d.f., assuming that overload noise is neglected. Comparison of Equations (2.15) and (2.17), for signals with Gaussian p.d.f. reveals that 3 dB gain in SNR is achieved over uniform quantizers.

Another approach in designing Opt-Quant's is due to Max[44], Stroh-Paez[45] and Paez-Glisson.[46] Max, in his often referenced paper, used iterative techniques so that the analysis is generalized and valid for both uniform and non-uniform quantizers (for signals having stationary p.d.f.'s). No restriction is imposed on N_Q. Input and output levels are calculated such that $\partial\sigma_q^2/\partial x_k$, $k=2,3,\ldots,N_Q$, and

and $\partial\sigma_q^2/\partial\hat{x}_k$, k=1,2,...,$N_Q$ are zero, see Appendix A. Hence,

$$x_k \doteq \frac{1}{2}(\hat{x}_k + \hat{x}_{k-1}) \tag{2.20}$$

$$\hat{x}_k = \frac{\int_{x_k}^{x_{k+1}} xP_d(x)dx}{\int_{x_k}^{x_{k+1}} P_d(x)dx} \tag{2.21}$$

From Equations (2.20) and (2.21), we note that

a) the threshold level, x_k, is half way between two successive output levels,

b) $\hat{x}_k$ is the conditional mean value of the input x for $x_k<x<x_{k+1}$.

The iterative process starts with an initial guess of $\hat{x}_1$, and x_2 is obtained from Equation (2.20) and $\hat{x}_2$ from Equation (2.21), and so on for other quantized levels. An apriori knowledge of the p.d.f. is required. Fleischer[47] shows that Equations (2.20) and (2.21) are sufficient for Opt-Quant having a specified p.d.f provided

$$\frac{d^2}{dx^2}\log[P_d(x)] < 0 \text{ , for all } x.$$

Paez-Glisson[46] applied Max's algorithm to the signals having Laplacian and Gamma p.d.f.'s.

All the Opt-Quant's described so far are matched to the p.d.f. of the amplitude levels of the input signal, and by this method, σ_q^2 is made small. In comparison to uniform quantization, the advantages of Opt-Quant for speech signals are:

a) Non-uniform spacing gives a lower quantization noise.

b) During idle channel conditions, the quantization noise performance of uniform quantizers is inferior to that of Opt-Quant as the latter has many levels near the origin.

However both the uniform quantizer and Opt-Quant have a poor dynamic range, (the dynamic range, DR of a system is generally defined as the range of input signal levels for which the SNR of the coder remains within 3 dB of the maximum), and the value of the peak SNR, obtained at one power level of the input signal, decreases rapidly with the changing levels of the input. In telephony, a large dynamic range of the input signal levels, typically 40 dB, is required. The quantizer is arranged over the desired dynamic range to be substantially independent of the input power level. This can be achieved by using logarithmic quantizers, Log-Q.

Log-Q's have similar peak SNR as uniform quantizer, but possess the virtue of a wider dynamic range. In practice, log-Q's are not truely logarithmic, having a law that is often linear at low signal levels and logarithmic at high signal levels. Two logarithmic laws are widely used. The μ-law proposed by Smith[35] is used for PCM systems in Japan and U.S.A. It is defined as,

$$H_{\mu}(x) = \frac{V \log(1+\mu x/V)}{\log(1+\mu)} \quad , \quad x>0 \qquad (2.23)$$

The compressor has odd-symmetry for negative values of x, $H_{\mu}(x)=-H_{\mu}(-x)$ and V is the overload level. A common choice for μ for 7-bit speech coders is either 100 or 255. In terms of dynamic range 7-bit log-PCM is almost equivalent to 11-bit uniform-Q-PCM.

Another type of companding proposed by Cattermole[5] and now accepted (outside Japan and U.S.A.) as the international standard, is the A-law. It is described by,

$$H_A(x) = \frac{Ax}{1+\log A}, \text{ for } 0 \leq x \leq \frac{1}{A}$$

$$= \frac{1+\log Ax}{1+\log A}, \text{ for } \frac{1}{A} \leq x \leq 1 \qquad (2.24)$$

where a typical value of A is 87.6 for a 7-bit speech coder. A-law and μ-law have similar performances when used in PCM speech coding.

Stroh-Paez[45] compared the Opt-Quant and log-Q, and concluded that although Opt-Quant gives subjectively 3-4 dB improvement in terms of SNR for a given N_Q, it has greater idle channel noise and poorer dynamic range, DR.

A different way of obtaining the aforementioned specification in digital telephony, i.e., 40 dB dynamic range with a SNR of 35-38 dB's is achieved by employing a quantizer whose step size is both large enough to cover the maximum signal range, and yet small enough to contain the quantization error. These demands on the step size magnitudes can be satisfied by arranging for the quantizer step size to adjust to the signal level being quantized. Such schemes, known as adaptive quantization, AQ, do not require a knowledge of the p.d.f. of the signal being quantized. They are capable of handling the rapid fluctuations of signal amplitudes. AQ's of Noll[48] and Stroh[49] are designed to match the short-term variance of the signal to be quantized over a block of length W samples; typically W may be 32 to 512.

Sometimes these AQ's are referred to as block adaptive quantizers where the calculation of step size is made using the input data. They are called adaptive quantizers with forward (explicit) determination of step size, and abbreviated to AQF.[50]

A different approach using backward adaptation has been studied by Jayant[33,36,37,51] who calculates the step size at every sampling instant from the transmitted data, hence the use of the backward. These adaptive schemes will be described in detail in Chapters III,V. However, we will state at this time that, AQ's provide a wide dynamic range, lower idle channel noise compared to the Opt-Quant's, and can achieve higher SNR than those of both Opt-Quant's and log-Q's.

2.3.3 Differential Pulse Code Modulation, DPCM

In DPCM, the difference between the input speech sample and a locally reconstructed sample is quantized, binary encoded and transmitted. At the receiver, the recovered speech sample is reconstructed from the quantized difference sample. This technique is effective in reducing the transmitted bit-rate compared to PCM because it exploits the correlation of speech signals, particularly, those of voiced speech. The difference signal being quantized has a lower variance, σ_e^2, than that of the original speech signal variance, σ_x^2. As a result, the range of quantizer to transmit the actual sample is considerably reduced, and fewer bits are required to code the difference signal for a given quality of reconstructed speech. The variance of the

quantization error in DPCM is proportional to the variance, σ_e^2, of the quantizer input, and if σ_e^2 can be reduced by, say a factor G, it is found that the variance of the quantization noise in the decoded speech signal is also reduced by G. Hence, the SNR increases by G compared to a PCM system. The DPCM system to be described here is primarily based on an invention by Cutler,[52] patented in 1952. DPCM is also known as *"predictive encoding"*. The term predictive is used because the quantizer input signal is formed by taking the difference between the actual speech sample and a locally reconstructed value that is formed using prediction of previously locally decoded speech samples. In 1966, O'Neal mathematically analysed predictive encoding of T.V. signals, while Nitadori[54] and McDonald[55] extended the work to speech signals. Stroh[49] also examined the performance of DPCM focusing attention on how the predictor and quantizer interact.

A typical DPCM coder is shown in Figure 2.14, where the symbols are displayed for the i^{th} sampling instant. The band-limited sequence of speech samples, whose value, x_i at the i^{th} instant is encoded into a binary word L_i and transmitted. At the receiver, the regenerated binary word, L_i' is decoded to give the recovered speech sample, $\hat{x}_i$. Assuming no transmission errors, the quantized output $\hat{e}_i$ is fed to the adders in both the encoder and decoder.

The locally recovered speech is

$$\hat{x}_i = \hat{e}_i + y_i \tag{2.25}$$

where y_i denotes the predicted value, and the quantized sample is

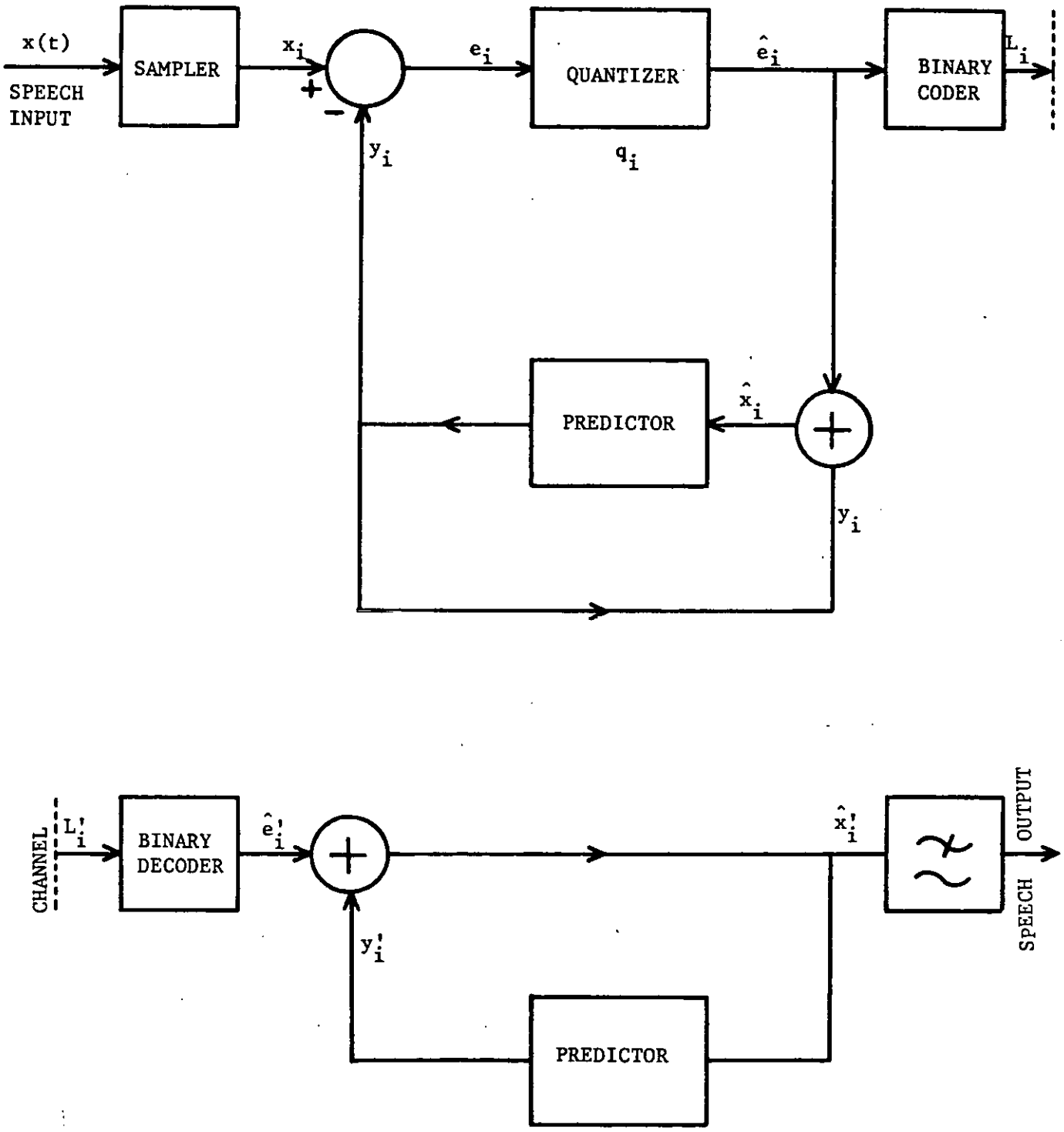


FIGURE 2.14: A Typical DPCM (Encoder-Decoder) System

$$\hat{e}_i = e_i + q_i \tag{2.26}$$

where e_i, q_i are the quantizer input and the quantization error samples respectively. As

$$e_i = x_i - y_i \tag{2.27}$$

then

$$\hat{x}_i = x_i + q_i \ . \tag{2.28}$$

Therefore, the i^{th} decoded sample is the sum of the speech sample and the quantization error sample at the i^{th} sampling instant.

$$\begin{aligned} q_i &= \hat{e}_i - e_i \\ &= \hat{x}_i - x_i \ . \end{aligned} \tag{2.29}$$

In Figure 2.14, it can be seen that the predictor operates on the locally decoded sample, $\hat{x}_i$ and not on input sample. It predicts y_i as the linear combination of past N decoded samples, viz.

$$y_i = \sum_{k=1}^{N} a_k \hat{x}_{i-k} \tag{2.30}$$

where N is the order of predictor and a_k's, k=1,2,...,N, are the predictor coefficients.

The SNR of the received signal $\hat{x}_i$ is defined here as,

$$SNR = \frac{\langle x_i^2 \rangle}{\langle (x_i - \hat{x}_i)^2 \rangle} \tag{2.31}$$

where $\sigma_x^2 = \langle x_i^2 \rangle$ is the mean-squared value of the input speech samples, given by

$$\sigma_x^2 = \frac{1}{NS} \sum_{i=1}^{NS} x_i^2 \tag{2.32}$$

where NS is the total number of sampling intervals over which the summation is performed. From Equations (2.29) and (2.31)

$$SNR = \frac{\sigma_x^2}{\sigma_q^2} \tag{2.33}$$

or

$$SNR = \frac{\sigma_x^2}{\sigma_e^2} \cdot \frac{\sigma_e^2}{\sigma_q^2} \tag{2.34}$$

The ratio of σ_x^2/σ_e^2 is the prediction gain and denoted by G, whereas, σ_e^2/σ_q^2 is the SNR of the quantizer and denoted by Q. Therefore, Equation (2.34) becomes

$$SNR = G.Q \tag{2.35}$$

In dB's

$$SNR(dB) = 10 \log_{10} G + 10 \log_{10} Q \tag{2.36}$$

or

$$SNR_{DPCM}(dB) = SNRI(dB) + SNRQ(dB) \ . \tag{2.37}$$

SNRI(dB) is the improvement factor in dB over PCM and depends on the auto-correlation of input speech samples, and the values of the prediction coefficients. The structure of the predictors and how the coefficients are computed is presented in Chapters III and IV.

Historically, most research activities have been done on first-order predictor, also known as leaky integrator. An ideal integrator has one coefficient, say "a_1", whose value is unity. The SNR gain of DPCM coder over PCM coder, G, for an ideal integrator is

$$G \simeq \frac{1}{2(1-c_1)} \tag{2.38}$$

where c_1 is the first correlation coefficient of the input speech

signal. In Equation (2.38), for G>1, c_1 must be greater than 0.5 for the performance of DPCM to be greater than that of PCM.

For $a_1=1$, the system is said to be non-optimal. The optimum performance is achieved when $a_1=c_1$, assuming that $\sigma_q^2 << \sigma_x^2$, and under this condition,

$$G \simeq \frac{1}{1-c_1^2} \quad . \tag{2.39}$$

Once again, as the value of c_1 is always less than 1, DPCM maintains its advantage of G over PCM. The SNR of the DPCM coder having a leaky integrator in its feedback loop is

$$SNR_{DPCM} = \frac{Q}{1-c_1^2} \quad . \tag{2.40}$$

McDonald[55] examined the various first-order predictors in the DPCM coder and concluded that the optimum prediction coefficient, $a_1=c_1$, is better than ideal predictor, $a_1=1$, in that it is more tolerant to the channel errors.

However, the relationships described by Equations (2.38)-(2.40) are theoretically viable assuming that:

a) $\sigma_q^2 << \sigma_x^2$

b) bits/sample, $b \geq 2$.

A more generalized relationship between SNR, c_1 and Q is given by O'Neal,[56] namely,

$$SNR_{DPCM} = \frac{Q(1-c_1^2/Q)}{1-c_1^2} \quad . \tag{2.41}$$

Equation (2.41) is independent of the assumption $\sigma_q^2 << \sigma_x^2$, and applies for a first-order Markov process input sequence. Q depends on the number of quantization levels, N_Q and p.d.f. of DPCM error sequence $\{e_i\}$. A reasonable estimate of Q can be made by assuming that the p.d.f. of $\{e_i\}$is identical to the p.d.f. of the original speech sequence $\{X_i\}$. SNR_{DPCM} values computed for Max's[44] optimum and uniform quantizers, where Gaussian inputs are assumed, and those calculated using Equation (2.41), are in close agreement with measured SNR values. The input signal to the quantizer in DPCM is not, however, truely Gaussian, being the convolution of two p.d.f.'s, namely those relating to $\{X_i\}$ and $\{q_i\}$. For large values of Q, Equation (2.41) is reduced to Equation (2.40).

The SNR value, Q for the quantizer may be estimated for Gaussian and Laplacian p.d.f.'s using Equation (2.17). From Appendix B, Q_G and Q_L, i.e., the quantizer SNR for Gaussian and Laplacian p.d.f.'s respectively are defined as

$$Q_G = \frac{N_Q^2}{2.73} \tag{2.42}$$

$$Q_L = \frac{2N_Q^2}{9} . \tag{2.43}$$

In deriving these relationships for N_Q-level, Opt-Quant's it is assumed that N_Q is large.

Also, $N_Q=2^b$, hence,

$$SNRQ_G(dB) \simeq 6b - 4.35 \tag{2.44}$$

and

$$SNRQ_L(dB) \simeq 6b - 6.53 \tag{2.45}$$

where b is the bits/sample.

For logarithmic quantizers[57] having μ=100 and μ=255,

$$SNRQ_{Log-Q}(dB) \simeq 6b - 8.5 \qquad (2.46)$$

and

$$SNRQ_{Log-Q}(dB) \simeq 6b - 10.1 \qquad (2.47)$$

respectively.

For small values of N_Q, Q values and SNRQ's are given in references (44,45,46) for signals having Gaussian and Laplacian p.d.f.'s.

2.3.4 Adaptive Differential Pulse Code Modulation, ADPCM

Fixed predictors designed on the knowledge of long-term signal statistics cannot be optimum at all times because of the non-stationary nature of the speech signal. Since the statistics of speech signals change with time, the predictor must also adapt to the changing statistics. If this adaptation is not done, the performance of DPCM, optimized for one type input statistics, experiences a loss in SNR when a signal with different statistics is applied. The main factors responsible for this reduction in SNR of a DPCM system are, viz:

a) the predictor, being no longer optimum,

and

b) the quantizer being mismatched to a new signal.

In order to achieve a more flexible DPCM system that can perform satisfactorily for all speakers, a variety of fixed and adaptive predictors and quantizers can be combined. The predictor can now be near to optimum in that it can follow variations in the signal with

considerable accuracy. A system that performs reasonably well for all speakers, and minimizes the necessity of a priori knowledge of the signal statistics, is referred to as ADPCM (either of which or both quantizer and predictor are adaptive). We now briefly describe some adaptive systems available in the literature.

2.3.4.A Block Adaptive Predictors[48]

One method of selecting prediction coefficients in accordance with the short-term variations of the speech signal is to update the predictor parameters periodically. In this way, the short-term autocorrelation coefficients in blocks of buffered input speech samples (with a duration of W/f_s sec.) are computed and then from Equation (2.9), $\hat{A}_{opt}$ is calculated for the same block. Vector $\hat{A}_{opt}$ is utilized for the encoding of the respective block of W samples. Since, the prediction coefficients are calculated from the input speech signal, the technique is referred to as Forward Block Adaptive (FBA) scheme. It follows that because the coefficients must be encoded and transmitted to the receiver in addition to the quantized sample $\hat{e}_i$, FBA scheme increases channel capacity. Further study of such schemes is given in Chapter III.

The second technique, known as Backward Adaptation method, computes the prediction coefficients from previously decoded speech samples and therefore eliminates the transmission of the prediction coefficients. The autocorrelation method can be employed on the locally decoded samples. However, it is not suitable for practical purposes since it requires the delay of one block.

2.3.4.B Adaptive Predictive Coding, APC

DPCM system usually does not attempt to predict the long-term redundancy in the input speech signal, namely pitch, as does a system, referred to as adaptive predictive coding, APC. APC is a DPCM system where the predictor is sub-divided into two components. This approach was used by Atal-Schroeder[58] whose long-term predictor exploits the fine spectral structure (quasi-periodic nature of voiced sound), while a short-term predictor is employed to exploit the spectral envelope of the speech (vocal tract shape). By this technique, the redundancy in the speech signal is removed in two steps. Figure 2.15 shows Atal-Schroeder's adaptive predictor. Long term predictor, $P_1(z)$, eliminates the redundancy due to similarity of the speech signal resulting from adjacent larynx pulses. This is achieved by a first-order predictor that is composed of gain and delay elements. Transfer function, T.F. of $P_1(z)$ is

$$P_1(z) = \beta_a z^{-M_p} \tag{2.48}$$

where M_p is typically a pitch period, calculated by a pitch detector. Constant β_a accounts for the amplitude variations from one pitch period to another. In the case of unvoiced segments of speech, β_a is almost zero and M_p is not significant.

The second predictor, $P_2(z)$, operates on the error signal, e_{1i}, i.e., output of predictor whose T.F. is $1-P_1(z)$. T.F. of the second predictor is

$$P_2(z) = \sum_{k=1}^{N} a_k z^{-k} \tag{2.49}$$

where N is typically 8.

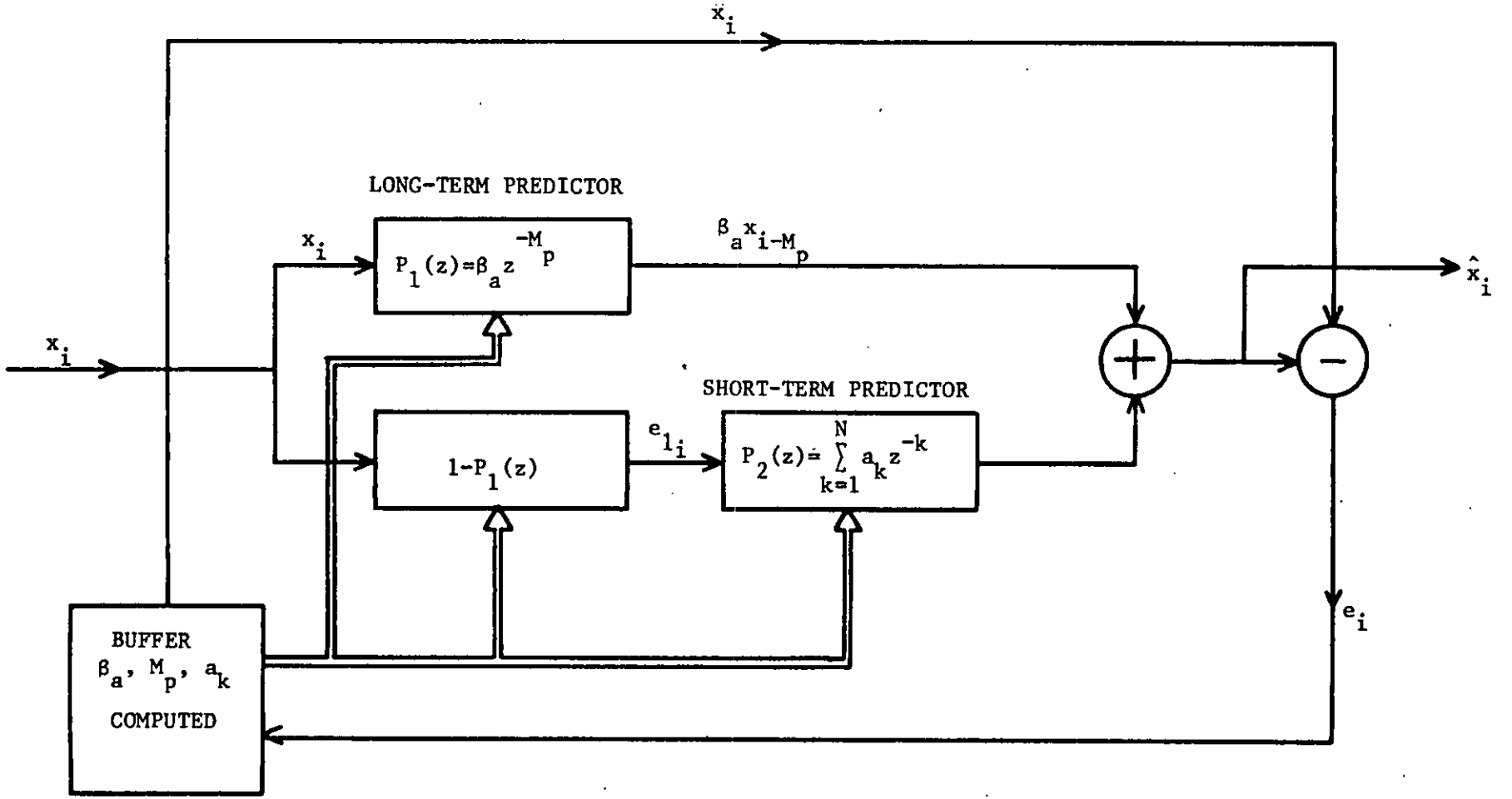


FIGURE 2.15: Atal-Schroder Adaptive Predictor

Therefore, the overall T.F. is given by

$$P(z) = P_1(z) + \{1-P_1(z)\}P_2(z) \tag{2.50}$$

The parameter M_p in Equation (2.48) is selected such that the correlation of the speech signal with a shift of M_p samples becomes a maximum or $\langle e_{1i}^2 \rangle = \langle (x_i - \beta_a x_{i-M_p})^2 \rangle$ is minimum. Hence the term $\langle e_{1i}^2 \rangle$ is minimized with respect to β_a. This results in

$$\beta_a = \frac{\langle x_i \, x_{i-M_p} \rangle}{\langle x_{i-M_p}^2 \rangle} \tag{2.51}$$

where $\langle . \rangle$ implies averaging over all the samples in a 5 msec. interval. The value of $\langle e_{1i}^2 \rangle$ for Equation (2.51) yields

$$\langle e_{1i}^2 \rangle = \langle x_i^2 \rangle \, (1-c_1^2) \tag{2.52}$$

$$c_1 = \frac{\langle x_i \, x_{i-M_p} \rangle}{\{\langle x_i^2 \rangle \langle x_{i-M_p}^2 \rangle\}^{\frac{1}{2}}} \tag{2.53}$$

where c_1 is the normalized correlation coefficient.

Thus, having decided on the value of β_a to give the minimum value of $\langle e_{1i}^2 \rangle$, the value of M_p is determined by the maximum value of c_1. Then Equation (2.49) is employed. The readjustment of the predictor parameters β_a, M_p and a_k coefficients, at every 5 msec., improve the prediction efficiency as the predictor considers both the excitation and vocal tract information in making its prediction.

When this predictor is used in DPCM, as shown in Figure 2.16, the error signal resulting from both types of predictor is quantized,

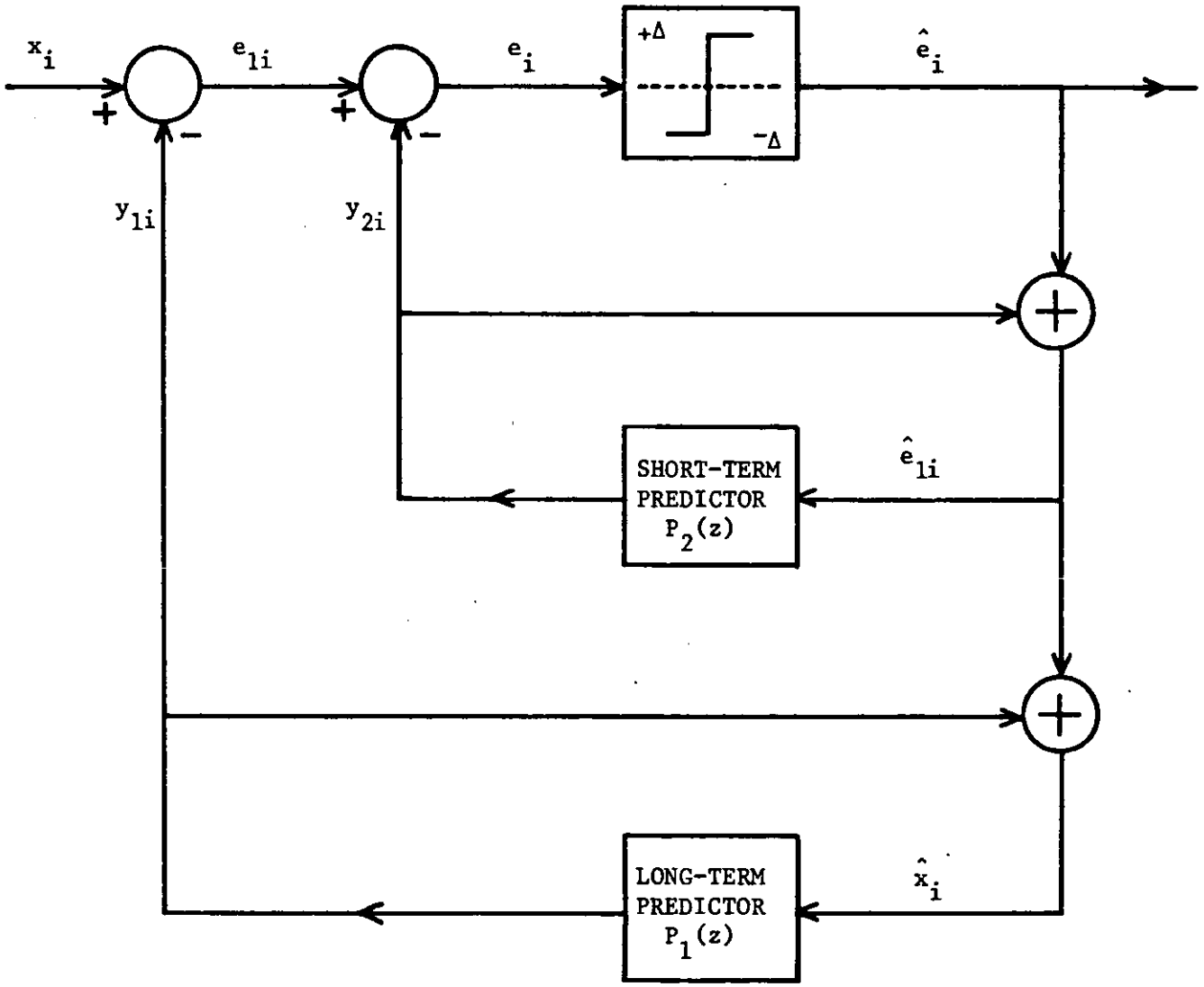


FIGURE 2.16: DPCM with Atal-Schroder Predictor = APC

encoded and transmitted, together with encoded values of M_p, β_a and a_k coefficients. In their simulations, the authors have used speech signal, sampled at 6.67 kHz and band-limited to 3.1 kHz. M_p is selected by maximizing c_1 for $20<M_p<150$ and N=8. The quantizer is a two-level, one-bit, quantizer with a variable step-size at every 5 msec. Although, they did not quantize M_p, β_a and a_k, but they did suggest that, in addition to 6.67 Kb/s for the error signal, 3 Kb/s are required for the predictor parameters, hence speech quality is sustained at a rate of 10 Kb/s. Subjective tests show that the quality of recovered speech was between the performance of 5 and 6 bit-log-PCM, i.e., SNR of approximately 24 dB.

The further improvement in SNR can be achieved by:

a. pre-emphasing the speech above 500 Hz prior to encoding and using the complimentary amount of de-emphasing filter after decoding. This is introduced since APC quantization noise has a flat spectrum, while the spectrum of voiced segments of speech tends to fall down above 500 Hz and gives low SNR at high frequencies.

b. introducing another gain, delay arrangement in $P_1(z)$, namely[13,59,60]

$$P_1(z) = \beta_a z^{-M_p} + \beta_b z^{-2M_p} \tag{2.54}$$

where β_a,β_b are calculated for minimum value of $\langle e_{1i}^2 \rangle$. This provides decoded speech which is superior to 6 bit-log-PCM, i.e., approximately SNR of 30 dB's.

c. using three estimates of the pitch, viz:

$$P_1(z) = \beta_a z^{-M_p+1} + \beta_b z^{-M_p} + \beta_c z^{-M_p-1} \quad . \tag{2.55}$$

Moye[17], reviewing Atal-Schroeder's results points out that if a system of reliably predicting larynx pulses can be made then Atal-Schroeder's system may yet prove of value, but SNR figures would certainly be lower if the prediction parameters were quantized. In addition, the complexity of calculation of a_k coefficients using covariance technique at every 5 msec and the computation of pitch period offsets the advantages reported here, in terms of implementation.

2.3.4.C Pitch Synchronous Techniques Used in ADPCM, PSADPCM

PSADPCM[60-61] is considered as middle ground between APC and ADPCM, in terms of both SNR performance and system complexity. The principle difference between APC and PSADPCM arises from the employment of multi-level quantizers in the latter case. The block diagram of PSADPCM is shown in Figure 2.17. Jayant,[61] in his PSADPCM employs two types of pitch extraction techniques, namely average magnitude difference function (AMDF) type of extractor and autocorrelation type of extractor. In Figure 2.17, $P_2(z)$ is fixed spectrum predictor, while $P_1(z)$ is adaptive pitch predictor and their i^{th}, output values are:

$$y_{1i} = \sum_{j=0}^{N2} a_{M_p+j}\, \hat{x}_{i-M_p-j} \tag{2.56}$$

$$y_{2i} = \sum_{k=1}^{N1} a_k\, \hat{x}_{i-k} \tag{2.57}$$

where M_p is the pitch period. Four sets of combinations of predictors are employed, namely:

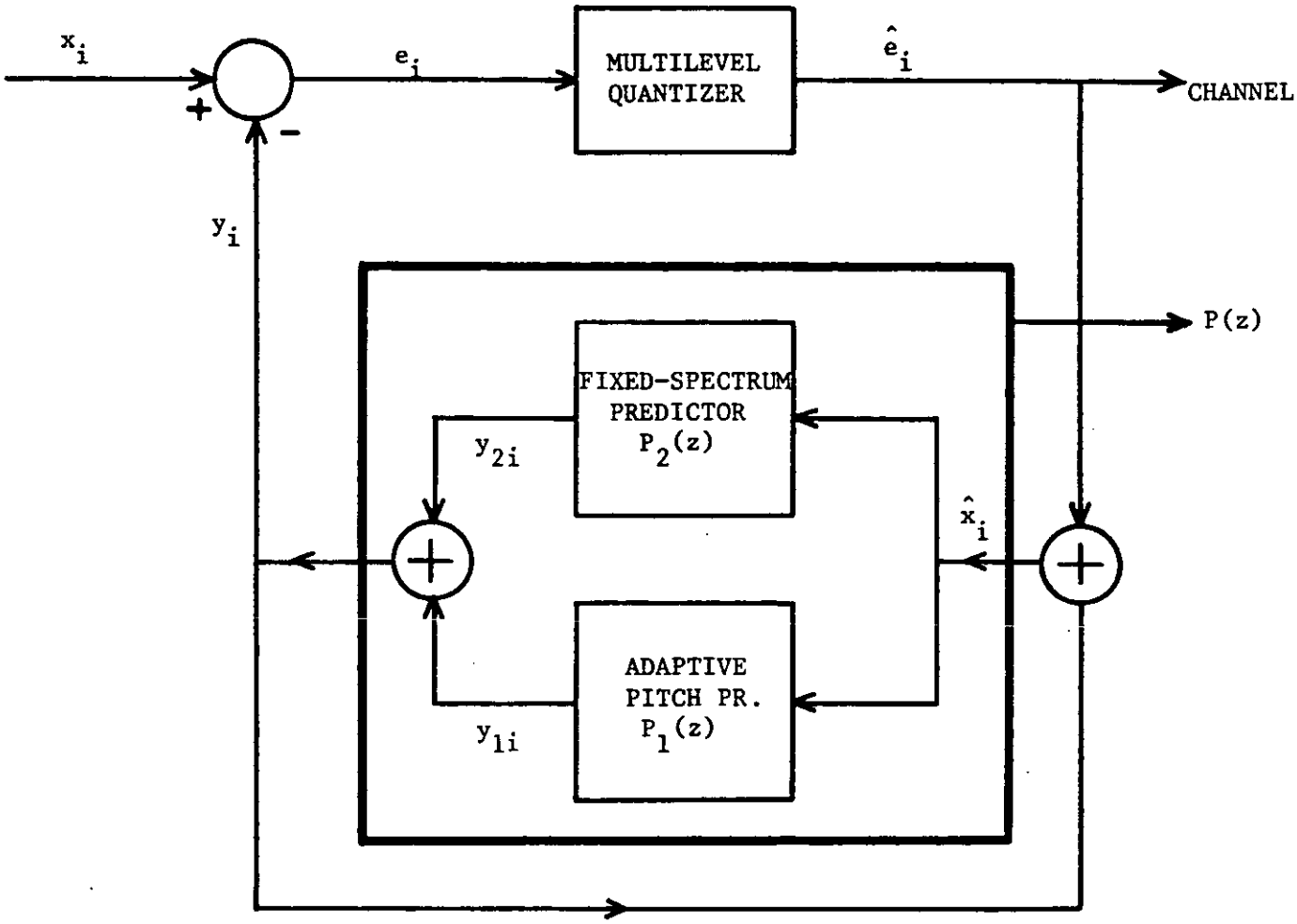


FIGURE 2.17: PSADPCM Encoder

a. k=1, j=0

$$y_i = a_1 \hat{x}_{i-1} + a_{M_p} \hat{x}_{i-M_p} \quad . \tag{2.58}$$

b. N1=N2=3

$$y_i = a_1\hat{x}_{i-1} + a_2\hat{x}_{i-2} + a_3\hat{x}_{i-3} + a_{M_p}\hat{x}_{i-M_p} + \ldots + a_{M_p+3}\hat{x}_{i-M_p-3} \quad . \tag{2.59}$$

c. N1=N2=1

$$y_i = a_1\hat{x}_{i-1} + a_{M_p}\hat{x}_{i-M_p} + a_{M_p+1}\hat{x}_{i-M_p-1} \quad . \tag{2.60}$$

d. N1=N2=0

$$y_i = a_{M_p}\hat{x}_{i-M_p} \quad . \tag{2.61}$$

where N1 and N2 are the orders of predictors.

(a-d) are employed when the speech signal has periodic behaviour, otherwise fixed spectrum predictor is used on its own. The switching between the two stages (periodic and non-periodic cases) are controlled by certain threshold levels calculated from AMDF technique. The Equation (2.61) is the simplest approach to PSADPCM and it generates highest SNR values when the correlations that are observed between x_i and x_{i-M_p} is very strong at 16 Kb/s (2 bits Q). Such a configuration together with adaptive quantizer offers 4 dB advantage in average SNR over conventional ADPCM employing 3^{rd} order fixed predictor. However, it necessitates approximately 1 Kb/s transmission-bit rate for transmitting pitch information.

In a thesis by Xydeas,[7] the samples in adjacent pitch are subtracted, after due allowance for differences in pitch duration, and the difference samples are encoded. By this method a 6 dB improvement was achieved over DPCM where both the pitch synchronous encoding arrangement and the conventional DPCM encoder used an ideal integrator and linear quantizer. The main differences between the systems advocated by Jayant[61] and Xydeas[7] are as follows:

a. Xydeas used an ideal integrator.

b. Non-equality of pitch periods are compensated in the latter case. Jayant, on the other hand did not perform this compensation and the difference between the pitch samples occasionally resulted in the quantizer having to handle large amplitude samples which caused overload noise.

2.3.4.D Sequential Predictors

These types of predictors arrange for the coefficients to be updated at every sampling instant from previously decoded samples, i.e., backward adaptation scheme. The beauty of such algorithms is that they eliminate both the computation of the matrix inversion and the need to transmit the coefficients to the receiver. The coefficients are updated at every instant in such a way that the prediction error is minimized. The k^{th} coefficient at the $(i+1)^{th}$ instant is found as:[60]

$$a_{i+1,k} = a_{i,k} + \zeta \qquad (2.62)$$

where $a_{i,k}$ is the value of the coefficient at the i^{th} instant and ζ is a function of previous predictor input samples, previous predictor error, pitch period and many other factors. Cummiskey[62] used a technique which minimizes the absolute error. Similar technique which minimizes the mean square error is known as Stochastic Approximation prediction (SAP). Gibson et al,[63] using an all-pole model, examined Kalman predictors for speech prediction and found them to be slightly better than SAP. In a different study, Gibson[64] compared the performance of SAP and Kalman predictors with a fixed predictor. He reported that at transmission bit-rates smaller than 16 Kb/s, the sequential predictors are superior, since in many cases fixed predictors may diverge,[65] i.e., they become unstable. However, at high bit-rates (>24 Kb/s), fixed and stochastic predictors behave almost the same, but the Kalman predictor produces a small improvement in SNR which is perceptually noticible. The details of these schemes are described in Chapters IV and V.

2.3.4.E Adaptive Quantizers

Although, the emphasis in this thesis is on predictors which are designed either using the locally decoded samples, $\hat{x}_i$, (sequential methods) or the original speech samples, x_i (fixed or block adaptive schemes), its performance is dependent on the presence of a quantizer when operating in a DPCM encoder. The quantizer and the predictor that can be designed to give optimum results in isolation, do not

necessarily behave optimally when operating in a DPCM encoder. We will consider the predictor-quantizer interaction in a DPCM encoder, and extend the discussion of Section 2.3.2 relating to quantizers.

Many algorithms have been devised for adaptive quantizers, AQ, so that the two adaptive components of a DPCM encoder may suit the changing statistics of the speech signal. The algorithm proposed by Stroh[49] calculates the step-size from the block of quantizer input samples of length W and designated this method as a forward quantization scheme. If the quantized samples are used to form the quantizer step-size, the quantization is known as backward adaptation quantization. Noll[48] produced similar algorithms and calculated the step size both from the input data and quantized data using predictors in his DPCM encoder that were either a first order feedback predictor or a N^{th} order predictor. These block methods assume that the signal is stationary over W samples and take the short-term variations of the input signal into account. These quantizers are also known as syllabically adaptive quantizers. A different approach proposed by Jayant[13,33,36,51] selects the step size at every sampling instant (instantaneously adaptive quantizers, AQJ) using a knowledge of previously quantized values. It is essentially a backward quantization scheme, and the step-size at i^{th} sampling instant is given by,

$$\Delta_i = \Delta_{i-1}^{\beta_q} . M(|L_{i-1}|) \qquad (2.63a)$$

where $M(|L_{i-1}|)$ is the time-invariant step size multiplier and $|L_{i-1}|$ is the magnitude of the code transmitted in the previous sampling instant. In practice, Δ_i is constrained by the quantizer to

$$\Delta_{min} \leq \Delta_i \leq \Delta_{max} \tag{2.63b}$$

resulting in the encoder having a dynamic range, DR, of approximately,

$$DR(dB) = 20 \log_{10} \left(\frac{\Delta_{max}}{\Delta_{min}}\right) \tag{2.63c}$$

The leaky constant, $\beta_q^{(66-68)}$, is introduced so that Equation (2.63a) becomes robust to channel errors. Gibson et al[63,64] also used such quantizer, AQJ, in their ADPCM encoders using adaptive predictors. AQJ was also used by Cheng[24] in conjunction with lattice predictors, and obtained good quality of speech at 1.12 bits/sample, a prediction gain of 10 dB over PCM, N being 8.

The quantizers described by Noll,[48] Jayant[13] lack the ability to cope efficiently with pitch variations in the signal assuming that the predictor is essentially attempting to remove the vocal tract information. The quantizer in this situation is often required to handle a residue signal which may have high amplitudes occur at pitch epochs. A backward quantizer which adapts its step size in a similar manner to AQJ, and also takes into consideration the high amplitude excitation pulses has been studied by Cohn-Melsa[69,70] and Qureshi-Forney.[71] Cohn-Melsa proposed a quantizer which has a set of high quantization levels that can accommodate the occasional high amplitude error sample. When the quantizer output corresponds to one of these high levels, the algorithm behaves as if a pitch pulse has occurred. The resulting large step size is then allowed to rapidly decay in order to handle subsequent small amplitude error samples. Thus, there are two modes of step size adaptation, one for the small residue sample (syllabic

adaptation), and the other for coping with the rare large amplitude excursions. The syllabic adaptation is controlled by the local time average of the magnitude of decoded signal sequence, $\{\hat{x}_i\}$. Quantizers operating in this way are referred to as pitch compensating quantizers, PCQ, and in general they employ mid-tread quantization characteristics. Qureshi-Forney have used two-loop Jayant Quantizer in their PCQ, i.e., syllabic adaptation is also accomplished using Jayant's algorithm. Further Xydeas et al[72] studied a different approach which produces a much faster adaptation and a greater reduction in the variance of the input signal to a fixed quantizer. Here, an adaptive quantizer was represented by a fixed quantizer having an adaptive structure at its input together with a non-linear element. Such a quantizer achieves a 3-4 dB SNR gain over AQJ.

In all quantizers, the quantized output level is binary coded by assigning a code word of length $\log_2 N_Q$, where N_Q is the number of quantization levels. A simple approach to code word assignment is to assume that the probability of occurrances of each level is equal and designate the code words accordingly. However, due to the non-stationary nature of the samples being coded, gains in SNR can be obtained by using variable length coding, VLC. Specifically short code words are assigned to quantization levels that occur with a high probability and long code words are given to those levels whose occurrance is low. The VLC is often called entropy coding, because, the average code word length is almost the same as the entropy of the symbols to be transmitted. VLC can produce average code word lengths less than $\log_2 N_Q$ bits/sample. Alternatively, at a given bit rate, the

number of quantization levels can be increased, resulting in a higher SNR. O'Neal[73,74] examined entropy coding in DPCM using optimum Gaussian, Laplacian and Gamma quantizers and achieved 2.81, 5.60, 10.2 dB's SNR advantage respectively, over a system without VLC's.

VLC also yields significant improvement in PCQ schemes since the probability of occurrance of the pitch pulses is low. By assigning long code words to the additional 2 outer levels in a 3 level quantizer (constant length code word), Cohn-Melsa[69] reported in their (3/5) level quantizers only a 10% increase in bits/sample (1.25 to 1.37 bits/sample), while in (3/5) level quantizers improvement is about 47%, namely from 1.58 to 2.33 bits/sample. Makhoul et al[75] obtained 2.1 bits/sample using VLC on a 19 level quantizer, in contrast to 4.25 bits/sample when the code words are assigned assuming they could occur with equal probability. VLC requires a buffer at the transmitter and receiver: no buffers are required in the case of all code words being the same length. A buffer smooths the fluctuations between the coder and channel operating at variable rate and it is transmitted over the channel at a uniform rate.

2.3.5 Quantization Noise Spectrum

The noise spectrum is a key factor in determining the subjective quality of the received speech. Consequently, the relationship between the spectrum of both the input speech and noise signal should be taken into account in designing a DPCM system. The long-term spectrum of

voiced speech decreases at -6,-12, dB/octave, while the noise spectrum tends to be uniform. Therefore, the SNR of the reconstructed signal falls with increasing frequency. At high frequencies, the low amplitude of the spectral components may be dominated by the quantization noise. To overcome this effect shaping of the noise spectrum to lie beneath the formant structure of the speech may be employed. This approach was initially reported by Atal-Schroeder[58] as it was discussed in Section 2.3.4.A and they achieved improved results using fixed both pre-emphasis and de-emphasis filters. Currently, noise shaping systems, primarily based on APC, are receiving a lot of attention as will be presented below.

2.3.6 Related DPCM Codecs

All the DPCM systems described so far employ prediction algorithms that attempt to remove vocal tract information, and in the more elaborate versions, pitch information. The resulting residual signal is encoded and transmitted. The receiver using, ideally, identical predictors, converts the residual signal back into speech. The predictors of DPCM, ADPCM encoders tend to minimize the mean square value of the prediction error regardless of the perceptual effects of the noise.

A system called noise feedback coder (NFC)[60,76] is another type of differential encoder which attempts to shape the noise spectrum to reduce the perceptual effect of the noise. NFC subtracts the quantized output from its input to give the quantization error. This error is

then filtered by filter F2(z) in Figure 2.18. The filtered noise signal is then subtracted from the input speech signal. Once, the filters F1(z) and F3(z) are selected from vocal tract and pitch information considerations and then the design of F2(z) ensures the restoration of the speech at the receiver has noise with a spectral composition which is located beneath formants.

Another differential encoding system called D*PCM, also known as a prediction error coder (PEC), is shown in Figure 2.19. Noll analysed[76] D*PCM and concludes that it is an intermediate stage between PCM and DPCM. This is true in the sense that D*PCM is both DPCM having its quantizer outside the loop and PCM having pre- and post-filters. As in DPCM, D*PCM has two identical predictors but they have different input signals. The disadvantage of D*PCM is to produce an error accumulation effect since the positive feedback at the receiver emphasises the quantization noise. The optimum coefficient for first-order predictor used in D*PCM is

$$a_1 = \frac{1}{c_1} \left[1 - \sqrt{1-c_1^2}\right] \tag{2.64}$$

and SNR gain over PCM is

$$G = \frac{1}{\sqrt{1-c_1^2}} \quad . \tag{2.65}$$

Comparison of Equations (2.65) and (2.39) points out the aforementioned fact that, in terms of G, D*PCM is inferior to DPCM. However, D*PCM is a system which behaves as a partially whitening filter, whereas DPCM is a full-whitening filter and both systems have the same sensitivity to channel errors at high bit-rates.

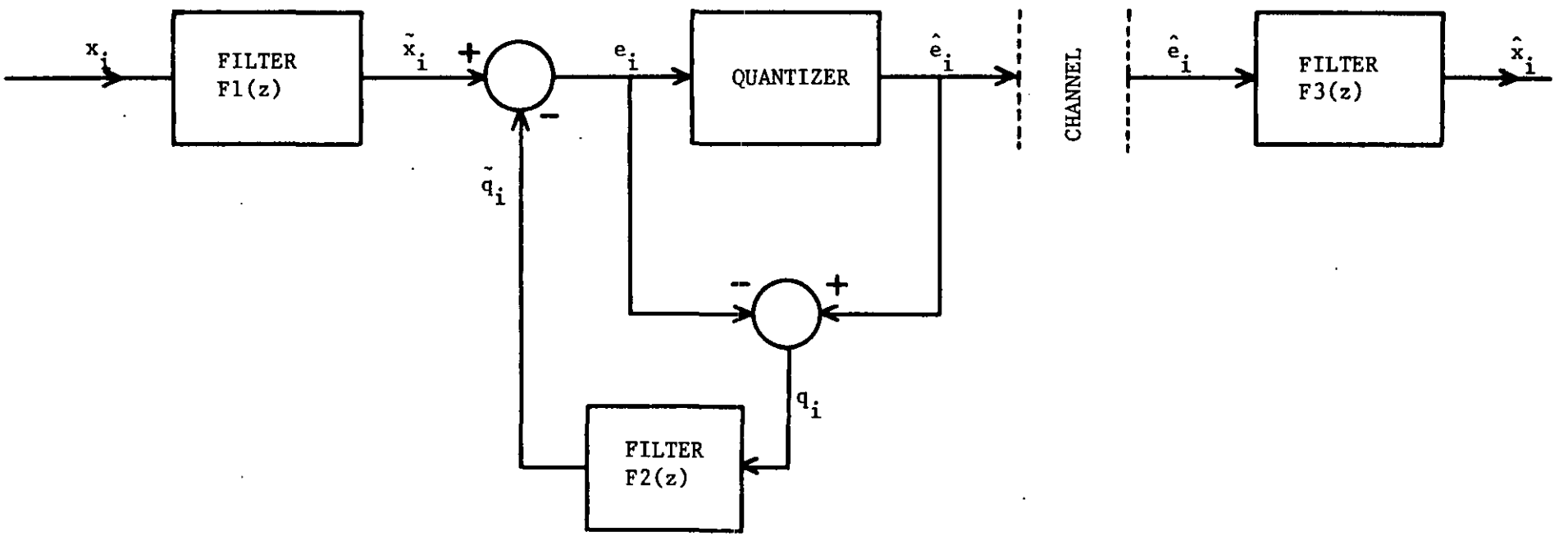


FIGURE 2.18: Noise Feedback Coder at i^{th} Instant

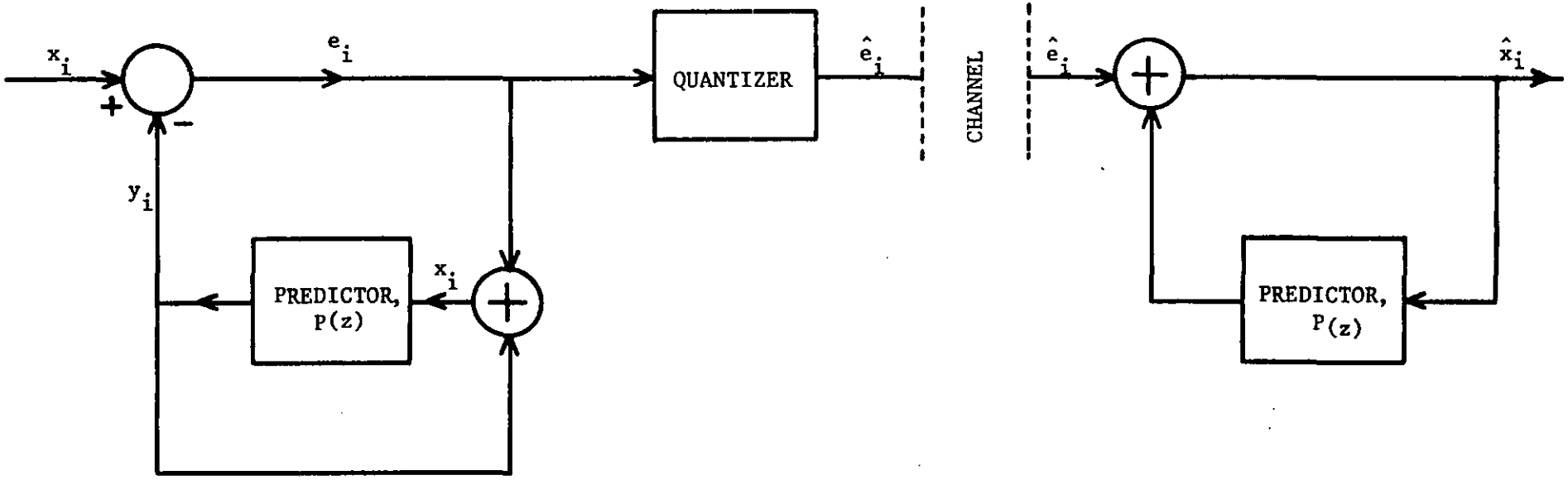


FIGURE 2.19: D*PCM System

NFC can be employed with D*PCM to shape the quantization noise with $F1(z)=1-P(z)$, $F3(z)=1/(1-P(z))$. Such a configuration produces good quality speech since it pre-filters the input speech at high frequencies and post-filters recovered speech at the receiver to eliminate the spectral distortions caused by pre-emphasizing. In addition, F2(z) redistributes the noise in the spectrum, decreasing it at high frequencies.

The general form of a differential coder attempts to remove vocal-tract and excitation from the speech signal, and applies noise shaping, as is shown in Figure 2.20. We observe that it is essentially the same as APC, see Figure 2.16, with the introduction of spectral noise masking.

Atal-Schroeder[78,79] investigated the noise shaping system under the following conditions: $F1(z)=1-P2(z)$ where $P2(z) = \sum_{k=1}^{N} a_k z^{-k}$, N is typically 10 (vocal-tract filter) and they remove the short-term predictor, P2(z) from the feedback loop. Long-term, pitch-predictor, P1(z) having 3 prediction coefficients is defined by Equation (2.55). Quantizer has 3 levels. The predictor and quantizer parameters are updated at every 10 msec. In addition, the authors employ pre-emphasis $(1-0.4z^{-1})$ and de-emphasis, $(1-0.4z^{-1})^{-1}$ and output of F2(z) is peak-limited in order to eliminate instability problems. In selection of F2(z), 3 configurations of noise shaping filter are examined, viz:

a) $F2(z)=0$; this gives the quantization noise spectrum having the same envelope as the input spectrum. The SNR is 13 dB, but the recovered speech is noisy.

b) $F2(z)=P2(z)$; this choice provides SNR of 23 dB and recovered speech is less noisy than in (a). However, the high SNR values

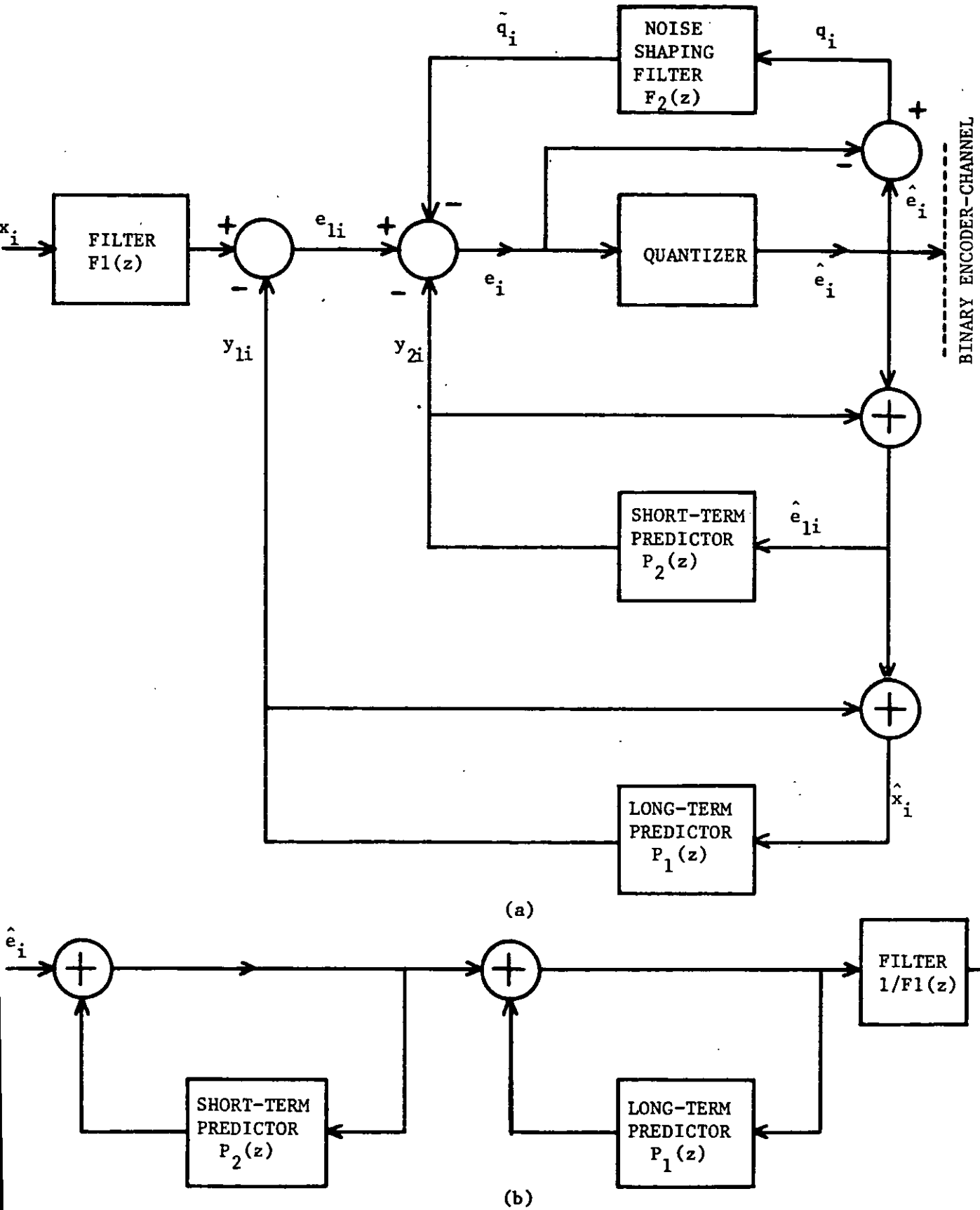


FIGURE 2.20: General Structure of Differential Coder-Decoder,
a. Transmitter b. Receiver

at the formants are degraded between the formants. Generally, speaking, choice (a) is good for high-frequencies, while (b) is for low frequencies.

c) $F2(z)=P2(\beta_a z^{-1})$ where $0<\beta_a<1$. $\beta_a=0$ and $\beta_a=1$ correspond to choice (a) and (b) respectively. The suitable value of β_a between $0<\beta_a<1$, increases the bandwidths of zeros of $1-F2(z)$, hence SNR is improved in the required band. The output SNR in this case is 21 dB and the quality of reconstructed speech is comparable to that of 7 bit-log PCM (SNR ~ 33 dB).

Makhoul and Berouti[75] have also reported on the combination of noise shaping and differential coding. Unlike Atal-Schroeder, they removed the pitch loop, i.e., $P1(z)=0$ and $F1(z)=1$. $P2(z)$ and $F2(z)$ are given by

$$P2(z) = \sum_{k=1}^{N} a_k z^{-k} \quad , \quad N=8 \tag{2.66}$$

$$F2(z) = \sum_{j=1}^{P} f''_j z^{-j} \quad , \quad 1<p<N. \tag{2.67}$$

The authors investigated both all-pole and all-zero designs for $1+F2(z)$. In all-pole design, $1+F2(z)$ is selected so that

$$1+F2(z) = \frac{1}{1+P2(z)} \quad , \tag{2.68}$$

where $P2(z)$ parameters are selected from the input speech signal. For $p=1$ or 2, the output speech contained low-frequency "*rumble*" and this roughness dominated for $p>2$. In all-zero design, the coefficients of $1+F2(z)$ are selected such that $1+F2(z)$ becomes an optimally inverse to $1+P2(z)$. This is simply achieved by calculating the correlation that exists between the a_k coefficients,

$$c_j = \sum_{k=0}^{N-|j|} a_k a_{k+|j|} \quad , \quad 1<j<p \tag{2.69}$$

and $a_0=1$. Therefore for p=1, {1+P2(z)} has the coefficient of $(-c_2/c_1)$ and for p=2 the coefficients of {1+P2(z)} are $c_2(c_3-c_1)/(c_1^2-c_2^2)$ and $c_2^2-c_1c_3/(c_1^2-c_2^2)$ respectively. In experiments, these configurations reduce both the hissing noise at high frequencies and the rumble and roughness at low frequencies.

In a final design, employing forward adaptive, entropy-coded quantizer (19 levels), predictor of N=8 and noise shaping filter of p=1 or 2, the authors obtained almost no difference between input and recovered speech during the subjective tests. The speech signal was sampled at 6.67 kHz, encoder was operating at 16 Kb/s, also quantizer and predictor parameters were updated every 25 msec.

As a conclusion of these configurations, described here, it is important to emphasise that the noise shaping techniques do not create any extra information to be transmitted to the receiver. Hence, the reduction of the perceptual effect of the noise results in some complexity in the design of transmitter at a given transmission bit-rate and improved quality of reconstructed speech specifically at bit-rates of 16 Kb/s and below.

2.3.7 Delta Modulation, DM

Delta Modulation is a simple differential quantization technique that is essentially a one-bit DPCM system, because it preceeded DPCM,

and is widely used, we will consider its operation in detail.

The input signal of DM is oversampled so that the successive speech samples become highly correlated. Therefore, samples can easily be predicted from one previous sample, using single-tap predictor, and a 2-level quantizer can provide reasonable SNR performance.[11] Consequently DM has a simpler structure than DPCM. In DM, the bit-rate is equal to sampling rate. Since its invention in 1946,[79] many papers have appeared on the subject. Steele,[8] in his often referenced book, presents an excellent comprehensive survey of delta modulation.

Like DPCM, DM coder can be either linear, non-adaptive (LDM) or adaptive (ADM). These classifications are usually associated with the techniques of calculating stepsize, Δ. In LDM, Δ is fixed, while in ADM, the stepsize follows the variations in the input slope according to a companding algorithm. We will now describe LDM and ADM in more detail.

2.3.7.A Linear DM, LDM Coder

Figure 2.21 shows a LDM coder having a 2-level quantizer with a fixed step size Δ, and single-delay predictor having a coefficient of unity. The feedback loop is composed of a delay D of one sample period followed by an accumulator. Transfer function of feedback loop is D/(1-D).

In the encoder, locally estimated speech sample $\hat{x}_{i-1}$ is subtracted

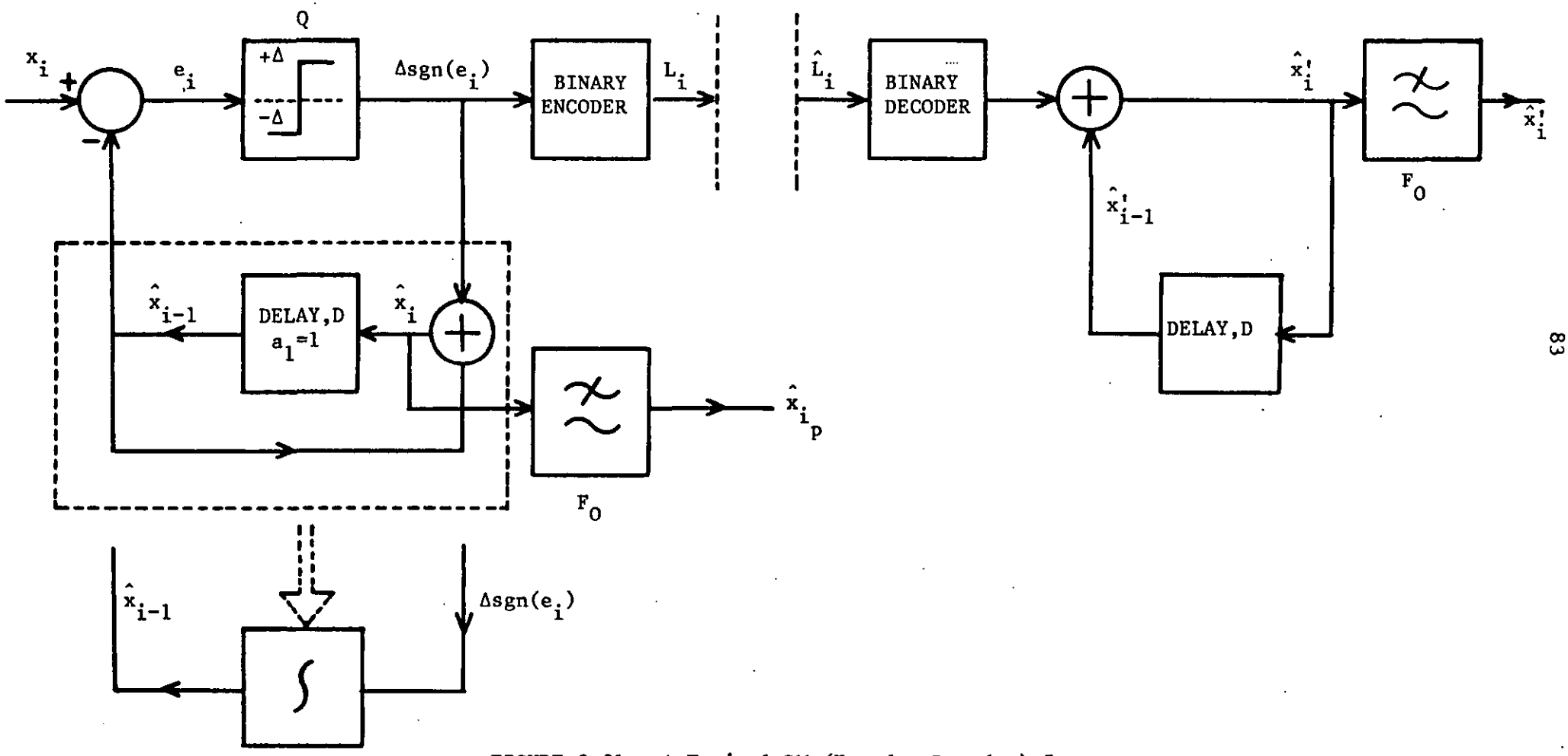


FIGURE 2.21: A Typical DM (Encoder-Decoder) System

from the actual speech sample x_i to form the error sample e_i which is subsequently quantized. The polarity of e_i is transmitted as a one-bit binary word L_i. The Equations governing the delta modulation system are:

$$e_i = x_i - \hat{x}_{i-1} \tag{2.70}$$

$$\hat{x}_i = \hat{x}_{i-1} + \Delta \text{sgn}(e_i) \tag{2.71}$$

$$q_i = x_i - \hat{x}_i \tag{2.72}$$

therefore, $$e_i = x_i - x_{i-1} + q_{i-1} \tag{2.73a}$$

The decoder integrates the received binary signal to yield a signal that is a staircase version of the original speech signal. The sharp edges of the staircase signal are removed by the low-pass filter F_0, to produce a speech signal that is a good replica of the original speech, provided the bit-rate is sufficiently high ($\geq$32 Kb/s). In the absence of coder overload, the Equation (2.73a) may be written as

$$e_i \simeq x_i - x_{i-1} \tag{2.73b}$$

i.e., the error signal is approximately the derivative of the input signal.[80]

Two types of distortion arise in the encoder, and are known as "*granular*" and "*slope overload*" noise. Slope overload occurs if the staircase waveform $\hat{x}_{i-1}$ is unable to track the input speech signal. To avoid slope overloading the following inequality should apply,

$$\left.\frac{dx(t)}{dt}\right|_{max} \leq \frac{\Delta}{T} \tag{2.74}$$

where T is the sampling period and equal to $1/f_s$, and dx/dt is the

derivative of the speech signal. If the input signal is a sinewave, $x(t)=V_m \sin\omega_m t$,

$$\dot{x}(t)\Big|_{max} = \omega_m V_m \quad . \tag{2.75}$$

For no slope overload,

$$\omega_m V_m \leq \Delta f_s \quad . \tag{2.76}$$

Granular noise occurs when the encoder is tracking x_i, and is the result of inband noise in the tracking error signal. For correct idle channel conditions when there is no input signal, the binary output signal is a sequence of alternative logical ones and zeroes. This binary signal does not generate an output at the output of F_0. Idle channel noise occurs when the .....1010101010.... output pattern is not maintained due to encoder asymmetry, and a noise signal occurs at the decoder output.

If the output binary sequence, $\{L_i\}$ has a form of the type 10111111 or 001000000, the probability of having slope overload is high, while if $\{L_i\}$ is say 1100101110010110, granular noise dominates. Slope overload noise is reduced by increasing stepsize, Δ, whereas the opposite is true for granular noise. Step size adjustments according to binary sequences at the coder output are efficiently utilized in ADM coders. Perceptually, the overload noise is preferable to granular noise, since the latter has a uniform spectrum (see Section 2.3.5). Abate[81] proved an imperical formula for the optimum step size, Δ_{opt} which minimizes the total noise power, namely,

$$\Delta_{opt} = \sqrt{\langle (x_i - x_{i-1})^2 \rangle}\, \log_2 FR \tag{2.77}$$

where $FR=f_s/2f_c$ (f_c is the bandwidth of the speech signal) and Equation (2.77) is valid for $FR \gg 1$.

The exact calculation of SNR in DM has been studied by many authors.[82-84] Some concentrated on granular noise,[82] while others investigated only slope overload noise.[83-84] O'Neal examined both types of noise and added the variance components due to each distortion for Gaussian signals.[85] Various formulations of SNR in DM can be summarized as follows:

a. De Jager[86] showed that the noise power is

$$\sigma_q^2 = K_q \cdot \frac{f_s^{-1}}{f_c^{-1}} \cdot \Delta^2 \tag{2.78}$$

The constant K_q is typically 1/3 for good encoding conditions. Equation (2.78) is applicable for flat noise spectrum in the message band, and

$$\text{SNR} = \left.\frac{\sigma_x^2}{\sigma_q^2}\right|_{\substack{\text{message}\\ \text{band}}} = \frac{f_c^{-1}.\Delta^{-2}}{K_q.f_s^{-1}} \cdot \sigma_x^2 \tag{2.79}$$

assuming no slope overload occurs.

b. Cummiskey[80] derived that

$$\text{SNR}\Big|_{\substack{\text{message}\\ \text{band}}} = \frac{(Q_{opt}-1)f_s/2f_c}{2(1-c_1)} \tag{2.80}$$

where Q_{opt} is $\langle e_i^2\rangle/\langle q_i^2\rangle$, c_1 is the correlation coefficient of speech.

In proving Equation (2.80) it was assumed that $e_i \simeq x_i - x_{i-1}$ and the resulting noise spectrum of the DM is flat, so that the ratio of total noise to the noise in the message band is $f_s/2f_c$. In a 2-level quantizer, Q_{opt} values for Gamma, Laplacian, Gaussian and Sinusoidal p.d.f. of quantizer input sequence are 1.50 (1.77 dB's) 2.0 (3 dB's),

2.75 (4.3 dB's) and 5.28 (7.2 dB's), respectively.

Equations (2.79) and (2.80) agree, and specify the upper bound in the case of single integrator. As an example, consider a sinewave, $V=V_m \sin 2\pi f_m t$ and $\sigma_x^2 \cong V_m^2/2$, where V_m is the amplitude and f_m is the frequency of the sinewave. At no slope overload, from Equations (2.76) and (2.79), the peak SNR, SNRP is written as,

$$SNRP\Big|_{\substack{\text{message}\\ \text{band}}} = 0.04 \frac{f_s^3}{f_c . f_m^2} \tag{2.81}$$

From Equation (2.80) it was also proved that[87]

$$SNRP\Big|_{\substack{\text{message}\\ \text{band}}} = 0.054 \frac{f_s^3}{f_c . f_m^2} \tag{2.82}$$

Observe that SNRP of DM for single integration is

$$SNRP_{DM} = k_{DM} . f_s^3 \tag{2.83}$$

(k_{DM} is constant),

while in PCM and DPCM, SNR is proportional to f_s.

De Jager[86] has shown that a DM can produce speech with quality equal to that of 7 bit log-PCM while operating at sampling rate of 120 kHz.

Another type of LDM is Delta-Sigma Modulator (DSM) in which the integrator[8] in the feedback loop of LDM is relocated in front of the quantizer, and the decoder is just a filter. Thus, the error sequence is integrated prior to quantization so that slope overload is independent of signal frequency f_m.

2.3.7.B Adaptive DM, ADM Coder

LDM has a peak SNR for one input power level when the coder is operating with a particular Δ and f_s. In the case of telephony,[88] this means that subscribers whose voices produce electrical signals having different power levels will be DM encoded with different SNR values. What is required is an almost constant SNR for all subscribers, i.e., a DM capable of operating with a similar SNR over a wide range of signal amplitudes. This is achieved by means of an adaptation strategy such that for a given f_s, Δ is allowed to decrease when the slope of the input speech is small, and vice versa. As the noise power $\langle q_i^2 \rangle$ depends on Δ, we attempt, in ADM, to arrange for $\langle q_i^2 \rangle$ to vary with signal power, resulting in an SNR that is independent of the signal power. (The same independence found in ADPCM systems). DM systems employing the adaptation algorithms are referred to as adaptive DM, or companded DM. Adaptation can be either syllabic (Δ changes at a rate which is dependent on the pitch or envelope information, or on the syllabic variations in the speech signal) or instantaneous (Δ changes at every sampling instant). Many different algorithms have been proposed in the literature.

Winkler's High Information Delta Modulation, (HIDM) coder[89] uses 3 consecutive pulses at the encoder output in its adaptation algorithm and is an instantaneous system. Adaptation logic makes a decision according to the polarity of the three pulses and selects the three multipliers. HIDM produces a high SNR, and is well-suited to pictorial signals due to its ability to cope with sudden changes in the input signal. However, at high transmission rates (64 Kb/s, HIDM operates satisfactorily with speech signals.

Another type of instantaneous companded DM is proposed by Jayant,[13,33,36,90,91] and is usually referred to as First Order Constant Factor Delta Modulation, (FCFDM) coder. The step size is calculated at every sampling instant in accordance with the polarity of the last two pulses at the encoder output. This is accomplished by two multipliers which are independent of the sampling frequency. Jayant optimized these multipliers (1.5 and 1/1.5) and obtained good telephony speech quality at a bit-rate of 60 Kb/s. At 40 Kb/s, degradation in quality is perceptible, while at 20 Kb/s the speech quality is significantly reduced, but most of the intelligibility is preserved. At 60 Kb/s SNR of FCFDM is 10 dB better than LDM for coding speech having a 3.3 kHz band-width. Also at low bit-rates, it behaves better than log-PCM, as SNR_{ADM} is proportional to f_s^3, while PCM increases 6 dB/ octave increase in clock rate.

A similar type of technique, referred to as the Second Order Constant Factor DM (SCFDM) coder investigated by Kyaw and Steele.[92] Instead of using the last two consecutive bits of FCFDM, SCFDM employs 3 recent bits and constitutes 8 possible binary patterns which are grouped in complementary patterns to give 4 different multipliers. For Gaussian input signal which is band-limited to 3.1 kHz, at 40 Kb/s, authors reported a 4.5 dB advantage in SNR over FCFDM.

In a different approach, known as the Song Voice Delta Modulator (SVDM),[93] the step sizes at i^{th} sampling instant generated as,

$$\begin{aligned} \Delta_i &= 2\Delta_0 \, \mathrm{sgn}(e_{i-1}) \quad , \text{ for } |\Delta_{i-1}| < 2\Delta_0 \\ \Delta_i &= |\Delta_{i-1}| \mathrm{sgn}(e_{i-1}) + \Delta_0 \mathrm{sgn}(e_{i-2}) \quad , \text{ for } |\Delta_{i-1}| > 2\Delta_0 \end{aligned} \tag{2.84}$$

where Δ_0 is the minimum step size of the system, typically 5-10 mV. The results show that at 16 Kb/s and 9.6 Kb/s, the word intelligibility is high and dynamic range is about 40 dB. This ADM, i.e., SVDM is closely related to the constant factor delta modulator.

In contrast to instantaneously companding, ADM employing syllabic companding changes its step size at a syllabic rate, dependent on properties of the speech signal (typically time constants of 5-10 msec., i.e., these associated with a pitch period are used). Continuously Variable Slope Delta Modulator (CVSD) finds favour in speech coding and is available on a single chip.[93] The CVSD coder is shown in Figure 2.22. (Notice that without the syllabic compander the CVSD reduces to an LDM). The DM step size is found from the output bit stream with the aid of an ℓ-bit shift register (ℓ is usually either 3 or 4 for f_s 30 kHz). Provided that 3 or 4 consecutive L_i's have the same polarity, pulse H is generated and activates the syllabic filter whose time constant, τ_2, is typically 5-10 msec. A pulse of height H_0 ($H_0 << H$) is added to ensure that the minimum step size is not zero. The output of syllabic filter, having a coefficient of a_2, is multiplied with the transmitted bit to give the step size Δ_i which is fed to the predictor (accumulator having $a_1 \approx 0.99$, $\tau_1 \sim 1$ msec.) At 16 Kb/s, suitable values of the time constants for the syllabic and predictor filters, namely, τ_1, τ_2, range from 1 and 6 msec,[93] to 6 and 25 msec[94] respectively. This shows that the second choice is more syllabic rate, while the first one is rather at pitch rate. The syllabic filter is designed to enable the DM to track the speech envelope. System functions are,

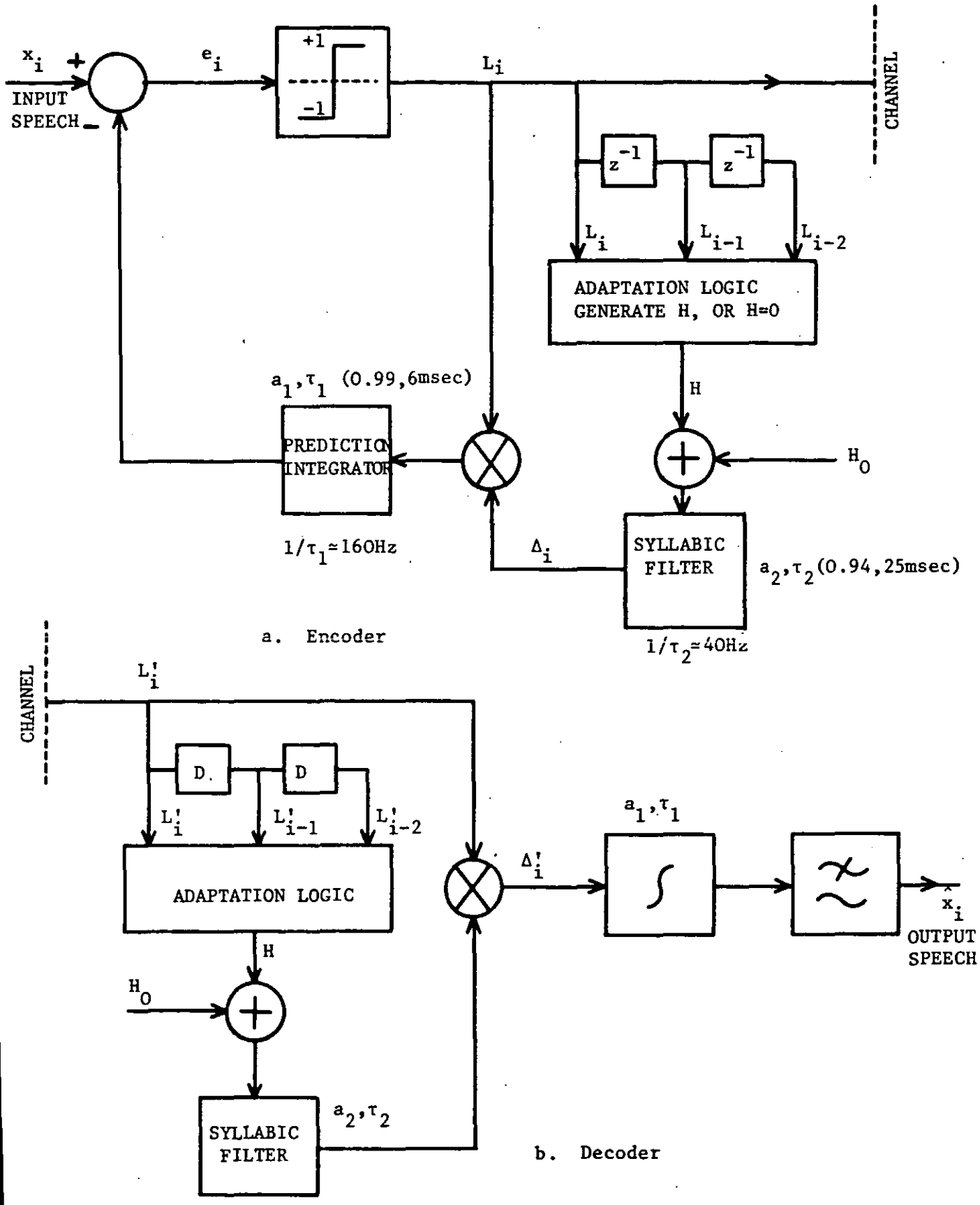


FIGURE 2.22: Block Diagram of CVSD (Encoder-Decoder) System

$$\Delta_i = a_2\Delta_{i-1}+(1-a_2)(H+H_0) \quad , \text{ for } L_i=L_{i-1}=L_{i-2}$$

$$\Delta_i = a_2\Delta_{i-1}+(1-a_2)H_0 \quad , \text{ otherwise.} \qquad (2.85)$$

Selection of τ_1,τ_2,a_1 and a_2 has an effect on the robustness of the coder to transmission errors.

Finally, in reference (94), CVSD and FCFDM are compared at 16 and 24 Kb/s. The results reveal that, in no channel error condition the dynamic range of FCFDM is wider than CVSD, whereas in a noisy channel FCFDM performance degrades rapidly as the error rate increases. CVSD is more robust to channel errors.

2.4 OTHER SPEECH CODING TECHNIQUES

PCM, DPCM and DM coders are time domain waveform coders. However, the speech band may be divided into a number of sub-bands, and waveform encoding processes applied to each band so that the reconstructed speech is the sum of the decoded values from each band. Such techniques are known as Frequency Domain Waveform Coders (FDWC).[13] They provide flexibility in assigning different bits/sample to each band since the noise is confined to each band, and appropriate coders can be designed for the different signal statistics in each band. Consequently, good quality of speech is obtained at lower bit-rates. Sub-band coding (SBC) and Transform Coding (TC) are typical examples of FDWC.

SBC[13,66,95-97] divides the speech band into 4-5 sub-bands according to perceptual criteria such that equal contribution of each sub-band to the articulation index occurs. Each band is low-pass translated (LPT) to zero frequency by a modulation process prior to encoding. The aim of LPT is to reduce the sampling rate, and LP Translated signal is now filtered with a cut-off frequency, f_n, equal to the bandwidth of the sub-band. The output of low-pass filters are sampled at $2f_n$ (f_n being different for each band) and encoded, for example, by APCM having Jayant's one-bit memory quantizer. SBC with its noise shaping ability has a speech quality that is higher than 2-bits ADPCM below 24 Kb/s and also at 9.6 Kb/s, its performance is almost the same as 19 Kb/s ADM.

Transform Coding (TC) is more complex than SBC (which is itself

more complex than ADPCM). Recently adaptive transform coding (ATC) techniques have been exhaustively investigated.[13,95,98,99] The speech signal is block processed. Each block is transformed and adaptive bit allocation and coefficient selection performed. As the discrete Fourier Transform (DFT) is often used, ATC is similar to SBC having many bands. In references (95,99) the speech signal which is sampled at 8 kHz and band-limited to 2.8 kHz encoded using 4 different types of encoder, namely, ATC, SBC, ADPCM (AQJ and first-order fixed predictor) and DPCM-AQF having block adaptive quantizer and 8^{th}-order block adaptive predictor whose parameters are updated at every 8-16 msec. The conclusions at transmission bit-rates of 9.6, 16 and 24 Kb/s are as follows:

If complexity/cost is of no concern then ATC is superior in terms of subjective quality of recovered speech. If complexity/cost is of concern, then SBC is an attractive choice, since it has better quality speech, but slightly more complex than the ADPCM having first-order predictor. Furthermore its quality is paired with that of costlier DPCM-AQF coder. ATC is not described in detail, since it is beyond the scope of this thesis. However, interested readers may consult references (95-99).

Another system to be mentioned in this section is referred to as voice-excited vocoder, (VEV). Such a vocoder attempts to gain the advantages of both waveform coder and the vocoder.[7,10] In the analyser, a low frequency region of the speech band, known as baseband, (typically

300-800 Hz) is encoded and transmitted using waveform coder and the rest of the speech band is treated as in the case of channel vocoder. An important feature of the base band signal is that, if there is any periodicity in the signal, it is inherently contained in the base band. Consequently, the difficulties due to voiced/unvoiced decision and pitch extraction can be avoided. At the receiving end, base band signal is processed by a non-linear distortion element which flattens and broadens the signal's power spectrum without disturbing the periodicity of the signal if there is any. This flattened signal is used as the excitation source in the synthesizer. The speech quality resulting from voice-excited channel vocoder is better than channel vocoder, however it requires higher transmission bit-rates typically 9.6 Kb/s.

Finally, a much simpler technique of speech coding for intermediate bit-rates, known as Time-Encoded Speech (TES)[100] is recently receiving a lot of attention. This method splits the speech waveform into intervals between successive zero-crossings and then each segment is defined by one of a pattern, and extra information about the amplitudes is also transmitted. From the reference (100), it is clear that TES, while still in its infancy, seems to require less computation than many other systems described so far aiming for the 4-16 Kb/s.

CHAPTER III

FIXED AND BLOCK ADAPTIVE PREDICTORS IN DPCM

3.1 INTRODUCTION

In the previous chapter, speech coders are briefly reviewed and their performance is compared in terms of SNR values. In this chapter, DPCM encoders employing either fixed or adaptive quantizers, together with fixed or block adaptive predictors, will be examined in detail.

The long-term SNR of DPCM is given by

$$SNR = G.Q \tag{3.1}$$

where G is the ratio of the signal power to quantizer input power and sometimes is referred to as the *"prediction gain"*. Q is the ratio of the quantizer input power to quantization noise power. Equation (3.1) may be written, in terms of dB's as

$$SNR(dB) = SNRI(dB) + SNRQ(dB) \tag{3.2}$$

where

$$SNRI(dB) = 10\log_{10}G \tag{3.2a}$$

$$SNRQ(dB) = 10\log_{10}Q. \tag{3.2b}$$

The SNR improvement term, SNRI of DPCM over PCM, is the result of the formulation of the difference signal sequence $\{e_i\}$ between the input speech samples, $\{X_i\}$, and their estimates, $\{Y_i\}$, at the output of the predictor. In particular, the high correlation that often exists between successive speech samples will ensure that the variance of $\{e_i\}$ is considerably smaller than the variance of the input speech samples, $\langle x_i^2 \rangle$, where i is the i^{th} sampling instant, i.e., $1<i<NS$ and $\langle . \rangle$ is the time average of (.) taken over the total number of samples, NS. As shown in Figure 2.14, the noise introduced in DPCM encoding is

equal to the noise produced by the quantizer, i.e., at i^{th} sampling instant, $\hat{x}_i = x_i + q_i$ and $\hat{e}_i = e_i + q_i$. Furthermore, the power of this quantization noise, $\langle q_i^2 \rangle$ is proportional to the power of the quantizer's input signal $\langle e_i^2 \rangle$, and as $\langle e_i^2 \rangle < \langle x_i^2 \rangle$, it follows that DPCM will outperform PCM when encoding speech signals.

Obviously, the magnitude of the SNRI term in Equation (3.2) will depend on the efficiency of the predictor used in the DPCM encoder. Efficient predictors can be first designed according to the long-term or short-term statistics of speech and embedded in a DPCM loop. When a predictor is operating in the feedback loop of a DPCM encoder, however, the input speech samples are predicted from previously decoded speech samples and as a result the input signal to the predictor is contaminated with quantization noise and its estimation accuracy is affected. This limitation effect in the performance of the prediction and the resulting decrease in the SNR of the encoder, is emphasized at low bit-rates where coarse quantization is used and the predictor-quantizer *"mismatching"* becomes significant. This seems to suggest that the optimization of the predictor should be performed using the statistics of the decoded speech samples, including information related to the quantization noise. In the simulation experiments presented in this chapter however, the transmission bit-rates concerned are relatively high, allowing for the predictors to be designed using the statistics of the original speech data.

The predictor, employed in a DPCM coder (encoder-decoder), can be

i) fixed, i.e., time-invariant, ii) block adaptive or iii) sequentially adaptive, and various algorithms can be used for the calculation of the prediction coefficients. McDonald,[55] in his often referred paper, studied the case where the predictor is designed with fixed coefficients. Later, Noll[38] examined block adaptive schemes where the non-stationary nature of speech is considered. In particular, values of the prediction coefficients vary with time in a block adaptive manner and the predictor can cope with changes in the statistics of speech.

The reason for the inclusion of the present chapter is two-fold:

a. to discuss the relative merits of the fixed and block adaptive predictors,
b. to prepare the background for the subsequent chapters which deal with sequentially adaptive prediction schemes.

We start with a description of performance criteria which can be used for the comparison of different predictors. Then, the design of fixed and block adaptive predictor is presented. Further, computer simulation results of DPCM codecs employing these predictors and fixed or adaptive quantizers are discussed. Finally, the tolerance of the resulting codecs in the presence of channel errors is examined.

3.2 VARIOUS CRITERIA OF SYSTEM PERFORMANCE

3.2.1 Long-Term Signal-to-Noise Ratio, SNR

The calculation of long-term SNR is simple and is extensively used throughout the thesis in evaluating the performance of various encoders. It is defined as,

$$SNR = \frac{\langle x_i^2 \rangle}{\langle (x_i - \hat{x}_i)^2 \rangle} \tag{3.3}$$

where $\langle (x_i - \hat{x}_i)^2 \rangle$ is the variance of the error samples, $(x_i - \hat{x}_i)$, and $1 \le i \le NS$, where NS is the total number of samples. In many cases, SNR is a reliable performance indicator, especially at high transmission bit-rates, where it is closely related to the subjective performance of the codec. However, this is not always true as sometimes higher SNR values are not necessarily related to improved quality speech. This inaccuracy arises from the importance of the subjective measure of the quantization noise, rather than its power as it has been discussed in Section 2.3.5.

3.2.2 Segmented SNR, SNRSEG

A different criterion to measure the performance of waveform encoders was proposed by Noll[38,48] and it is known as *"Segmented SNR"*. It is similar to the total SNR measurement of Equation (3.3), but instead of evaluating one SNR value, the speech data is divided into successive blocks of W samples and a SNR(dB) value is computed for each

block. The SNRSEG is then formed as the average of the block SNR(dB)'s i.e.,

$$\text{SNRSEG(dB)} = \frac{1}{NB} \sum_{m=1}^{NB} 10 \log_{10} \frac{\sum_{i=1}^{W} x_{i+(m-1)W}^2}{\sum_{i=1}^{W} \{x_{i+(m-1)W} - \hat{x}_{i+(m-1)W}\}^2} \qquad (3.4)$$

where NB is the total number of blocks and m corresponds to the m^{th} block.

An important feature usually added to this formula consists of discarding the SNR computation for segments whose signal power is below some threshold, typically -40,-50 dB's. This threshold, set to determine periods of silence, is introduced in order to improve the accuracy of the performance measure in the presence of idle channel noise.[68]

Many research workers are in favour of SNRSEG rather than SNR. This preference is related to a closer agreement of objective results with the subjective performance ratings of coded speech.

3.2.3 SNR Improvement Factors, SNRI, SPR

In DPCM, the gain, G, over PCM is given by

$$G = \frac{\langle x_i^2 \rangle}{\langle (x_i - y_i(x_i))^2 \rangle} \qquad (3.5)$$

or

$$\text{SNRI(dB)} = 10 \log_{10} G \qquad (3.5a)$$

where y_i is the output of the predictor operating on previous input

speech samples $x_{i-1}, x_{i-2}, \ldots$, and hence y_i is a function of $\{x_i\}$, i.e., $y_i(x_i)$. The parameter G is considered as the amount by which linear prediction can reduce the input signal power. Furthermore, as G is obtained when the predictor input is noise free, it is frequently used in simplified mathematical analysis of DPCM codecs where the interaction between quantization noise and prediction accuracy is assumed to be negligible.

In the case where y_i is computed using previous decoded samples, i.e., $y_i(\hat{x}_i)$, the prediction gain becomes more accurate and is denoted by G_a:

$$G_a = \frac{\langle x_i^2 \rangle}{\langle (x_i - y_i(\hat{x}_i))^2 \rangle} \tag{3.6}$$

or

$$SPR(dB) = 10 \log_{10} G_a \, . \tag{3.6a}$$

SPR is the true SNR improvement factor and the presence of quantization noise obviously affects its value, which differs from SNRI(dB). SNRI however, is often preferred in performance comparisons to SPR and can be interpreted as the upper bound of SNR gain of DPCM over PCM.

Sound spectrographic displays can also be used for performance evaluation, especially with codecs operating at low bit-rates where the power of the quantization noise is considerable. This is because spectrograms provide a good indication of both the tracking capacity of the codec and the relative nature of the noise spectra with respect to the input spectra.

Finally, the subjective quality of the recovered speech is the most

important criterion to be considered. Our approach, therefore, in the presentation of codecs on a comparative basis at bit-rates between 16 and 40 Kb/s, is to compute the long term SNR, SNRSEG, SNRI or SPR and then to evaluate subjectively the quality of the reproduced speech signal through a series of informal listening tests.

3.3 TIME-INVARIANT PREDICTORS WHEN USED IN DPCM SYSTEMS

The term, predictor, is referred to a device which estimates the current speech sample as a linear combination of past samples. The diagram of such a linear predictor is shown in Figure 3.1 with the predicted value y_i formed as

$$y_i = \sum_{k=1}^{N} a_k x_{i-k} , \tag{3.7}$$

where a_k is the k^{th} prediction coefficient or tap gain and N is the order of the predictor. Obviously, the impulse response of predictor is controlled by the values of the predictor coefficient's a_k, k=1,2,...,N. The methods of selecting the a_k coefficients are of interest in this thesis.

In designing *"optimum predictors"*, Wiener[101] developed Kolmogoroff's[102] earlier studies and calculated the value of the coefficients so that the mean squared error between the input and the predicted samples $\langle e_i^2 \rangle = \langle (x_i - y_i)^2 \rangle$ is a minimum. This method is often referred to as the Least-Mean-Square error optimization, LMS, procedure.

In DPCM systems, y_i is given by Equation (3.7), with x_i replaced by the decoded sample $\hat{x}_i$. Hence, the prediction error power is

$$\sigma_e^2 = \langle e_i^2 \rangle = \langle (x_i - \sum_{k=1}^{N} a_k \hat{x}_{i-k})^2 \rangle . \tag{3.8}$$

Equation (3.8) is expanded as

$$\sigma_e^2 = \langle (x_i - \sum_{k=1}^{N} a_k x_{i-k} - \sum_{k=1}^{N} a_k q_{i-k})^2 \rangle \tag{3.9}$$

since $\hat{x}_i = x_i + q_i$, and q_i is the noise sample introduced by the

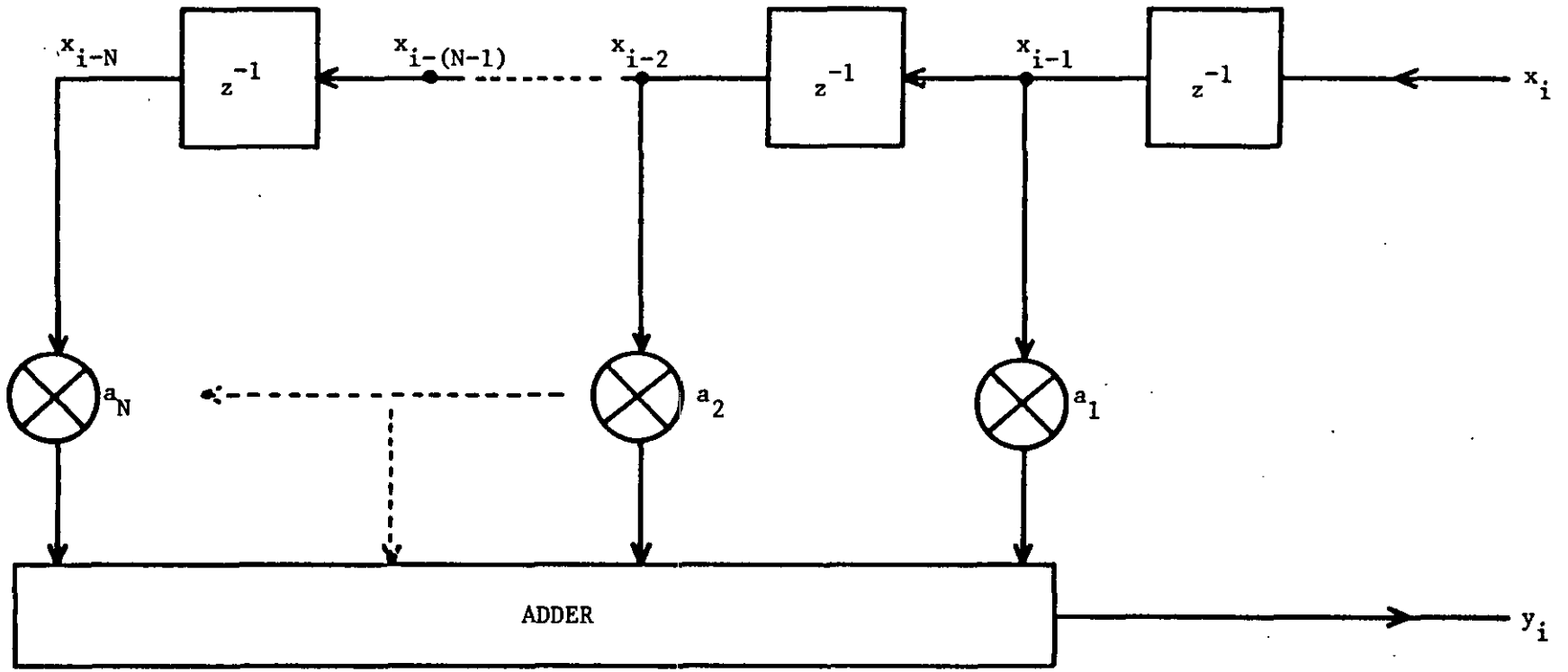


FIGURE 3.1: Linear Predictor-Transversal Filter Structure

quantization process at the i^{th} sampling instant. Assuming that the correlation between the q_i samples and also between q_i and x_i is negligible, Equation (3.9) is rewritten as

$$\sigma_e^2 \simeq \langle (x_i - \sum_{k=1}^{N} a_k x_{i-k})^2 \rangle + \langle q_i^2 \rangle \sum_{k=1}^{N} a_k^2 . \tag{3.10}$$

For a coder having a high SNR, $\langle q_i^2 \rangle$ can be neglected ($\langle q_i^2 \rangle = \sigma_q^2 << \sigma_x^2 = \langle x_i^2 \rangle$ and the number of bits, $b \geq 2$) and Equation (3.10) is approximated by

$$\sigma_e^2 \simeq \langle (x_i - \sum_{k=1}^{N} a_k x_{i-k})^2 \rangle . \tag{3.11}$$

Furthermore, Equation (3.11) may be expanded as

$$\sigma_e^2 \simeq \langle x_i^2 \rangle - 2 \sum_{k=1}^{N} a_k \langle x_{i-k} . x_i \rangle + \sum_{k=1}^{N} a_k \sum_{r=1}^{N} a_r \langle x_{i-k} . x_{i-r} \rangle \tag{3.12}$$

since the square of the sum is expressed as a double sum with two separate dummy summation indices, k,r. $\langle x_{i-k}, x_i \rangle$ is the cross-covariance between x_i and x_{i-k}.

The optimum coefficients are selected so that the partial derivative of σ_e^2 with respect to a_r becomes zero, i.e.,

$$\frac{\partial \sigma_e^2}{\partial a_r} = -2\langle x_{i-r} x_i \rangle + 2 \sum_{k=1}^{N} a_k \langle x_{i-k} x_{i-r} \rangle$$

$$= 0 \tag{3.13}$$

Hence,

$$\langle x_{i-r} x_i \rangle = \sum_{k=1}^{N} a_k \langle x_{i-k} x_{i-r} \rangle \tag{3.14}$$

where k,r=1,2,...,N.

Let $\langle x_{i-r} x_i \rangle$ be ϕ_r and $\langle x_{i-k} x_{i-r} \rangle$ be ϕ_{rk}, then Equation (3.14) is rewritten as:

$$\phi_r = \sum_{k=1}^{N} a_k \phi_{rk} \tag{3.15}$$

and from Equation (3.15), the optimum vector, $\hat{A}_{opt}$, of prediction coefficients a_k, k=1,2,...,N, is found to be

$$\hat{A}_{opt} = \Phi^{-1}\Psi \tag{3.16}$$

where Φ and Ψ are the correlation/covariance matrix and vector respectively, see Equation (2.9). The elements of matrix Φ can be defined in two different ways, resulting into what is known as *"autocorrelation"* or the *"autocovariance"* solution. In our notation, regarding the autocorrelation method, the correlation matrix will be denoted by C.

In Autocorrelation[11,20,26,28] method, assuming that the error is minimized over the finite duration, correlation is

$$C_r = \sum_{k=1}^{N} a_k C_{rk} \quad , \ 1 \leq r \leq N \tag{3.17}$$

and by substituting Equation (3.14) in Equation (3.12), the minimum error power is obtained as

$$\sigma_e^2\Big|_{min} = C_0 - \sum_{k=1}^{N} a_k C_k \tag{3.18}$$

where

$$C_0 = \langle x_i^2 \rangle$$

$$C_r = \sum_{n=-\infty}^{\infty} x_i x_{i+r} \tag{3.19}$$

C_r is known as the r^{th} autocorrelation coefficient of signal and it is an even function of r, i.e.,

$$C_r = C_{-r} \tag{3.20}$$

Now, using Equation (3.16), the Equation for the autocorrelation solution is obtained as

$$C\hat{A}_{opt} = C_0 \tag{3.21}$$

where C_0 is the autocorrelation vector.

The matrix C is symmetrical and generally referred to as a TOEPLITZ matrix, i.e., a symmetric matrix where all the diagonal elements are equal. When the elements of both the matrix C and the vector C_0 are normalized by C_0, the mean-square signal power, the following normalized autocorrelation equation is obtained,

$$\begin{bmatrix} 1 & c_1 & c_2 & \cdots & c_{N-1} \\ c_1 & 1 & c_1 & \cdots & c_{N-2} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ c_{N-1} & c_{N-2} & c_{N-3} & \cdots & 1 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_N \end{bmatrix} = \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_N \end{bmatrix} \tag{3.22}$$

where

$$c_r = \frac{C_r}{C_0} \tag{3.23}$$

In practice, the signal is defined over a finite interval, rather than having $-\infty<i<\infty$. This indicates that the samples of the signal is multiplied by a window function wb_i so that another signal x_i' is obtained that is zero outside a specified interval $1<i<W$, viz:

$$x_i' = \begin{cases} x_i\ wb_i & 1<i<W \\ 0 & \text{otherwise} \end{cases}$$

The normalized autocorrelation coefficients are then defined by

$$c_r = \frac{C_r}{C_0} = \frac{\sum_{i=1}^{W-r} x'_i x'_{i+r}}{\sum_{i=1}^{W} (x'_i)^2}, \quad r \geq 1. \tag{3.24}$$

The shape of the window function w_i can be of importance. The simplest one is the rectangular window where $wb_i=1$ for $1<i<W$. Further details of the windowing are to be discussed later in this section.

In the autocovariance method, speech analysis and synthesis problems originally applied by Atal and Hanauer,[31] the elements of the matrix Φ and the vector Ψ, see Equation (3.16) are calculated so that the error σ_e^2 is minimized over a finite interval, say $1<i<W$. Therefore, from Equations (3.12) and (3.14), we get

$$\sum_{k=1}^{N} a_k \phi_{kr} = \phi_r \quad , \quad 1 \leq r \leq N \tag{3.25}$$

and

$$\sigma_e^2|_{min} = \phi_0 - \sum_{k=1}^{N} a_k \phi_k \tag{3.26}$$

where

$$\phi_{kr} = \sum_{i=1}^{W} x_{i-k} x_{i-r} \tag{3.27}$$

is the covariance function of the signal samples $x_{i+1}, x_i, x_{i-1}, \ldots$. The matrix formed using ϕ_{kr} for $1 \leq k,r \leq N$ is known as the covariance matrix. Equation (3.27) may also be written as

$$\phi_{kr} = \sum_{i=-k}^{W-k} x_i x_{i+k-r} \tag{3.28}$$

or

$$\phi_{rk} = \sum_{i=-r}^{W-r} x_i x_{i+r-k} \tag{3.29}$$

hence

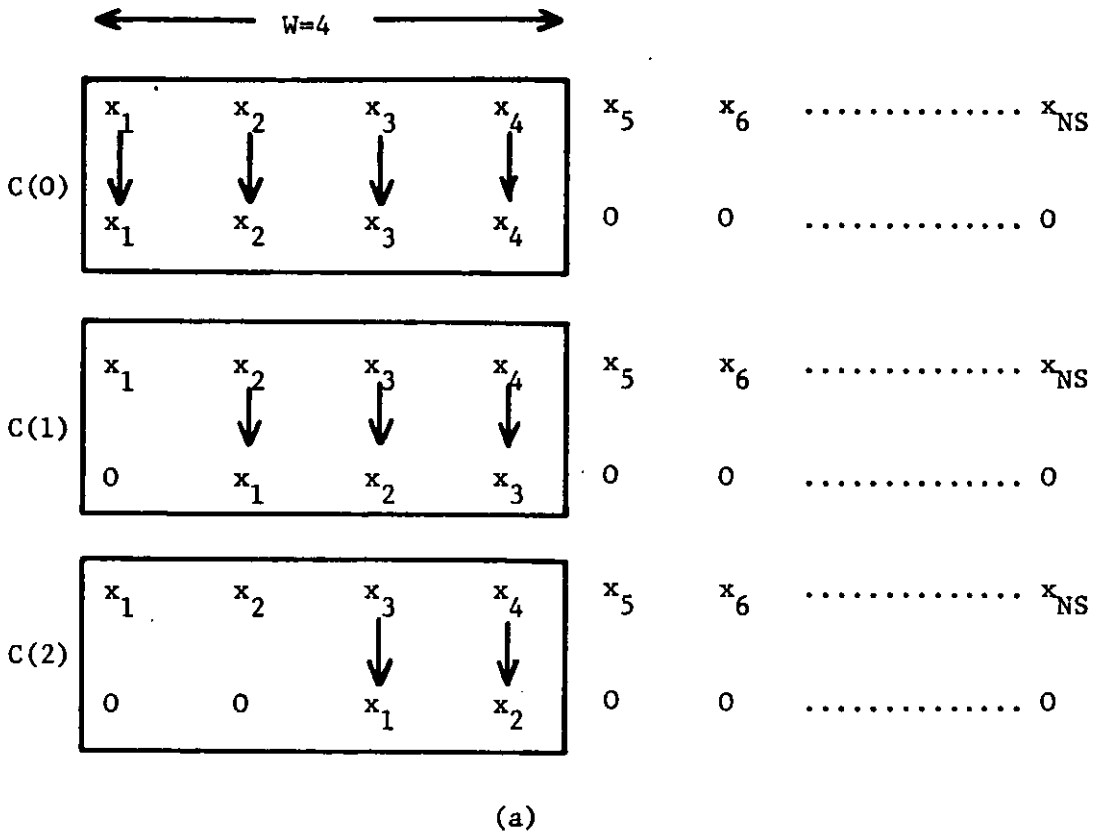
$$\phi_{rk} = \phi_{kr} \tag{3.30}$$

The importance of Equations (3.28)-(3.30) is two-fold. Firstly, the covariance matrix is a symmetric matrix with unequal diagonal elements and therefore, it is not a Toeplitz matrix. Secondly, the number of samples required for the computation of the covariance function is increased outside the interval $1<i<W$, i.e., to $N+W$, compared to the W samples used in the autocorrelation method. The covariance equation is therefore formed as

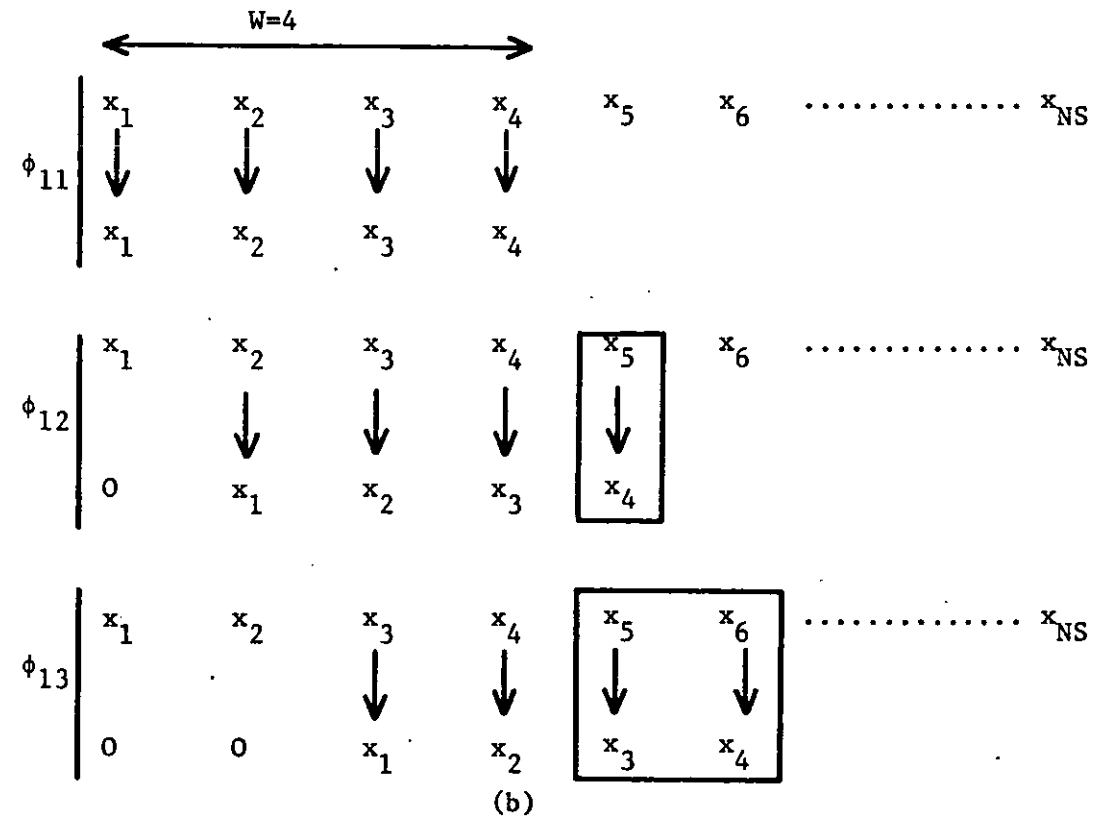
$$\begin{bmatrix} \phi_{11} & \phi_{12} & \cdots & \phi_{1N} \\ \phi_{21} & \phi_{22} & \cdots & \phi_{2N} \\ \vdots & & & \\ \phi_{N1} & \cdots & & \phi_{NN} \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_N \end{bmatrix} = \begin{bmatrix} \phi_{10} \\ \phi_{20} \\ \vdots \\ \phi_{NO} \end{bmatrix} \qquad (3.31)$$

Now, if we are to briefly compare the two methods, we note that as the time interval over which the optimization procedure is applied goes to infinity or in practice to the overall duration of a speech sound, the autocorrelation solution approaches the solution provided by the covariance method, see Figure 3.2. However, in many applications, the error power is minimized over a short segment of the speech waveform, rather than over the total number of samples in an utterance. Thus, the question arises as to whether to use the autocorrelation or the autocovariance method for modelling the vocal tract and the subsequent use of linear predictors in vocoders and DPCM systems.

In the autocorrelation method, the first N samples are predicted from the speech samples outside the segment of speech. Since these samples are zero, a large error may occur. The windowing process is introduced to reduce the error so that it smoothly tapers the signal



(a)



(b)

FIGURE 3.2: Samples Used in the Calculation of:
(a) Autocorrelation,
(b) Covariance Techniques,
where NS is the total number of samples

to zero at the window ends. The choice of window depends on the type of signal to be analysed. If the signal is considered to be stationary for a long period of time, then a rectangular window is adequate. However, speech-like signals can be considered stationary for a duration of only a few pitch periods and a Hamming or a Hanning window is more appropriate. The Hamming or Hanning[103] functions are defined as follows:

$$wb_i = \begin{cases} 0.54-0.46\ \cos(\frac{2\pi i}{W-1}) & 0 \leq i \leq W-1 \\ 0 & \text{otherwise} \end{cases} \tag{3.32}$$

$$wb_i = \begin{cases} \frac{1}{2}\ [1-\cos(\frac{2\pi i}{W-1})] & 0 \leq i \leq W-1 \\ 0 & \text{otherwise} \end{cases} \tag{3.33}$$

The process of multiplication of a signal by a window is equivalent to the convolution of the frequency response of the window with the speech spectrum. This results in smearing effects in the speech spectrum and the degree of smearing obviously depends on both the size and the type of the window. For example, when input speech signals having a duration of a few tens of milliseconds, are multiplied by a rectangular window or a Hamming window, the spectrum of Hamming-windowed data shows a substantial reduction of spectral distortion.

On the other hand, the autocovariance method does not require the windowing process and parameters such as the bandwidth of the formants can be estimated more accurately than with the autocorrelation method.

When the complexity for a given number of prediction coefficients, N is to be considered, the autocovariance methods requires N^3 operations (multiplications, divisions) while the autocorrelation technique needs

N^2 operations and it is therefore considerably faster solution. However, when both methods are to be carefully compared when utilized in practical applications the above mentioned computational efficiency becomes less important.[20] This is because,

a. the time required to compute the matrix of autocorrelation/ covariance coefficients is greater than the time to solve the matrix relationships given by Equations (3.16), (3.21), (3.31).

b. the time interval, W/f_s, required for both methods is not the same. For the correlation method, this is typically 10-30 msecs, whereas covariance method can be used with 2-3 msecs of time intervals.[58]

Another difference between the two methods arises when the roots of the predictor polynomial which are the poles of predictor and provide the properties of vocal tract, are considered. For stability reasons, the roots must lie inside the unit circle. The covariance method does not guarantee stability[31] whereas the autocorrelation provides always a stable solution.[26]

To summarize all these remarks, we can conclude that a predictor designed using the autocovariance method models more accurately the vocal-tract characteristics compared to the case where the autocorrelation method is employed. The difference between these algorithms, in terms of both SNR and subjective speech quality, is however negligible, when used in DPCM. Therefore, due to its stability, the autocorrelation method was selected to be used in our experiments.

3.3.1 First-Order DPCM, DPCM(1,b)

The order of a DPCM is defined by the order of the predictor. Thus, a first-order DPCM employs a predictor having only one delay element and one prediction coefficient, as illustrated in Figure 3.3. The importance of the first order DPCM (1,b) lies in its simplicity and this encoder is often used as a reference in system comparisons. The SNR gain of DPCM(1,b) over PCM can be calculated from Equation (3.12). That is

$$\sigma_e^2 \approx \langle x_i^2 \rangle \left[1 - 2 \sum_{k=1}^{N} a_k \frac{\langle x_{i-k} x_i \rangle}{\langle x_i^2 \rangle} + \sum_{k=1}^{N} a_k \sum_{r=1}^{N} a_r \frac{\langle x_{i-k} x_{i-r} \rangle}{\langle x_i^2 \rangle} \right] \quad (3.34)$$

For a first order predictor, the above expression is written as

$$\sigma_e^2 \approx \sigma_x^2 (1 - 2a_1 c_1 + a_1^2) \ . \quad (3.35)$$

In order to achieve the $\sigma_e^2 << \sigma_x^2$ condition, the term between the brackets must be a minimum, therefore,

$$1 - 2a_1 c_1 + a_1^2 \Big|_{\min} = \frac{\partial}{\partial a_1}(1 - 2a_1 c_1 + a_1^2)$$

$$0 = -2c_1 + 2a_1 \ . \quad (3.36)$$

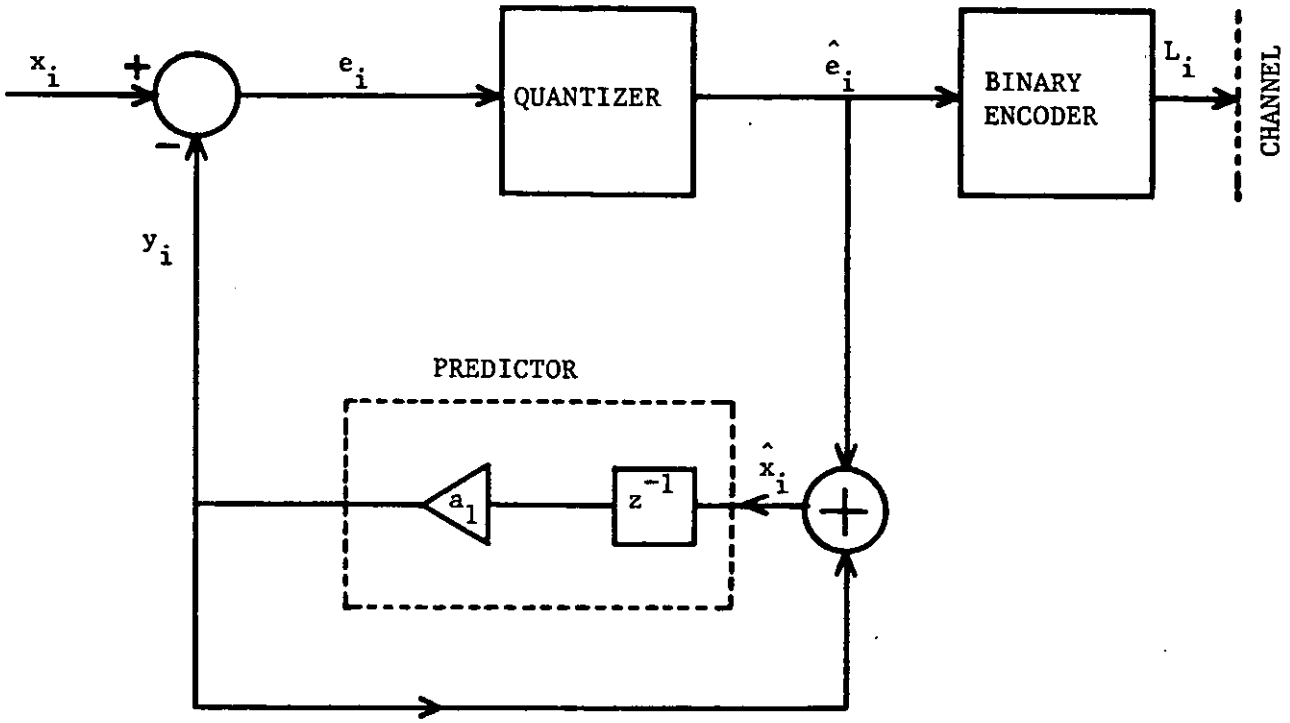
and

$$a_1 = c_1 \ . \quad (3.37)$$

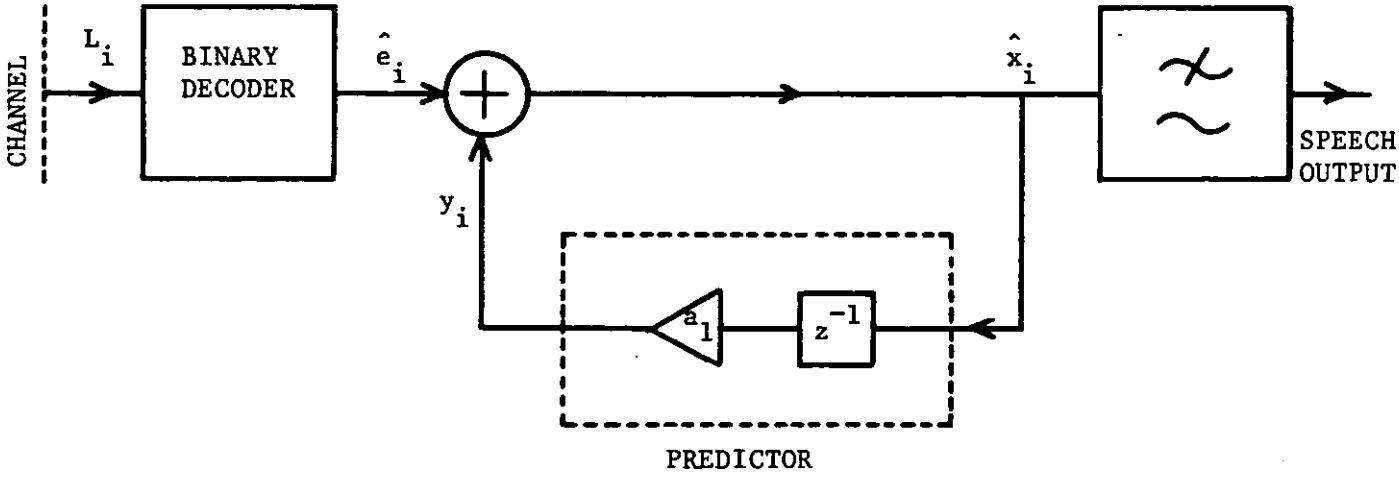
Using Equation (3.35)

$$\frac{\sigma_x^2}{\sigma_e^2} = G = \frac{1}{1 - c_1^2} \ . \quad (3.38)$$

As $c_1 < 1$, G is always greater than 1. Equation (3.38) indicates the superiority of DPCM(1,b) over PCM.



a. DPCM(1,b) Encoder



b. DPCM(1,b) Decoder

FIGURE 3.3: First-Order DPCM Encoder-Decoder

In the case where an ideal integrator is used in the feedback loop of the DPCM codec, i.e., $a_1=1$, the SNR advantage, G, obtained from Equation (3.35) is equal to

$$G = \frac{1}{2(1-c_1)} \tag{3.39}$$

and provided that $c_1 \geq 0.5$, a condition which is true for most of the time with speech signals, G>1. Hence, the simplest DPCM(1,b) system having $a_1=1$ shows a long term SNR gain over PCM.

In practice, DPCM(1,b) with $a_1=c_1$ is always preferred to DPCM(1,b) having an ideal integrator in the feedback loop as the former is more robust to channel errors. This is because the leaky constant, c_1, attempts to reduce any error accumulation effects in the receiver.

3.3.2 N^{th} Order DPCM, DPCM(N,b)

When an N^{th} order predictor is used in the feedback loop of a DPCM encoder, Equation (3.12) may be written in matrix notation as

$$\sigma_e^2 \simeq \sigma_x^2(1 - 2\hat{\mathbf{A}}^T\mathbf{C}_0 + \hat{\mathbf{A}}^T\mathbf{C}\hat{\mathbf{A}}) \tag{3.40}$$

where $\mathbf{C}_0$ and $\mathbf{C}$ are the normalized autocorrelation vector and matrix, respectively. The value of σ_e^2 is a minimum when $\hat{\mathbf{A}}_{opt}=\mathbf{C}^{-1}\mathbf{C}_0$ (see Equation 3.21) and Equation (3.40) can be written as

$$\sigma_e^2\Big|_{min} \simeq \sigma_x^2(1 - \hat{\mathbf{A}}_{opt}\mathbf{C}\hat{\mathbf{A}}_{opt}) \tag{3.41}$$

then

$$G \simeq \frac{1}{1 - \hat{\mathbf{A}}_{opt}.\mathbf{C}.\hat{\mathbf{A}}_{opt}} \tag{3.42}$$

and using $\hat{A}_{opt}=C^{-1}C_0$ the denominator of Equation (3.42) may be written as $(1-\hat{A}_{opt}C_0)$, then,

$$G \simeq \frac{1}{1 - \sum_{k=1}^{N} a_k c_k} \tag{3.43}$$

and

$$SNRI = 10 \log_{10} G \tag{3.43a}$$

An Appendix C presents an interesting result relating to the second order fixed predictors.

In deriving Equation (3.43), it is assumed that the quantization noise is negligible and x_i and q_i are orthogonal, i.e., $\langle x_i q_{i+t} \rangle = 0$ for all t.

In references (11,60) the authors however considered the effect of the quantization noise in the performance of the DPCM(N,b) predictor, i.e., $\hat{x}_i = x_i + q_i$ and formulated a more accurate expression for the gain G_a, over PCM;

$$G_a = \frac{1}{1 - \sum_{k=1}^{N} a_k c_k \left(\frac{SNR}{1+SNR}\right)} \tag{3.44}$$

and SPR(dB) is given by Equation (3.6a).

The value of G_a approaches that of G in Equation (3.43) for high values of SNR where the quantization noise is negligible $(\sigma_q^2 << \sigma_x^2)$.

In all above formulations, the quantizer SNR, Q, is assumed to be constant (see Equation (3.2)). Many authors searched for more realistic formulations of SNR under the assumption that the speech input and quantization noise are orthogonal. In his DPCM(1,b) analysis, O'Neal,[56]

considered the variations in Q and developed an improved SNR expression for N=1, $a_1=c_1$, see Equation (2.41). For N-order predictor, this expression is

$$SNR = Q \cdot \frac{1 - (\sum_{k=1}^{N} a_k^2)/Q}{1 - \sum_{k=1}^{N} a_k c_k} \tag{3.45}$$

i.e., for $a_1=c_1$ Equation (3.45) reduces to Equation (2.41). Later, Noll[104] generalized the above expression by taking the normalized autocorrelation matrix of the noise samples into account, viz:

$$SNR = Q \cdot \frac{1 - \hat{A}_{opt}^T C_q \hat{A}_{opt}/Q}{1 - \sum_{k=1}^{N} a_k c_k} \tag{3.46}$$

It is easily observed that when $C_q=I$ (for $\langle x_i q_{i+t} \rangle=0$, for all t, the quantization noise becomes white), SNR, in Equations (3.46) and (3.45) becomes the same.

In an attempt to define the effect of the quantizer-predictor interaction in the encoding performance of DPCM, Gibson[105] showed that the optimization of the predictor by maximizing the SNRI can produce a change in long-term SNR that is much greater or less than the increase in SNRI. Gibson attributed this cause to the change in the p.d.f. of the signal at the input of the quantizer and argued that any change in p.d.f. at the quantizer input can produce a significant improvement or reduction in the quantization noise power. Therefore, the SNR of DPCM systems must be calculated in such a way that any variation in quantization input p.d.f. is taken into account.

3.4 BLOCK ADAPTIVE PREDICTORS WHEN USED IN DPCM SYSTEMS

As discussed in Section 2.3.4.A, one method which improves the encoder SNR performance is to compute the prediction coefficients at every W/f_s seconds, where W is the number of samples to be used by the autocorrelation/autocovariance analysis. In this way, the prediction coefficients are designed to match the short-term characteristic of the input signal and the predictor is called *"Block Adaptive Predictor"*, BAP.

There are basically two techniques for updating the predictor's coefficients. In the first one, known as *"Forward Block Adaptive Prediction"*, FBAP, the coefficients are computed from the original input data which is delayed by W/f_s seconds. Figure 3.4 shows the block diagram of a FBADPCM(N,b) encoder, where N is the number of prediction coefficients. The prediction coefficients are defined to minimize the block error power, σ_b^2, given by

$$\sigma_b^2 = \sum_{i=1}^{W} (x_i - \sum_{k=1}^{N} a_k x_{i-k})^2 \qquad (3.47)$$

In our experiments, the autocorrelation method is applied to W input samples and the resulting a_k coefficients are employed for the encoding of these samples. Notice from Equation (3.47) that the input speech samples are used to calculate the a_k coefficients and therefore, the prediction coefficients must be encoded and transmitted (forward transmission) together with the speech information at the output of the receiver. Obviously, with the predictor being able to respond to the short-term changes in the statistics of speech, the performance of DPCM is improved compared to that where a fixed predictor is employed.

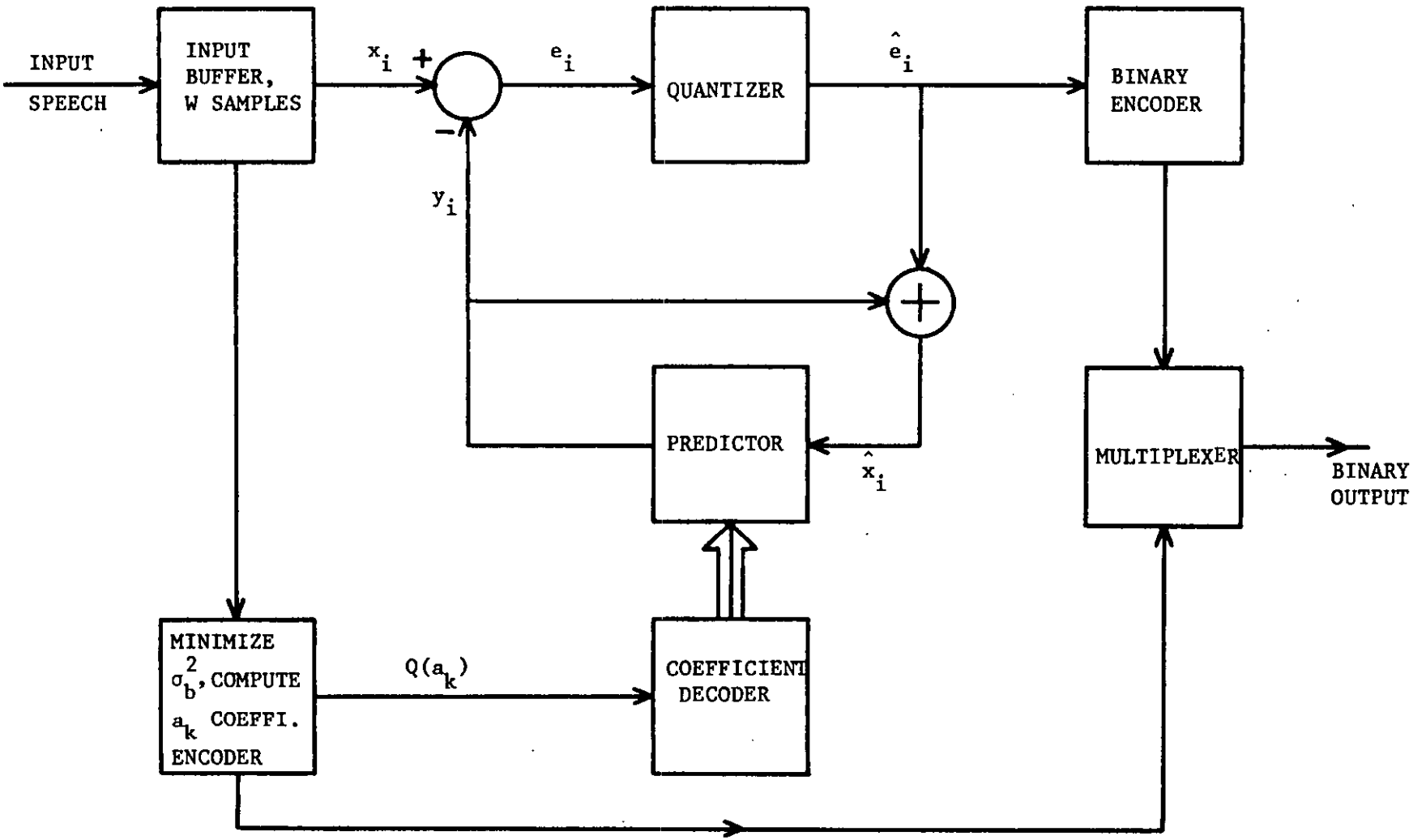


FIGURE 3.4: FBADPCM(N,b) Encoder

The transmission of the coefficient values to the receiving end, however, in addition to the quantized samples $\{\hat{e}_i\}$, increases the channel capacity. It is to say that for the available bit-rate, the number of bits for sending the error signal $\hat{e}_i$ is reduced since the coefficient transmission occupies some portion of the channel capacity. This increase in transmission bit-rate can be considered as an equivalent loss in the SNR performance of the encoder and is given by,[48]

$$\text{LOSS IN SNR} = 6.02 \frac{b_c}{W} \text{ dB} \tag{3.48}$$

where b_c is the number of bits per coefficient per block and W is the number of samples per block. The loss in SNR, however, resulting from the transmission of the predictor coefficient to the receiver is considerably smaller compared to the SNR gains obtained by the use of the Forward adaptive predictor in the DPCM system.

The second method, known as the *"Backward Adaptation"*, BA scheme, computes the prediction coefficients from the previously decoded speech samples. Thus the transmission of the prediction coefficients as side information is eliminated. Figure 3.5 shows the schematic block diagram of BA scheme. The autocorrelation method can be employed on the locally decoded samples in a similar way to that of FBA and the coefficients are selected to minimize

$$\sigma_b^2 = \sum_{i=1}^{W} (\hat{x}_i - \sum_{k=1}^{N} a_k \hat{x}_{i-k})^2 . \tag{3.49}$$

Although the $\hat{x}_i$ samples are available at both the transmitter and receiver, the employment of the autocorrelation method in a backward

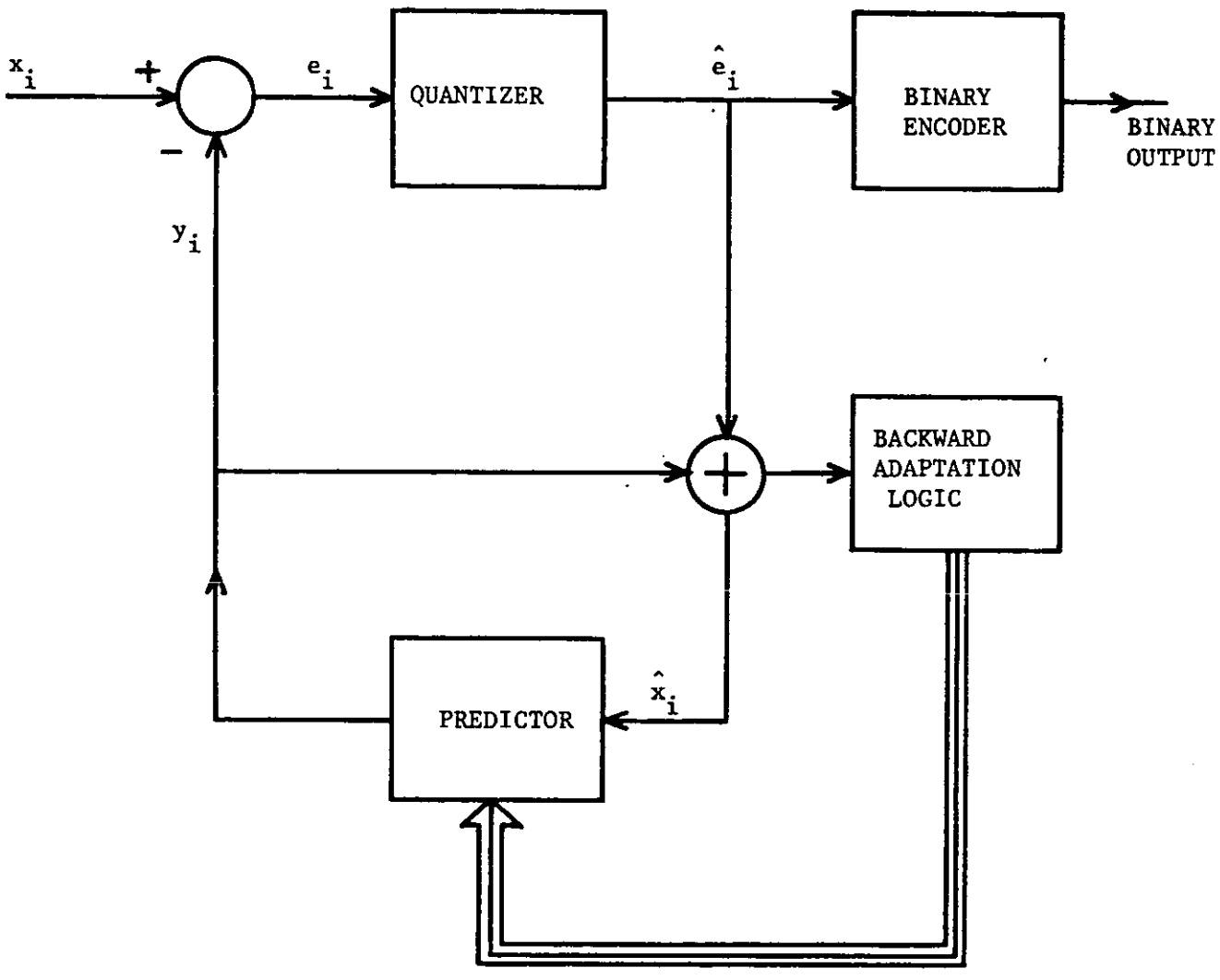


FIGURE 3.5: Generalised Backward Adaptation Encoder

adaptive prediction scheme is not suitable for practical purposes. This is because, it requires the delay of one block of decoded speech samples and therefore the values of the coefficients calculated from $\{\hat{x}_i\}$ are used in the encoding of the next incoming block of speech samples. Recently,[60] backward schemes which update their coefficients in a sequential manner every sampling instant, gained a lot of attention and they are referred to as *"Sequentially Adaptive Backward Schemes"*.

The comparison of Figures 3.4 and 3.5 indicates that forward adaptation has a more complex encoder than backward adaptation scheme but the latter necessitates a more complex decoder.

3.5 QUANTIZER SELECTION

The selection of a quantizer to be used in a DPCM encoder is an important issue, especially at low bit-rates where the effect of the increased quantization noise on the prediction algorithm is prominent. In this chapter, we will examine both fixed quantizer, FQ, and Jayant's Adaptive Quantizer, AQJ, for bit-rates between 16 and 40 Kb/s.

In FQ's, described in detail in Section 2.3.2, the quantization thresholds are defined in accordance with the p.d.f. of their input signal, i.e., $\{e_i\}$. The level allocation and step-size are unchanged during the encoding process and consequently the fixed quantizers are often referred to as time-invariant quantizers. As the bit-rate decreases (<16 kbits/sec.), the small number of fixed quantization levels in FQ's is not able to successfully quantize the input signal and adaptive quantizers (AQ) are used due to their large dynamic range and their SNR superiority over fixed quantization.

An AQJ, as described in Chapter II is basically a fixed, unit range quantizer with its input weighted by a factor that depends on the quantizer level occupied in the previous sampling instant. This is equivalent to a scheme where the stepsize of the quantizer is updated every sampling instant. Representing the magnitude of the sample transmitted in the previous sampling instant by $|L_{i-1}|$, the stepsize at i^{th} instant is given by,[36]

$$\Delta_i = \Delta_{i-1} M(|L_{i-1}|) \tag{3.50}$$

where M(.) is a time-invariant multiplier whose value depends on the

quantizer level occupied in the previous sampling instant. Notice that the Equation (3.50) is a special form of Equation (2.63a) having $\beta_q=1$.

To visualize better the system, the characteristic of a 3 bit quantizer is illustrated in Figure 3.6 where 8 code words and 4 multipliers are assigned to the output levels of the quantizer. The output sample, $\hat{e}_i$, corresponding to the quantizer input sample, e_i, is given by,

$$\hat{e}_i = \mp \frac{\Delta_i}{2} |L_i| \tag{3.51}$$

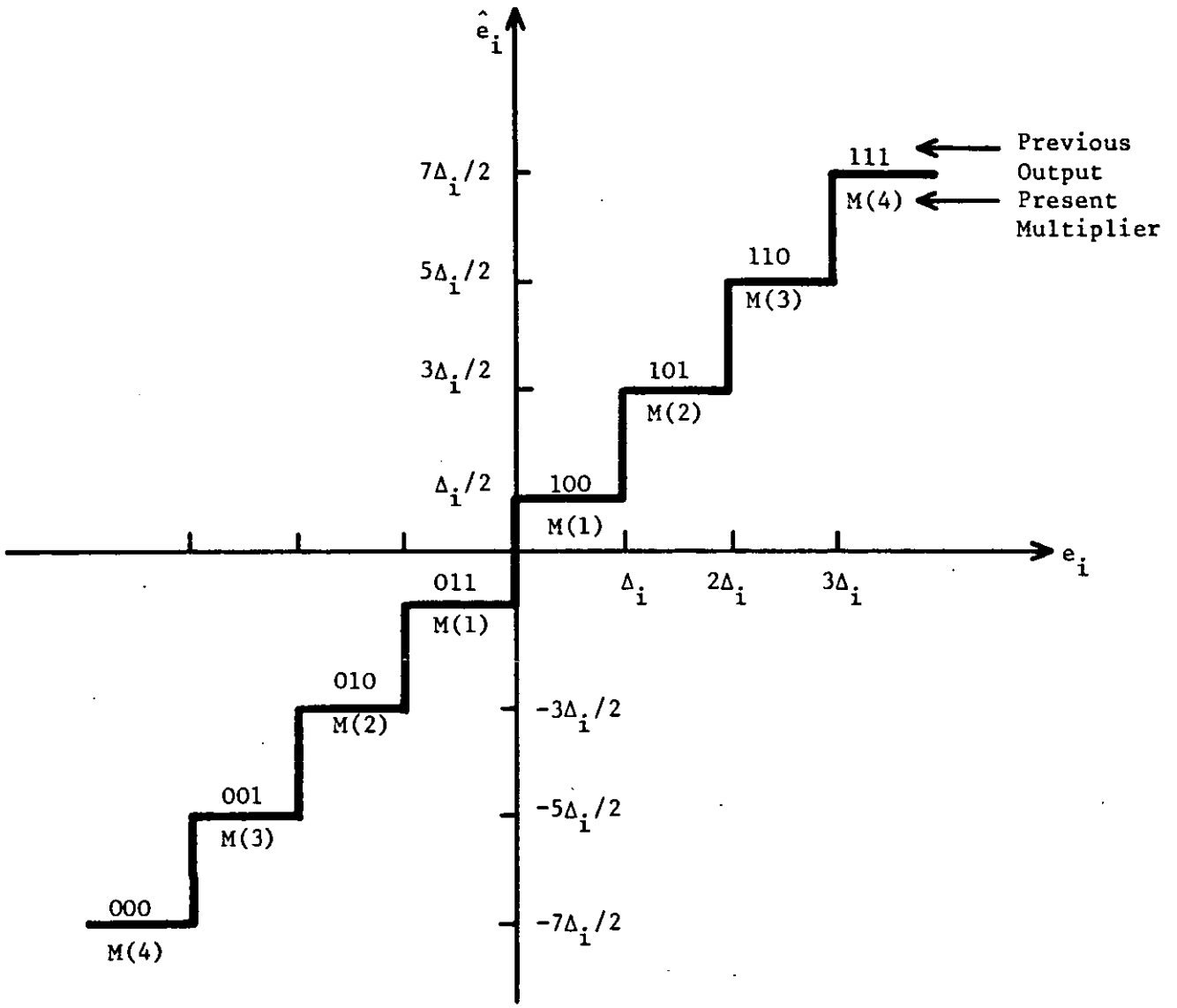
with L_i being $|L_i|=1,3,5,\ldots,2^b-1$, for an "b" bits quantizer.

In the simulation of the quantizer, the ratio of maximum step-size Δ_{max} to minimum step-size Δ_{min} is selected as 128 so that from Equation (2.63c), a dynamic range of approximately 42 dB is maintained.

Typical step-size multipliers[36] used in DPCM-AQJ encoders operating at 8 kHz sampled speech signals are tabulated in Table 3.1.

In our computer simulation experiments only uniform quantization thresholds were considered.

Finally, in an attempt to distinguish various DPCM structures, different abbreviations are assigned to each coder as shown in Table 3.2 and will be used throughout the thesis. For example, DPCM(3,4) means 3rd order, fixed predictor and 4 bits fixed quantizer, where DPCM with no brackets refers to a general DPCM system.



Code-Word	$\lvert L_{i-1} \rvert$
011-100	1
010-101	3
001-110	5
000-111	7

FIGURE 3.6: 3-Bits-AQJ

STEP-SIZE MULTIPLIERS, M	QUANTIZER OUTPUT	b=2	b=3	b=4
M(1)	$\Delta_i/2$	0.8	0.8	0.8
M(2)	$3\Delta_i/2$	1.6	0.8	0.8
M(3)	$5\Delta_i/2$		1.2	0.8
M(4)	$7\Delta_i/2$		2.0	0.8
M(5)	$9\Delta_i/2$			1.2
M(6)	$11\Delta_i/2$			1.6
M(7)	$13\Delta_i/2$			2.0
M(8)	$15\Delta_i/2$			2.4

TABLE 3.1: Step-size Multipliers

DESCRIPTION OF DPCM CODER	ABBREVIATIONS N=Order of Pre., b=bits/sample
Fixed Predictor, Fixed Quantizer	DPCM(N,b)
Sequential Predictor, Fixed Quantizer	ADPCM(N,b)
Block Adaptive Predictor, Fixed Quantizer	FBADPCM(N,b)
Fixed Predictor, Adaptive Jayant's Quantizer	DPCM(N,b)-AQJ
Sequential Predictor, AQJ	ADPCM(N,b)-AQJ
Block Adaptive Predictor, AQJ	FBADPCM(N,b)-AQJ

TABLE 3.2: Abbreviations for Various Codecs

3.6 SIMULATION RESULTS AND DISCUSSION

In the simulation experiments described in this chapter the long-term SNR(dB) was selected as the performance criterion. Furthermore, in determining the long-term SNR, the quantization noise power was calculated by filtering the difference between the original speech samples and their decoded values, so that the quantization noise components lying outside the input speech band of 0-3.4 kHz were eliminated. The filter used for this purpose is 8^{th}-order Butterworth digital low pass filter whose design and a computer program subroutine is given in Appendix D. The programming language used is the extended FORTRAN IV (FORTRAN 1900). All the computer facilities, including a collection of special algorithms for solving numerical problems, such as matrix inversions, were provided by Loughborough University of Technology Computer Centre.

Figure 3.7 shows the schematic flow-chart of the procedure employed in the DPCM computer simulations. For DPCM(N,b) and DPCM(N,b)-AQJ purposes, we start with the calculation of long term autocorrelation function of the speech samples according to

$$c_r = \frac{\sum_{i=1}^{NS} x_i x_{i+r}}{\sum_{i=1}^{NS} x_i^2} \qquad (3.52)$$

where c_r is the r^{th} autocorrelation coefficient normalized by the signal power, C_0 and c_r is obtained by shifting the speech samples by r samples. Then, c_r values are employed in forming the autocorrelation

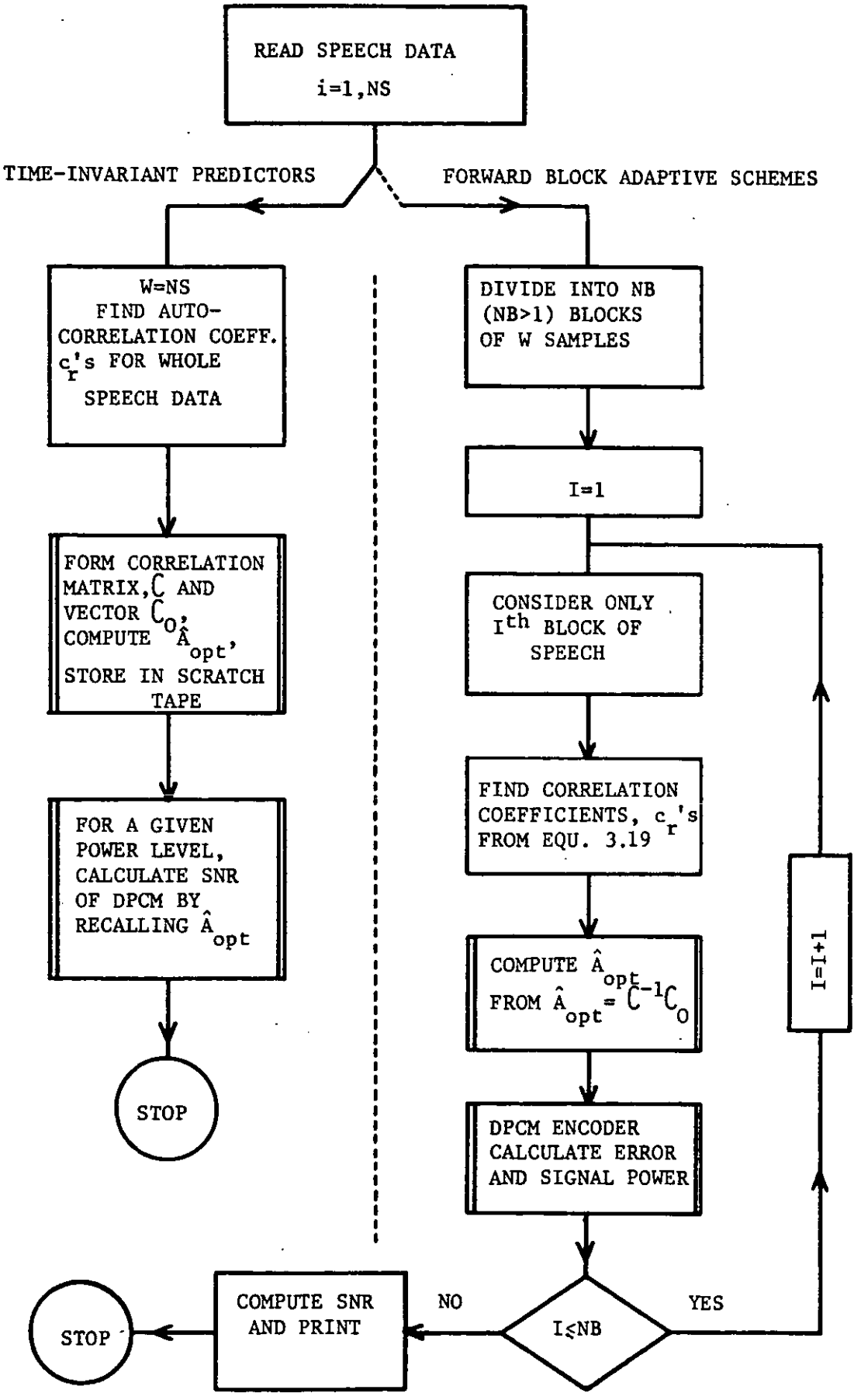


FIGURE 3.7: A Schematic Flow-Chart for Simulation Purposes

matrix C (see Equation (3.22)). The optimum predictor coefficients are computed in accordance with Equation (3.21). The matrix inversion process was achieved by using special Nottingham Algorithm Group, NAG Library.[106] These coefficients are then stored in a scratch tape. Finally, a subroutine called DPCMENCODER recalls the optimum set of vector, $\hat{A}_{opt}$ and uses them to digitize the input speech samples with a predetermined number of bits per sample. SNR(dB) values for various power levels of speech are then calculated and when Jayant's quantizer is used the operating point is selected at a power level which is half-way through the dynamic range of the encoder. For a fixed quantizer however only peak SNR values are considered.

The right-hand branch of Figure 3.7 illustrates a similar procedure for FBADPCM(N,b)-AQJ. This time however,the autocorrelation solution is applied every W samples and the encoder processes a total of NB blocks of W speech samples.

3.6.1 Input Speech Data

In order to facilitate performance comparisons, we allowed the DPCM systems to operate on the same sentence, *"An apple a day keeps the doctor away"*, which was spoken by a male. This sentence, low pass filtered to 3.4 kHz (3 dB), sampled 10,000 times/sec., was provided on a digital tape by the Joint Speech Research Unit, JSRU.

3.6.2 Upper-limits of SNR Improvement, SNRI, for DPCM

The long-term autocorrelation coefficients has been measured for the input speech signal. Figure 3.8, curve (a) shows the first 18 time-lags of the normalized autocorrelation coefficients, c_r. In addition, the corresponding values obtained using McDonald's[55] average speech data, sampled at 9.6 kHz, are also indicated in Figure 3.8, curve (b). As can be seen from Equation (3.43), the knowledge of the autocorrelation function alone is sufficient for evaluation of the upper bound of the SNR improvement in non-adaptive DPCM systems, since the optimum vector, $\hat{A}_{opt}=(a_1,a_2,\ldots,a_k)$ is also computed from c_r, r=1,2,...,k.

In Figure 3.9, curve (a) illustrates the SNRI(dB) as a function of the predictor order N, when the c_r values of Figure 3.8, curve (a) were used, while the second curve (b) is related to the average speech data. Clearly, the higher correlation between the speech samples of the sentence *"An apple a day keeps the doctor away"*, compared to the average data, manifests itself as an increase in SNRI(dB) values, for a given predictor order, N. Also it is noticeable that in both curves (a,b) of Figure 3.9, most of the gain is achieved when the order of predictor increases to 2. For higher-order predictors (N>2), the SNRI values reach a saturation level. In our speech data, the second-order predictor, N=2, provides 15.2 dB improvement where 9.78 dB of this amount arises from the first-order predictor. This suggests that most of the SNR improvement of DPCM over PCM when using fixed predictors, can be provided by a second-order predictor. It should be emphasized however, that the SNRI of 15-16 dB's, obtained by optimizing the

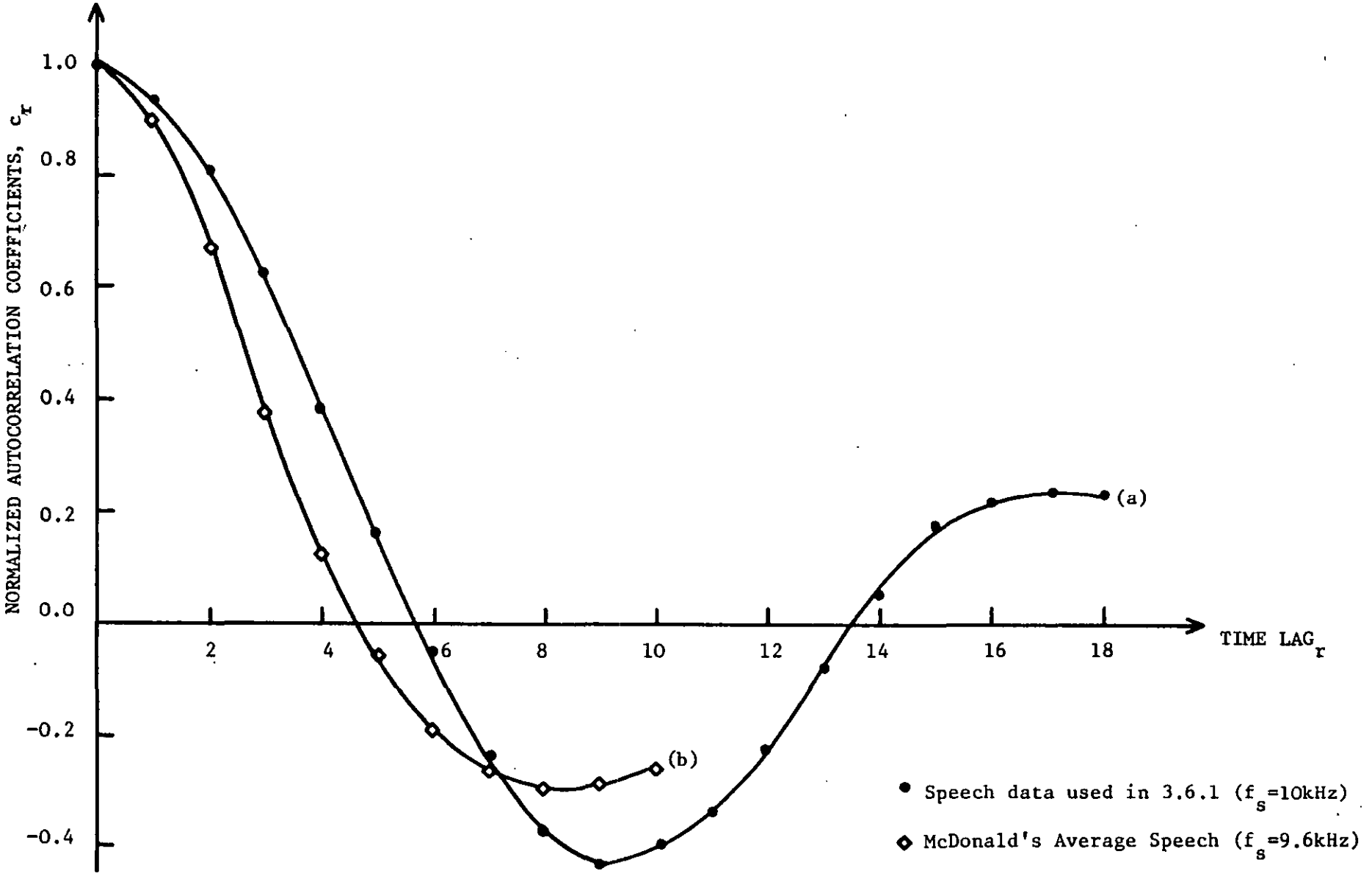


FIGURE 3.8: Normalized Autocorrelation Function for Two Speech Data

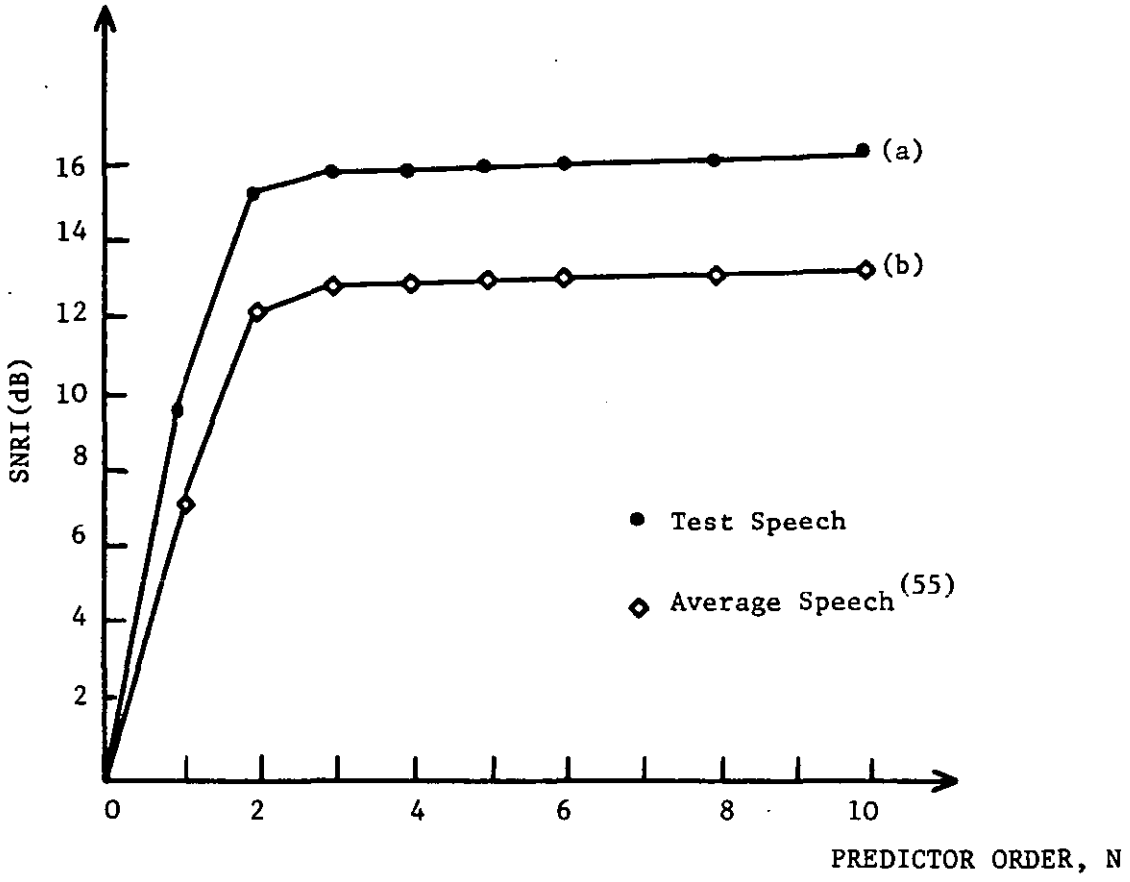


FIGURE 3.9: SNRI Versus Predictor Order, N

predictor on a particular speech sentence, can not be attained when the DPCM is handling speech samples of different talkers. This is attributed to the variations in the values of the autocorrelation function for different speakers. That is why, in practice, where a sub-optimum or average predictor is used, designed to cover a wide range of speakers and speech material, the SNRI values are significantly lower.

3.6.3 Performance of Fixed-Predictors in DPCM(N,b), DPCM(N,b)-AQJ

When first and second order predictors designed using the input speech statistics, were embedded in the feedback loop of a 3 bits per sample DPCM, DPCM(N,3), the computer simulation SNR of the codecs were found to be 18.93 and 22.29 dB's respectively. In addition at 4 bits/sample, the SNR values were calculated to be 23.81 and 28.73 dB's for N=1 and N=2 respectively. These observations clearly indicate that the performances of DPCM(1,3) and DPCM(1,4) are inferior to those obtained from DPCM(2,3) and DPCM(2,4). In terms of SNR gains in dB, the latter systems out-performs the first-order codecs by 4.36 and 4.96 dB's. These gains are comparable with those SNRI values of Figure 3.9, i.e., here, by changing N from 1 to 2, SNRI improves by 5.4 dB. However, as we reduce the bit-rate, we also see that the upper-bound improvement factor, 15.2 dB does not show itself in SNR measurement. In this case, rather than SNRI given by Equation (3.43.a), the $SPR_a(dB)$ [60] becomes a more accurate representation of the prediction gain, viz.

$$SPR_a(dB) = 10 \log_{10} \frac{1}{[1 - \sum_{k=1}^{N} a_k c_k + \sum_{k=1}^{N} a_k^2/SNR]} \qquad (3.53)$$

where SNR is the signal-to-noise ratio of the DPCM codec.

Table 3.3 shows both the theoretical results of $SPR_a(dB)$ computed from Equation (3.53) and the simulation results calculated from the ratio of $\langle x_i^2 \rangle / \langle e_i^2 \rangle$. The transmission bit-rate varies between 15.84 Kb/s (3 level fixed quantizer) and 40 Kb/s, while the predictor is of order 2.

The results in Table 3.3 indicate that as codec SNR increases for higher bit-rates, the value of $SPR_a(dB)$ approaches that of SNRI(dB) which is defined in Equation (3.43a).

Table 3.4 presents the SNR performances of various codecs, such as, PCM, APCM, DPCM(N,b) and DPCM(N,b)-AQJ, operating at 30 and 40 Kb/s. Notice that the SNR of the fixed quantizer, FQ, at 40 Kb/s, is 15.8 dB, while from Table 3.3, and at the same bit-rate, SNRQ is only 13.71 dB. The latter is computed as the power ratio of the input signal of the quantizer to the quantization noise, when the quantizer is inside the DPCM loop and thus the 2 dB discrepancy from the different nature of the input signals to the PCM and DPCM quantizers. That is, the input speech signal to the quantizer of PCM is replaced by the much wider spectrum prediction error signal of DPCM.

TRANSMISSION BIT-RATE Kb/s	SNR(dB) OF DPCM(2,b)	SPR$_o$ (dB)		SNRQ(dB)
		SIMULATION	THEORETICAL	
15.84	14.97	8.72	8.70	6.25
20	18.12	11.33	10.57	6.79
30	22.29	13.24	12.98	9.05
40	28.73	15.02	15.13	13.71

TABLE 3.3: Relative Accuracy of Equation (3.53)

TYPE OF CODEC	3 bits/sample			4 bits/sample		
	SNR(dB)	GAIN OVER PCM(DB)	GAIN OVER APCM(dB)	SNR(dB)	GAIN OVER PCM(dB)	GAIN OVER APCM(dB)
PCM	10.6	-	-	15.8	-	-
DPCM(1,b)	18.93	8.33	-	23.87	8.07	-
DPCM(2,b)	22.29	11.69	-	28.73	12.93	-
APCM	15.42	4.82	-	21.81	6.01	-
DPCM(1,b) -AQJ	27.87	12.27	7.45	28.20	12.40	6.39
DPCM(2,b) -AQJ	26.24	15.64	9.82	31.05	15.25	9.24

TABLE 3.4: Relative Merits of Various Codecs

Figures 3.10 and 3.11 show the SNR(dB) variations of DPCM codecs employing fixed (FQ) and Jayant's adaptive (AQJ) quantizers, respectively. In Figure 3.10, we observe that when the order of the predictor in DPCM(N,b) is changed from 2 to 4, the SNR(dB), at 30 and 40 Kb/s is 16.48 and 21.4 dBs. Also we notice that the performance of the 4^{th}-order DPCM is inferior to both the first and second order DPCM systems. This is in contrast to what was obtained in the SNRI, curve (a) of Figure 3.9 which indicates that the SNRI remains almost unchanged as the order of predictor increases from 2 onwards. If this SNRI of 15.8 dB is to be achieved by the DPCM(4,4) codec, we would expect to obtain a SNR value of about 31.6 dBs, since SNRQ from Table 3.4 is 15.8 dBs. However, if we utilize the SPR_a relationship given by Equation (3.53), we notice that the effect of the upper-bound improvement factor, SNRI in SNR values of the codec is significantly reduced. Table 3.5 shows both the theoretical and simulation results of SPR_a at 30, 40 Kb/s when predictor of order is 4.

N=4

DPCM(4,b)	UPPER-BOUND SNRI,Equ.(3.43)	SPR(dB) Equ.(3.44)	SPR_a(dB) Equ.(3.53)	SPR_a(dB) SIMULATION	CODEC SNR
40 Kb/s	15.80	15.80	8.01	7.22	21.4
30 Kb/s	15.80	14.30	4.58	5.82	16.48

TABLE 3.5: SNR Improvement Factors for DPCM(4,b)

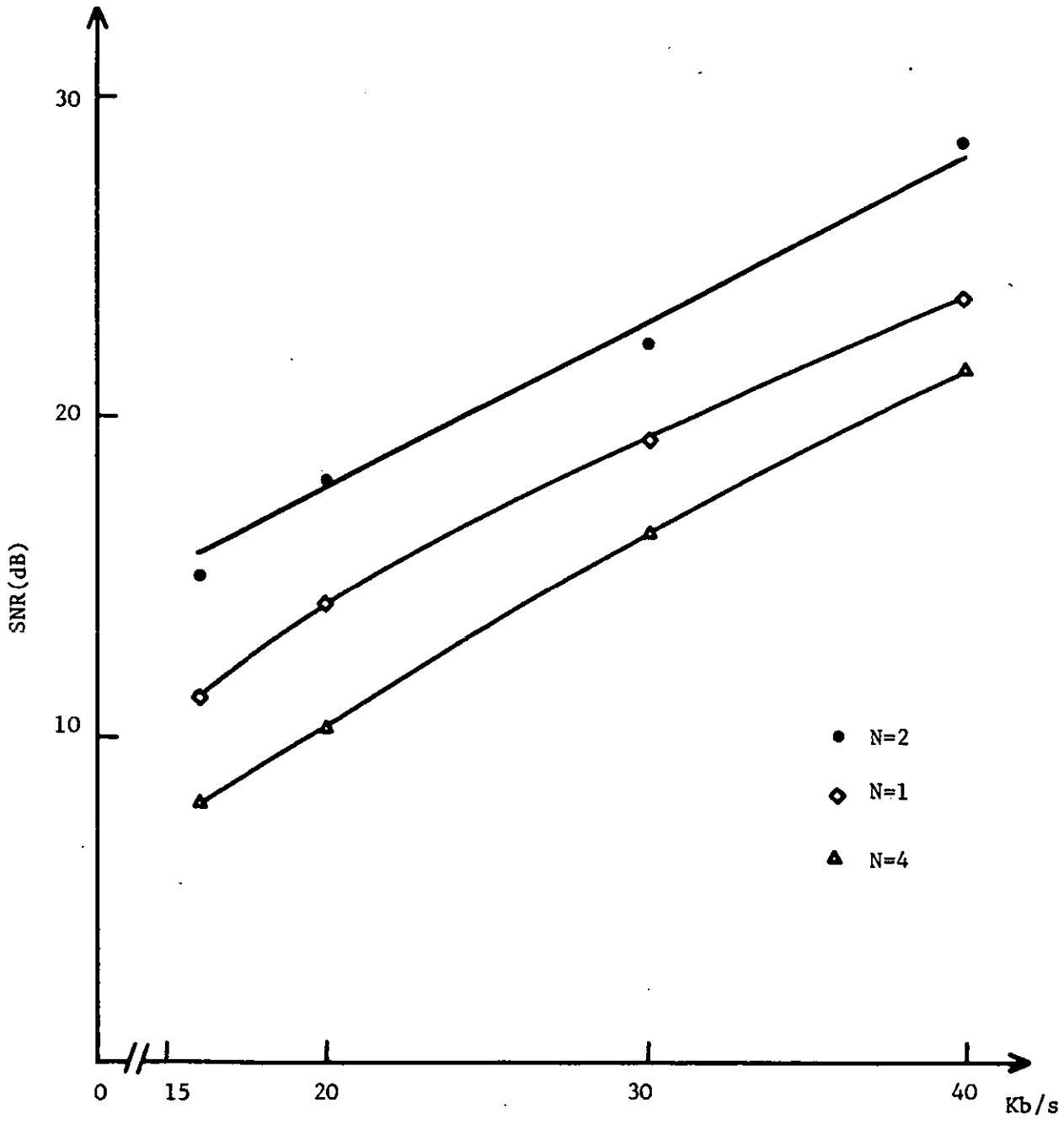


FIGURE 3.10: DPCM(N,b) At Various Transmission Bit-Rates

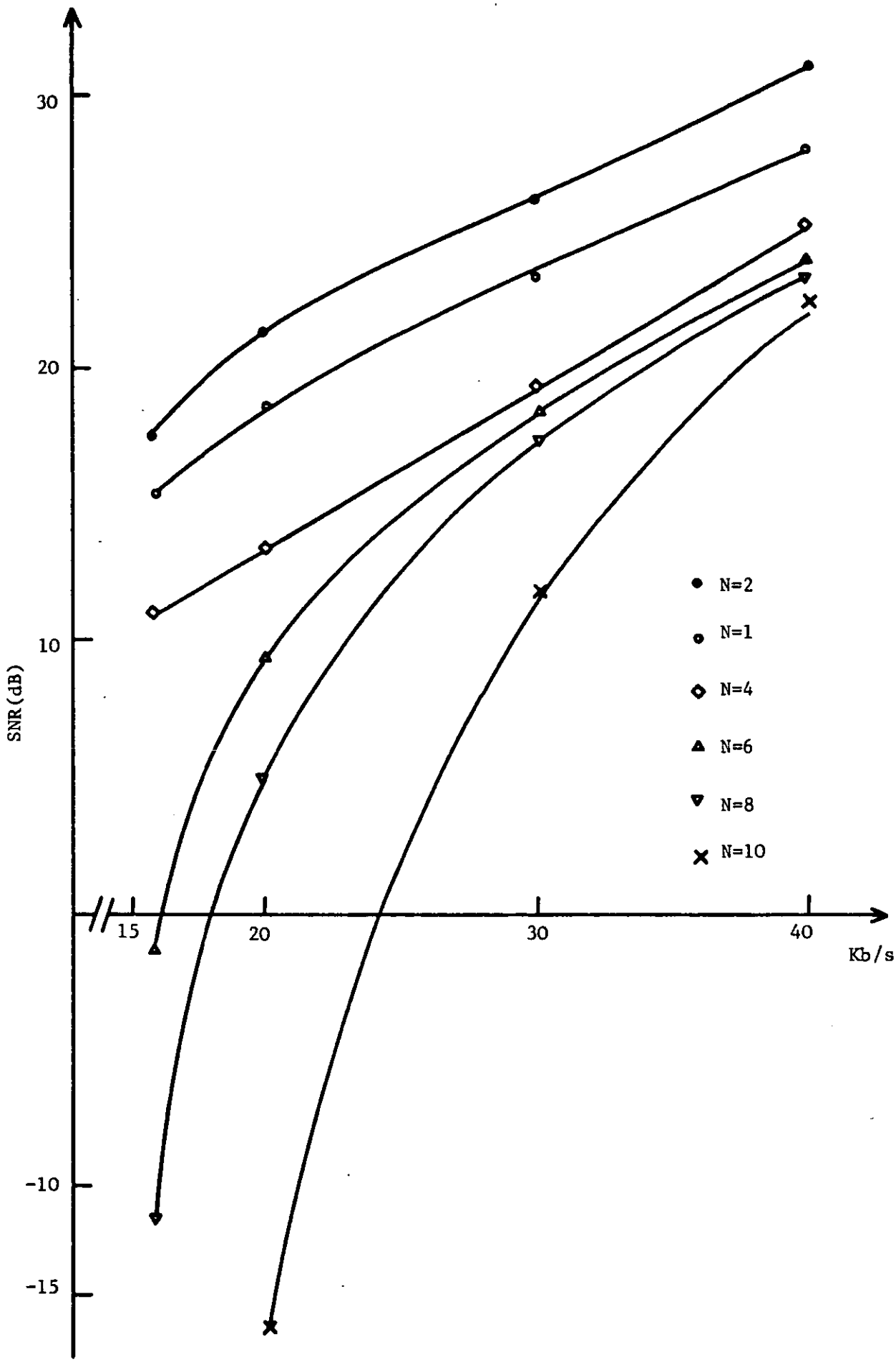


FIGURE 3.11: DPCM(N,b)-AQJ At Various Transmission Bit-Rates

Comparison of Table 3.3 with Table 3.5 reveals that, at 40 dB/s, SPR_a simulation of the second order predictor is reduced by about 8 dB when the predictor is replaced by a fourth-order one.

Similar points arise when Jayant's adaptive quantizer, AQJ is used instead of the fixed quantizer, see Figure 3.11. Furthermore, at lower transmission bit rates, an extremely surprising situation occurs. The higher order fixed predictors ($N \geq 6$) cause divergence.

At this stage our aim is to analyse the reasons why in Figures 3.10 and 3.11,

a. SNRI is not in agreement with the SNR measurements of the codec when $N \geq 4$,

b. The SNR values of the DPCM(4,4) and DPCM(4,4)-AQJ codecs are lower than the SNR values obtained from the DPCM(1,4) and DPCM(1,4)-AQJ codecs,

c. At lower data rates DPCM codec employing higher order predictors become unstable.

The aforementioned points can be attributed to many factors. Firstly, in deriving the SNRI in Equation (3.43) we assume that a) the power of the difference signal at the quantizer input, σ_e^2, satisfies the inequality $\sigma_e^2 << \sigma_x^2$, hence $\sigma_q^2 << \sigma_x^2$ and b) the quantization noise and the speech samples are uncorrelated. As a consequence of condition (a) it is also assumed that the input to the predictor is free from quantization noise. Several published reports state that the aforementioned assumptions hold reasonably well with the quantizer

having many levels ($N_Q \geq 8$). In our simulation results, however, the SNRI and SPR performance indicators differ substantially for both DPCM(4,4) and DPCM(4,3), i.e., for 8 and 16 levels quantization of $\{e_i\}$ see Table 3.5. This seems to suggest that the magnitude of the quantization error sample q_i is comparable to the speech sample x_i and thus the optimum set of coefficients, computed from the input speech statistics, is not well-matched to the noise contaminated feedback signal, $\hat{x}_i$. As a result, the inaccurate prediction of the input speech signal would give an increase in the power of the residual signal with a subsequent reduction in SNR. Furthermore, if the residual signal has a given p.d.f. when coupled with the second-order fixed predictor, it will experience a different p.d.f. for N=4. Consequently, instead of arranging the quantization thresholds with respect to the new p.d.f., if we still employ the same design for the quantizer, the quantization error will be significant and it will be correlated with the input speech. Since the error imposed on each sample can be as high as $\mp\Delta/2$, where Δ is the step size of the quantizer, it will reflect itself in $y_i = \sum_{k=1}^{N} a_k(x_{i-k}+q_{i-k})$, where y_i is the prediction output. This is to say that the random variation of q_{i-k} between $\mp\Delta/2$, coupled with optimally selected a_k prevents the accurate prediction of the incoming speech signals.[107] This effect of mismatching between the quantizer and the predictor will accumulate as we go along the encoding process of whole utterance samples, therefore the SNR of the codec will be degraded. In referring to question (a) and (b) we can conclude that maximizing the SNRI values

for N=4 onwards, will not precisely show itself in the SNR measurements of DPCM if the p.d.f. of the quantizer input signal is disregarded.

In Figure 3.11, we also observe that SNR increases almost uniformly with an increase in the bit-rate for N=1,2,4, but for N=6,8,10, DPCM codec causes divergence at lower data rates. This arises again from the mismatching between the two elements of the DPCM codec, namely the quantizer and the predictor. In other words, the error samples created by coarse quantization are now more significant than those at higher bit-rates. The coupling of such errors with the higher order optimum predictor ($N \geq 6$) certainly increase the prediction error and so the noise power in the codec. Furthermore, we thought that the effect of mismatching can be reduced by using the average rather than optimum, prediction coefficients computed from McDonald's[55] average speech data sampled at 9.6 kHz, see Figure 3.8, curve (b). This is because the location of the poles resulting from the average predictor polynomial is more tolerant to the noisy input signal. Table 3.6 presents both the optimum and the average prediction coefficients for N=1,2,4,6.

When we simulate the DPCM-AQJ codec with the average prediction coefficients we found that, at higher bit-rates the SNR of the codec for N=1,2, remains almost unchanged, while N=4 produces an SNR which is close to that of the first order codec. This confirms the aforementioned remark, i.e., that the average prediction coefficients are less sensitive to the noisy input. At lower bit-rates, the predictors of order 6,8 in the codec still improves the previously reported SNR value.

	N	a_1	a_2	a_3	a_4	a_5	a_6
OPTIMUM	1	0.9460					
	2	1.7460	-0.8461				
	4	2.550	-2.870	1.810	-0.5700		
	6	3.050	-4.590	4.690	-3.580	1.840	-0.490
AVERAGE	1	0.9035					
	2	1.6316	-0.8057				
	4	2.0230	-1.7505	0.8050	-0.2130		
	6	2.0590	-1.8860	1.1411	-0.7830	0.5631	-0.2442

TABLE 3.6: Predictor Coefficients Both for the Sentence in Section 3.6.1 and for the Average Data

For example, SNR of DPCM(6,1.58)-AQJ with optimum coefficients was increased from -1.2 dB by a few dB's although it is still far below an acceptable threshold of SNR. In other words, system divergence at lower bit-rates occurs even if we employ the average prediction coefficients.

To obtain a quantitative measure of the predictor mismatch, a distortion measure, d_m was introduced.[108] For N=1,2,4,6, we found d_m values as 0.0043, 0.0775, 0.089 and 0.26 respectively. The values of d_m suggest that as we increase N from 1 to 6, d_m also increases. Especially at N=6, d_m=0.26 is very close to the statistically significant threshold level of 0.3. This, once again confirms that system divergence with higher-order predictors takes place. Under these circumstances, without going into further detail, we can possibly conclude that the performance of higher-order predictors coupled with coarse quantization is limited unless:

(i) in selecting the prediction coefficients we minimize

$$\sigma_e^2 = \langle (x_i - \sum_{k=1}^{N} a_k \hat{x}_{i-k})^2 \rangle$$

rather than,

$$\sigma_e^2 \simeq \langle (x_i - \sum_{k=1}^{N} a_k x_{i-k})^2 \rangle ,$$

(ii) the memory of the adaptive quantizer is restricted so that the past poor prediction values are forgotten with time.

In (ii), the quantizer step-size is adjusted in accordance with the Equation (3.50) having β_q values. In this way excessive quantization noise may be reduced. Therefore, in referring to question (c), we can emphasize that if we insist on employing higher-order fixed predictors

at lower bit rates, the above two points (i) and (ii) should be taken into account.

3.6.4 Performance of Forward Block Adaptive Predictors in DPCM Codec Employing AQJ, FBADPCM(8,b)-AQJ

As described in Section 3.4 the FBA predictor updates its coefficients every W/f_s seconds so that the prediction coefficients are matched to the short term, rather than to the long term characteristics of the input speech signal.

When experimenting the FBADPCM codec, we have selected the order N of the predictor to be 8. The reason for using such a high order predictor is twofold. First, from our recent experience we know that the higher order fixed predictors cause divergence at lower bit-rates. Second, at high bit-rates, SNR performance of codec with higher order predictors is inferior to that employing lower order predictors.

Figure 3.12 curve (a) presents the results of FBADPCM(8,b)-AQJ whose coefficients are updated every 2 msec. (W=20) by using Equation (3.47). For reference purposes results of a DPCM(8,b) codec are also included, see curve (b). We observe that, for a wide range of transmission bit-rates, the FBADPCM codec shows a better SNR performance when compared to the DPCM system with the fixed predictor, see curve (b). Furthermore, at lower bit-rates, a significant improvement in SNR effectively hinders system divergence. This is because, frequent

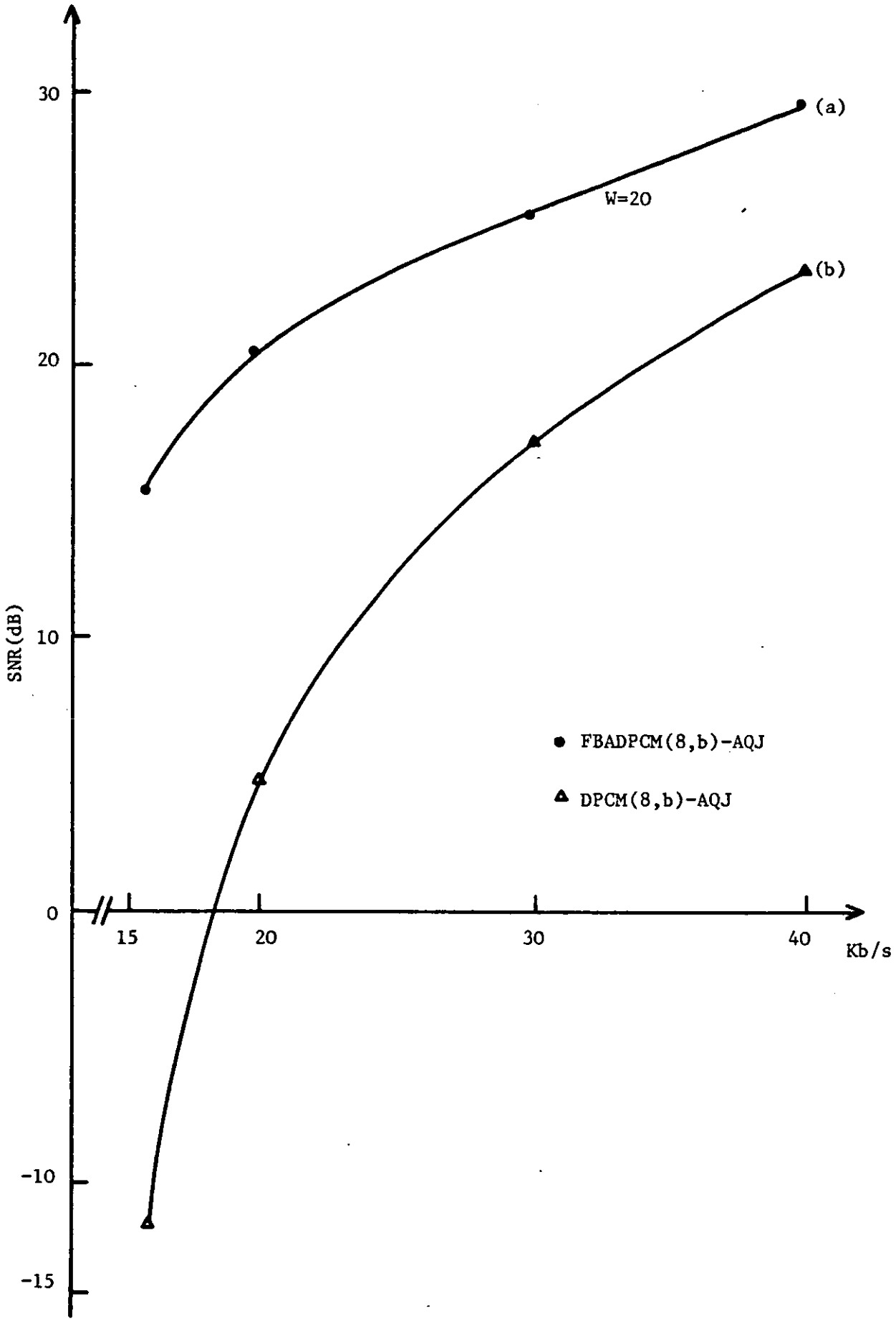


FIGURE 3.12: Variation of SNR with Transmission Bit-Rates
a. FBADPCM(8,b)-AQJ, b. DPCM(8,b)-AQJ

adaptation of the prediction coefficients switches the codec between unstable configurations and as a result it becomes stable.

When the block length, W varies between 20 and 120 samples the SNR of FBADPCM is reduced as shown in Figure 3.13. This is attributed to the content of the speech segment. For example, a large block of W samples is likely to include a change in the statistics of the speech signal and in this case the autocorrelation method will provide an average, sub-optimum solution. Thus as we decrease W, we effectively improve the prediction accuracy of the forward block adaptive predictor.

Although FBA predictors, both stabilize the encoder when operating at relatively low bit-rates, approximately 16 Kb/s and improve its SNR performance at higher bit-rates, they require additional channel capacity for the transmission of the prediction coefficients. The usual procedure is to reduce the bit-rate assigned to encode the residual signal $\{e_i\}$, and thus to accommodate the side information for a specified transmission bit-rate. In this way, the multiplexing of the side information together with the reduced bits for the quantization, keeps the total channel bit-rate unchanged. In our experimental results, presented in Figures 3.12 and 3.13, however, no such attempt has been made. Therefore, for a given output bit-rate, the SNR is reduced compared to the values shown in Figures 3.12 and 3.13. The loss in SNR given by Equation (3.48) is minimal and depends on the block length, W and the number of bits used for the encoding of each prediction coefficient prior to the multiplexing process.

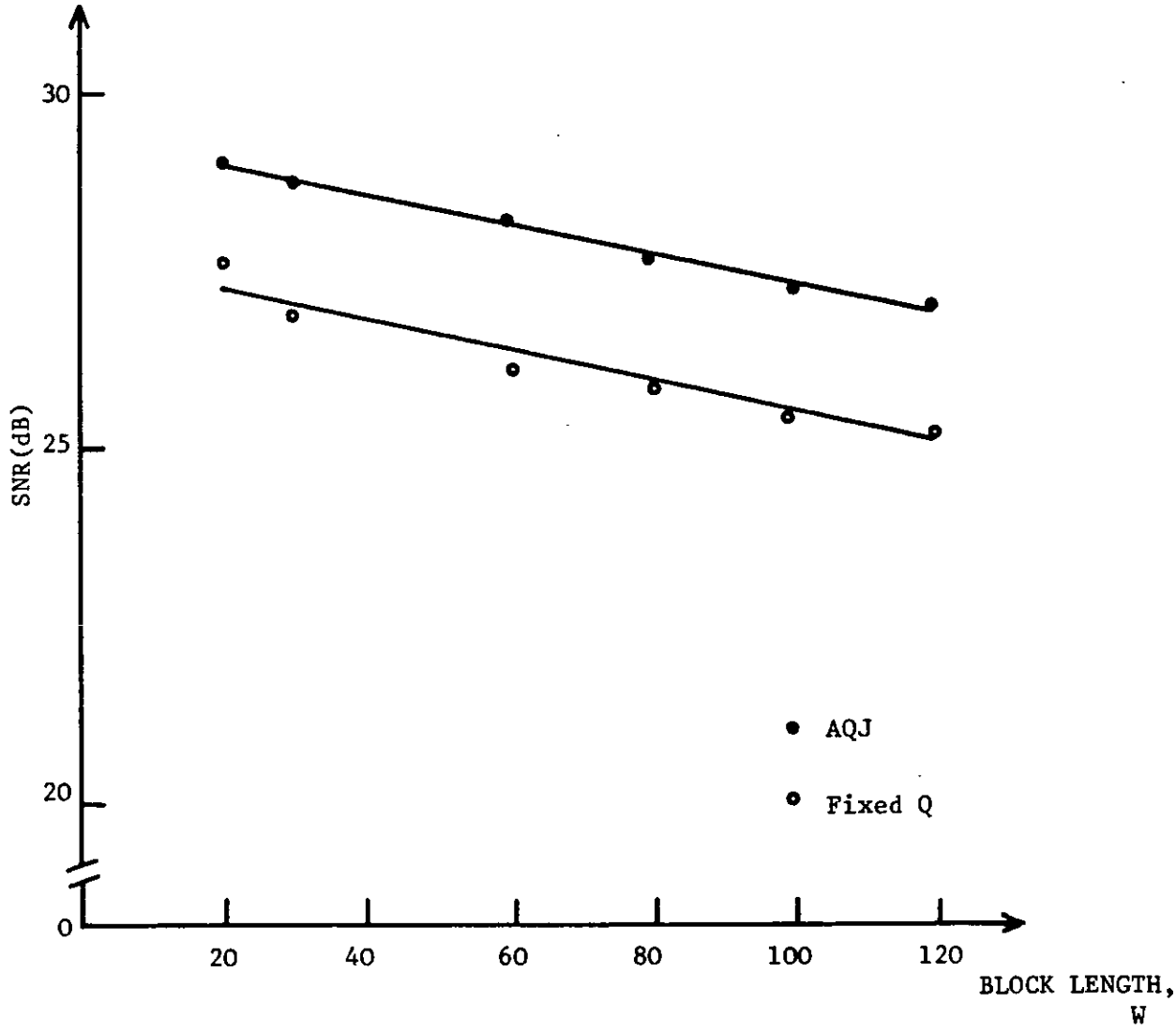


FIGURE 3.13: Variation of SNR(dB) with Block Length, W

3.6.5 An Effect of Channel Errors in the Performance of Codec

A digital communication system, like DPCM, is in general, designed to withstand a certain amount of signal distortion. These signal impairments, are usually, attributed to:-

(i) terminal effects - quantization errors

and (ii) channel effects - random errors and the inaccuracy of the regenerators in the channel.

It is well known that in (i) the quantization errors can be reduced by using a large number of quantization levels and therefore by increasing the size of the code-words assigned to the quantized samples. In addition, adaptive quantizers whose step size is designed to expand or contract, in accordance with the amplitude range of the input signal to the quantizer can efficiently reduce the amount of quantization noise for a given number of bits per sample. In (ii), both the amplitude and the frequency modulation effects of the channel together with the inaccurate regeneration of the transmitted code-words deteriorate the ability of the receiver to form the correct code-words.

We have already presented the results of adaptive quantizer having large number of levels and now we examine the case (ii). In order to simulate a DPCM codec operating with random channel errors, the following steps have been adopted:

a) In Figure 2.14, the channel input vector L_i of dimension "b" bits/sample is evaluated from

$$\hat{e}_i = L_1.2^{b-1}+L_2.2^{b-2}+\ldots+L_b.2^0 \qquad (3.54)$$

where $L_1, L_2, \ldots, L_b$ are 0 or 1.

For b=4, Equation (3.54) is rewritten as

$$\hat{e}_i = L_1.2^3+L_2.2^2+L_3.2^1+L_4.2^0 \qquad (3.55)$$

where L_1 is the most significant bit, MSB, and L_4 is the least significant bit, LSB.

b) N'_Q, the number of the quantization level occupied by the quantizer output sample $\hat{e}_i$, is divided by the weighting factor of the MSB, for b=4, it is $2^3=8$. Since $0 \leq N'_Q \leq 16$, L_1 is either 1 or 0. If $L_1=1$, $N''_Q=N'_Q-8$. Now N''_Q is divided by the weighting factor of L_2 ($2^2=4$). Since $0 \leq N''_Q \leq 8$, L_2 is either 1 or 0. If $L_2=1$, $N'''_Q=N''_Q-4$. This process continues until the binary value of L_4 is evaluated. In this way, the maximum negative level of quantizer is assigned with L_i=0000 while L_i=1111 corresponds to the maximum positive level of the quantizer.

c) In order to introduce random errors in the channel a random number generator was employed. For a given Bit-Error-Rate, BER, a total number of samples, NS, the total number of samples deemed to be in error, NSAM, was calculated from

$$NSAM = \frac{NS.BER.b}{100} \qquad (3.56)$$

Then the locations of these NSAM samples were selected using a NAG Library subroutine, called G05DAF,[106] which generates a sequence of random numbers in the range of 1 to NS.

Then, for each of the selected NSAM samples the G05DAF random generator is employed again to provide a number between 1 and b which identifies the erroneous bit inside the code-word. The value of this bit is then reversed from 0 to 1 or vice versa and in this way the channel errors are introduced. Finally, using Equation (3.54), the erroneous code-word is used to produce the received e_i' sample which is then further processed by the following decoding process.

Figure 3.14 presents the variation of the SNR versus bit-error rate, BER %, for the FBADPCM(N,4) system. In this case we assume that a) the probability for two adjacent samples to be in error is very small, and b) the predictor coefficients are transmitted in an error protected form. In Figure 3.14, two points can be clearly observed. First, without the use of an error protection scheme to combat the channel errors, the SNR decays rapidly as the BER % increases. Second, as the order, N of the predictors used in the system increases the SNR deteriorates rapidly in the presence of channel errors. This is because using larger values of N and thus longer feedforward filter structures, the channel errors affect more decoded speech samples causing extended accumulation of errors and instabilities in the detection process. With regard to the first point mentioned above, concerning channel error protection schemes, the severity of the degradation depends on the bit which is inverted. Specifically, if the error occurs in the MSB, the effect becomes detrimental. Noll[41] considered two formats of error protection, namely the protection of MSB and the protection of two MSB's. Although such codes improve the SNR performance of the

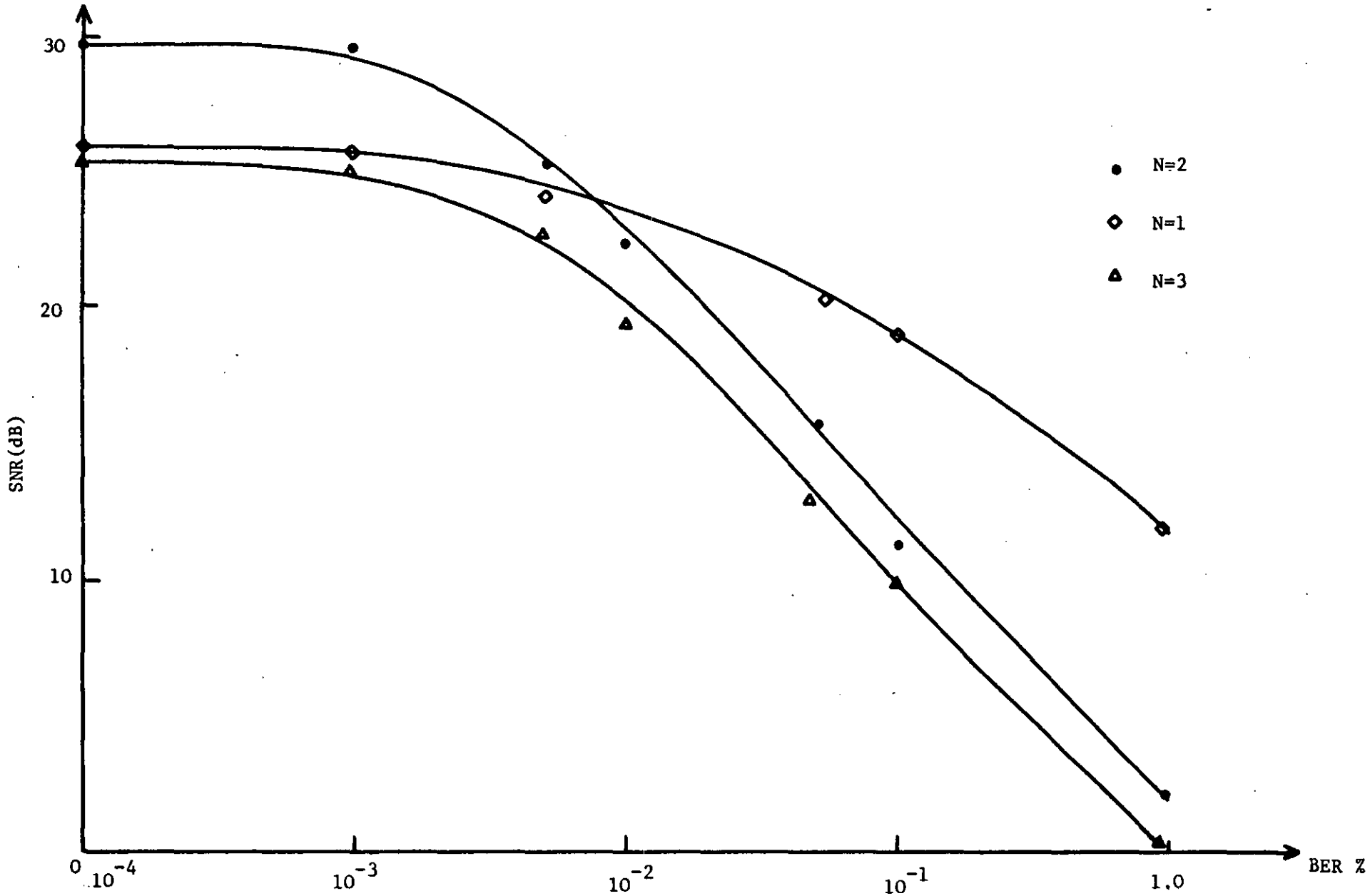


FIGURE 3.14: Variation of SNR with BER % in FBADPCM(N,4) Codec, W=128

codec, they increase both the channel bandwidth and the complexity of the system. A different approach suggested by Steele et al,[109] known as Difference Detection and Correction, DDC, attempts to locate, by statistical means, and correct erroneous samples at the receiving end. The authors claim significant improvement in SNR without an increase in channel capacity. However, one possible drawback in DDC is that some correct samples can be judged as erroneous and falsely corrected. Jayant[110] examined the effect of channel errors in mobile radio telephony where the BER is in excess of 10^{-2} (in most digital transmission systems, BER is less than 10^{-5}). In his 3-bit codec, he sends the MSB three times, the second bit twice and LSB once. The decision on the value of the first bit is achieved by majority count. If the two received values of the second bit differ, then the value accepted is that which constrains the output waveform to have the lower slope. The LSB is accepted as correct. This technique increases the channel bandwidth and does not guarantee accurate decisions, since multipath Rayleigh fading type of distortion in the link may invert all the bits for any sample.

3.7 CONCLUSIONS

This chapter has examined both fixed and forward block adaptive, FBA, predictors as applied to DPCM systems for encoding speech signals. Both predictors were used in these differential structures with fixed and Jayant's adaptive quantizers. The SNR curves were considered as performance indicators in the comparison of the various codec configurations.

The design of fixed predictors employs the long term statistics of the speech signal and for this purpose the autocorrelation, rather than autocovariance, method was employed. The upper bound SNR improvement factor, SNRI was evaluated both for the tested speech data and for McDonald's[55] average data. Because of the high correlation between the samples of the input speech signal used in our experiments, high SNRI values were obtained.

When such predictors with an order, N, between 1 and 10 were used in DPCM codec, it was observed that for all the bit rates from 15.84 to 40 Kb/s, the systems having only 2 taps provided an improved SNR over the codecs with N=1 and N>2. An inspection of second order predictor results shows that SNRI values are reflected in SNR measurements when the quantization is fine. However, as we reduce the bit-rate SNRI looses its significance because of the excessive noise produced by the coarse quantization which appears at the input to the predictor. Furthermore, when the SNRI is replaced by SPR_a, both theoretical and

simulation results show that the latter is a more accurate improvement factor than SNRI.

An increase in the order of the predictor from 2 to 4 at higher bit-rates, indicated that the SNR performance of the codec employing fixed or Jayant's Adaptive quantizer is deteriorated, in contrast to results shown by the SNRI measure. The reason for this was attributed to the mismatching effect of the 4^{th}-order optimum predictor coefficients to its decoded input samples and the p.d.f. of the input signal to the DPCM quantizer. This also agrees with Gibson's[105] results that maximizing the SNRI or SPR_a factors does not necessarily maximize the SNR unless the p.d.f. of the quantizer input signal is taken into account. Then, replacing the optimum coefficients by the average coefficients, we notice that previously reported SNR measurements of the 4^{th}-order predictor are improved by a few dB's since the average coefficients are more tolerant to the decoded signal, contaminated by quantization noise. At lower bit-rates, higher-order ($N \geq 6$) fixed predictors cause system divergence confirming another of Gibson's observations.[65]

Finally, in an attempt to improve the performance the DPCM codec, the fixed predictors were replaced by FBA predictors where the coefficients are updated periodically. When compared to 8^{th}-order DPCM at higher bit-rates, the FBADPCM system, changing their coefficients, every 20-120 samples shows an SNR improvement of about 5 dB's. Another substantial advantage, at lower bit-rates, of the FBADPCM over the DPCM

with the fixed predictor, is that the FBA predictors being switched between unstable configurations, prevent the previously reported system divergence of the fixed systems. On the other hand, the disadvantages of the FBADPCM system are:

(i) the evaluation of the optimum coefficients per block basis requires the inversion of the autocorrelation matrix and its multiplication by the autocorrelation vector which considerably increases the complexity of the system.

(ii) the required transmission of the prediction coefficients increases the channel bit-rate. As an example the codec with a 2^{nd}-order predictor which updates its taps every W samples requires 8 bits/block overhead information provided that each prediction coefficient is encoded using a 4 bit code-word. The transmission of these coefficients to the receiving end can be viewed as a SNR loss of 1.2 or 0.37 dB for block sizes of 40-128 samples respectively.

Thus, at the end of this chapter we see that the DPCM codecs having fixed predictors have their limitations in following the rapid variations in speech statistics while the disadvantage of the DPCM systems with FBA predictors results from the required overhead information.

In the following chapters, we direct our efforts towards the development of novel and efficient speech prediction algorithms.

CHAPTER IV

SEQUENTIAL PREDICTORS

4.1 INTRODUCTION

The simulation results of Chapter III demonstrated the inadequate performance of DPCM system employing a fixed *"optimized"* predictor. The main reasons for the limitations in the performance of such a codec are as follows:

a. The fixed predictor attempts to model the characteristics of an *"average"*, fixed vocal tract shape. Obviously, this type of prediction can be efficient with stationary signals, but not for speech where the characteristics of the vocal tract are varying with time.

b. Since voiced speech occurs much more frequently than unvoiced, a fixed predictor designed from the long term statistics of the speech signal is more accurate in the prediction of voiced than unvoiced speech. Indeed, for unvoiced signals the SNR performance of a fixed predictor is relatively poor.

c. At low transmission bit-rates, having the decoded speech samples at the input of the predictor, severely distorted by excessive quantization noise, the system becomes unstable. This is due to the mismatch between the statistics of the decoded speech and the statistics of the input speech used in the design of the predictor.

"Block Adaptive" predictors were also examined in the previous chapter. Here, the weighting coefficients of the linear predictor were adjusted according to the short-term statistics of the speech

signal and the changes in the vocal tract transfer function could be handled more efficiently when compared to fixed prediction. As a result, the prediction error signal has now a smaller amplitude range which produces a smaller quantization noise power. In addition, this block adaptive predictor eliminates the system divergence at lower bit-rates. The block adaptation strategy has however, the following disadvantages: when employed in a feedback mode where the prediction coefficients are estimated from the decoded speech samples, their accuracy is considerably reduced because of the one block delay introduced in updating the coefficients. Also, in the case where *"block adaptation"* is used in a *"feedforward"* mode, the transmission of the coefficients is required as side information.

In recent years alternative *"sequentially adaptive"* prediction techniques which avoid the disadvantages of both the previous methods, have received considerable attention in the field of speech coding. The advantage of sequential predictor when compared to the block adaptive scheme arises from the fact that a DPCM with a sequential predictor eliminates the necessity for transmitting the values (side information) of the prediction coefficients.

Thus, our research efforts, described in this chapter were directed towards the behaviour of the sequentially adaptive estimation methods whose coefficients are updated in a sample-by-sample basis.

In all sequential algorithms examined here, the canonical form of Equation (3.7) is adopted, rather than the lattice structure of

Itakura-Saito.[32] Initially, the Stochastic Approximation Predictor, SAP is analysed and the adaptation rate of its prediction coefficients is shown to be inadequate to follow the fast variations in the statistics of a speech signal. In order to obtain faster convergence to the *"optimum"* coefficient values, a novel technique called Sequential Gradient Estimation Prediction, SGEP, is devised and investigated. The mathematical analysis of its convergence is also presented. Further, parallel SAP/SGEP configurations and an application of sequentially adaptive prediction in a noise cancellation scheme, are briefly mentioned.

4.2 SEQUENTIAL PREDICTION PROBLEM APPROACH

The technique of designing an optimum predictor in a mean square error sense was first conceived in 1942[102] and then improved by Wiener.[101] The latter work made possible the design of a linear predictor for estimating stationary signals. For higher order predictors, Wiener relationship in a matrix form, is given by Equation (3.21), i.e. $\hat{A}_{opt}=C^{-1}C_0$, where C is the autocorrelation matrix and C_0 is the autocorrelation vector. This method applies to signals whose autocorrelation function is known and therefore the prediction can be optimum when the statistical characteristics of the signal to be predicted match the apriori information used in the design of the predictor.

In order to apply, in an optimum way, Wiener's prediction approach to non-stationary signals, the short term signal statistics must be considered. That is the input speech signal is assumed to be stationary within short time intervals and the *"short term"* autocorrelation function is measured for successive blocks of speech samples.

Now, when the average long term speech statistics are used in computing the prediction coefficients, we minimize the error function, FU,

$$FU = \langle e_i^2 \rangle = \langle (x_i - \sum_{k=1}^{N} a_k x_{i-k})^2 \rangle . \qquad (4.1)$$

The optimum solution, $\hat{A}_{opt}$ is obtained by equating to zero the gradient of FU, with respect to the a_k, k=1,2,...,N,[111] i.e.,

$$\nabla(FU) = \nabla\langle e_1^2\rangle \simeq \nabla e_i^2\Big|_{\hat{A}=\hat{A}_{opt}} \tag{4.2}$$

$$\simeq 2(C\hat{A}_{opt} - C_0) = 0 \ . \tag{4.3}$$

The set of optimum coefficients which form $\hat{A}_{opt}$, zeroes the value of $\nabla(FU)$ and represents a minimum point of the $FU=\langle e_i^2\rangle$ function. This function can be visualized as a parabolic shape function of the prediction coefficients $\{a_k\}$, see Equation (4.1). The bottom point of this parabolic surface is unique and corresponds to the optimum solution $\hat{A}_{opt}$.

When the short term statistics are known, the error function, $FU=\langle e_i^2\rangle$ is minimized over a short segment of speech and the parabolic surface with its minimum point, changes according to the short term autocorrelation functions obtained from the speech signal.

A sequential predictor however, that possesses the incoming signal while updating its coefficients at every sampling instant, eliminates the requirement of apriori knowledge of the input signal statistics. Thus the sequential adaptive predictors, being able to adapt the characteristics according to the varying statistics of non-stationary signals have significant applications in the fields of speech encoding, noise cancellation, equalization, modelling of transfer function,etc. The adaptation of such a predictor is achieved by an iterative algorithm where the new coefficients are calculated, every sampling instant, using information related to signal samples which have been already processed.

In sequential predictors, there exist two modes of operation, namely the *"learning"* mode and the *"normal"* mode. The first one is observed when the system starts to operate on the signal and before the algorithm settles down. In this mode of operation large fluctuations in the values of the coefficient are inevitable. The *"normal"* operation which follows the learning mode is observed when the predictor models and predicts the input signal with a reasonable accuracy.

All sequential prediction techniques require an initial vector, $\hat{A}_0$, of prediction coefficients to be specified and they proceed by generating a sequence of vectors, $\{\hat{A}_{i,k}\}$ where i refers to the i^{th} sampling instant and k indicates the order of the predictor. The generalized equation for the adaptation of the k^{th} prediction coefficient at the $(i+1)^{th}$ sampling instant is given as:

$$a_{i+1,k} = a_{i,k} - g \frac{\partial(FU)}{\partial a_{i,k}} \tag{4.4}$$

where g controls the adaptation speed of the algorithm. The $g\partial(FU)/\partial a_{i,k}$ term, subtracted from $a_{i,k}$, is the gradient of the error function (FU) with respect to $a_{i,k}$, k=1,2,...,N, and the values of the prediction coefficients are updated in the direction opposite to the gradient of the error. That is why, such a predictor is sometimes referred to as a gradient predictor.

If the error function to be minimized is the mean square error, i.e., $FU=\langle e_i^2 \rangle$, the k^{th} component of the gradient is given by

$$\frac{\partial (FU)}{\partial a_{i,k}} = \frac{\partial < (x_i - \sum_{k=1}^{N} a_k x_{i-k})^2 >}{\partial a_{i,k}} \qquad (4.5)$$

or

$$\nabla (FU) \simeq -2 \begin{bmatrix} x_{i-1} \\ x_{i-2} \\ \vdots \\ x_{i-N} \end{bmatrix} (x_i - \sum_{k=1}^{N} a_k x_{i-k}) \qquad (4.6)$$

$$\downarrow$$

$$\hat{X}_i$$

The approximation sign ($\simeq$) comes in since, to differentiate Equation (4.5), $<e_i^2>$ has been approximated to the sample error, e_i^2. Substituting the k^{th} component of Equation (4.6) in (4.4) and having 2g=h, we obtain,

$$a_{i+1,k} = a_{i,k} + he_i x_{i-k} , \qquad (4.7)$$

or in vector form,

$$\hat{A}_{i+1} = \hat{A}_i + he_i \hat{X}_i , \qquad (4.8)$$

where h is a constant and controls the rate of adaptation. The convergence of Equations (4.7) and (4.8) is guaranteed for values of h between 0 and 2.[112]

Comparison of Equations (3.21), (4.6-8) reveals that the sequential methods seek to optimize the coefficient vector in a recursive way rather than solving a matrix equation. Figure 4.1 presents the block diagram of a sequential predictor. The predicted sample at i^{th} instant is y_i and

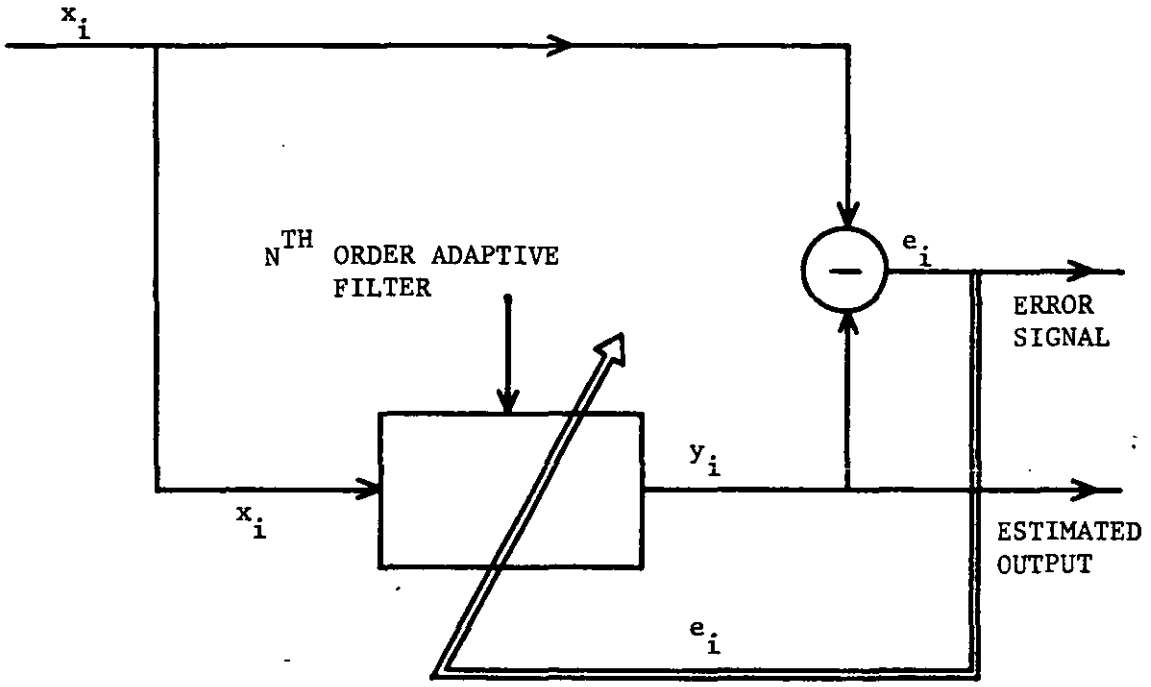


FIGURE 4.1: Simple Diagram of Sequentially Adaptive Filter

$$y_i = \sum_{k=1}^{N} a_k x_{i-k} \tag{4.9}$$

and the error introduced in predicting x_i is

$$e_i = x_i - y_i \tag{4.10}$$

Hence,

$$x_i = \sum_{k=1}^{N} a_k x_{i-k} + e_i \, . \tag{4.11}$$

In a vector form, Equation (4.11) may be expressed as

$$x_i = \hat{A}_i^T \hat{X}_i + e_i \tag{4.12}$$

where $\hat{A}_i$ is the coefficient vector and $\hat{X}_i$ is a vector whose elements are the N previous speech samples, i.e.,

$$\hat{A}_i^T = [a_1, a_2, a_3, \ldots, a_N]$$

$$\hat{X}_i = [x_{i-1}, x_{i-2}, \ldots, x_{i-N}]^T .$$

We are now to discuss three sequential predictors, namely the Kalman, the Stochastic Approximation Predictor, SAP and a novel technique called the Sequential Gradient Estimation Predictor, SGEP, which has been developed during the course of the work described in this thesis. The performance of SAP and SGEP will be evaluated on the basis of SNR curves, error waveforms, processing time and rate of convergence. Then, the efficiency of these two algorithms will be compared with the performance of the Kalman algorithm as reported in the literature.[113]

4.3 KALMAN PREDICTOR[60,63,64,113-115]

The general form of Kalman prediction coefficients is expressed as

$$\hat{A}_{i+1} = Y\hat{A}_i + \hat{W}_i^0 \tag{4.13}$$

where Y is an constant (N×N) matrix and $\hat{W}_i^0$ is an N dimensional column vector of white noise terms with zero mean and stationary variance V_w. The procedure to define and then solve the general Kalman algorithm of Equation (4.13) requires considerable processing time. This is perhaps why the majority of the research workers, working on Kalman predictors,[63,64] assumed that the effect of $\hat{W}_i^0$ on $\hat{A}$ is small, i.e., the coefficients change slowly from sample to sample and the Y matrix is the identity matrix I. The resulting algorithm is sometimes referred to as the Simplified Kalman Algorithm, and the prediction coefficients are obtained when the $h\hat{X}_i$ term of the gradient algorithm, described by Equation (4.8), is replaced by the vector $G_{KAL}(i)$, i.e.,

$$\hat{A}_{i+1} = \hat{A}_i + G_{KAL}(i)e_i \tag{4.14}$$

where $G_{KAL}(i)$ is the Kalman filter gain vector. $G_{KAL}(i)$ is computed recursively in accordance with

$$G_{KAL}(i) = \frac{V_{i-1}\hat{X}_i}{V_e + \hat{X}_i^T V_{i-1}\hat{X}_i} \tag{4.15}$$

where V_{i-1} is the predictor coefficient error variance (N×N) matrix and V_e is an experimentally selected constant. The V_{i-1} matrix can be obtained iteratively from

$$V_i = V_{i-1} - \frac{V_{i-1}\hat{X}_i\hat{X}_i^T V_{i-1}}{V_e + \hat{X}_i^T V_{i-1}\hat{X}_i} \tag{4.16}$$

where vector $\hat{X}_i$, as before is $[x_{i-1}, x_{i-2}, \ldots, x_{i-N}]^T$. Substitution of Equation (4.15) in Equation (4.16) yields

$$V_i = V_{i-1}[I - G_{KAL}(i)\hat{X}_i] \ . \tag{4.17}$$

The quantity in the denominator of Equations (4.15) and (4.16) is scalar and hence the necessity of a matrix inversion in obtaining $G_{KAL}(i)$ and V_i is eliminated. Further, $G_{KAL}(i)$ behaves as an automatic gain control factor since it adjusts the coefficients so that they are not being overcorrected when the speech amplitudes are large or vice versa.

Finally, it should be mentioned that, in speech coding applications of the Kalman algorithm a decaying constant β_d is introduced in such a way that it makes the algorithm more robust in the presence of channel errors, i.e.,

$$\hat{A}_{i+1} = \beta_d \hat{A}_i + G_{KAL}(i)e_i \tag{4.18}$$

and $\beta_d < 1$.

4.4 STOCHASTIC APPROXIMATION PREDICTOR, SAP

The SAP algorithm is also based on the Equation (4.7), i.e.,

$$a_{i+1,k} = a_{i,k} + he_i x_{i-k} \quad , \quad k=1,2,\ldots,N$$

which is originated from the general gradient adaptation procedure of Equation (4.4) with $FU=\langle e_i^2 \rangle$ taken as the error function.

Cummiskey[62] made an intensive study in this area and also examined the case where FU is a function of the absolute value of the error between the input and the predicted values, i.e., $FU=\langle |e_i| \rangle$. For more details, see Appendix E.

In the SAP algorithm, h is defined as

$$h = \frac{A}{B + \zeta(x_i,M)} \tag{4.19}$$

where A,B are constants and $\zeta(x_i,M)$ is a function of the M previous speech samples, that is

$$\zeta(x_i,M) = \frac{1}{M} \sum_{k=1}^{M} x_{i-k}^2 \quad . \tag{4.20}$$

Cummiskey[62] used a zero value for B without realising the necessity of this bias term B in the denominator of Equation (4.19), specifically, during the unvoiced or the silence periods of the speech signal. The complete form of SAP was examined later by Gibson et al.[63,64]

The term $B+\zeta(x_i,M)$ is a form of an automatic gain control and tends to equalize the adaptation rate of the algorithm as the input speech power level varies. As the power level increases $B+\zeta(x_i,M)$

also increases, therefore h decreases and overcorrections of the $a_{i+1,k}$ coefficient are avoided, preventing the occurrence of a large prediction error. For silence or unvoiced segments of speech data $\zeta(x_i,M)<<B$ and Equation (4.19) maintains a finite value. Thus the bias term B compensates for the low level input signals. Henceforth, h will be replaced by $P_i(x)$ since it is variable and changes at every sampling instant, i.e.,

$$P_i(x) = \frac{A}{B + \frac{1}{M}\sum_{j=i-M-1}^{i-1} x_j^2} \quad . \tag{4.21}$$

Figure 4.2 presents the block diagram of the SAP algorithm. Appendix E deals with Cummiskey's[62] algorithm with $FU=<|e_i|>$ and gives the differences between this algorithm and the SAP.

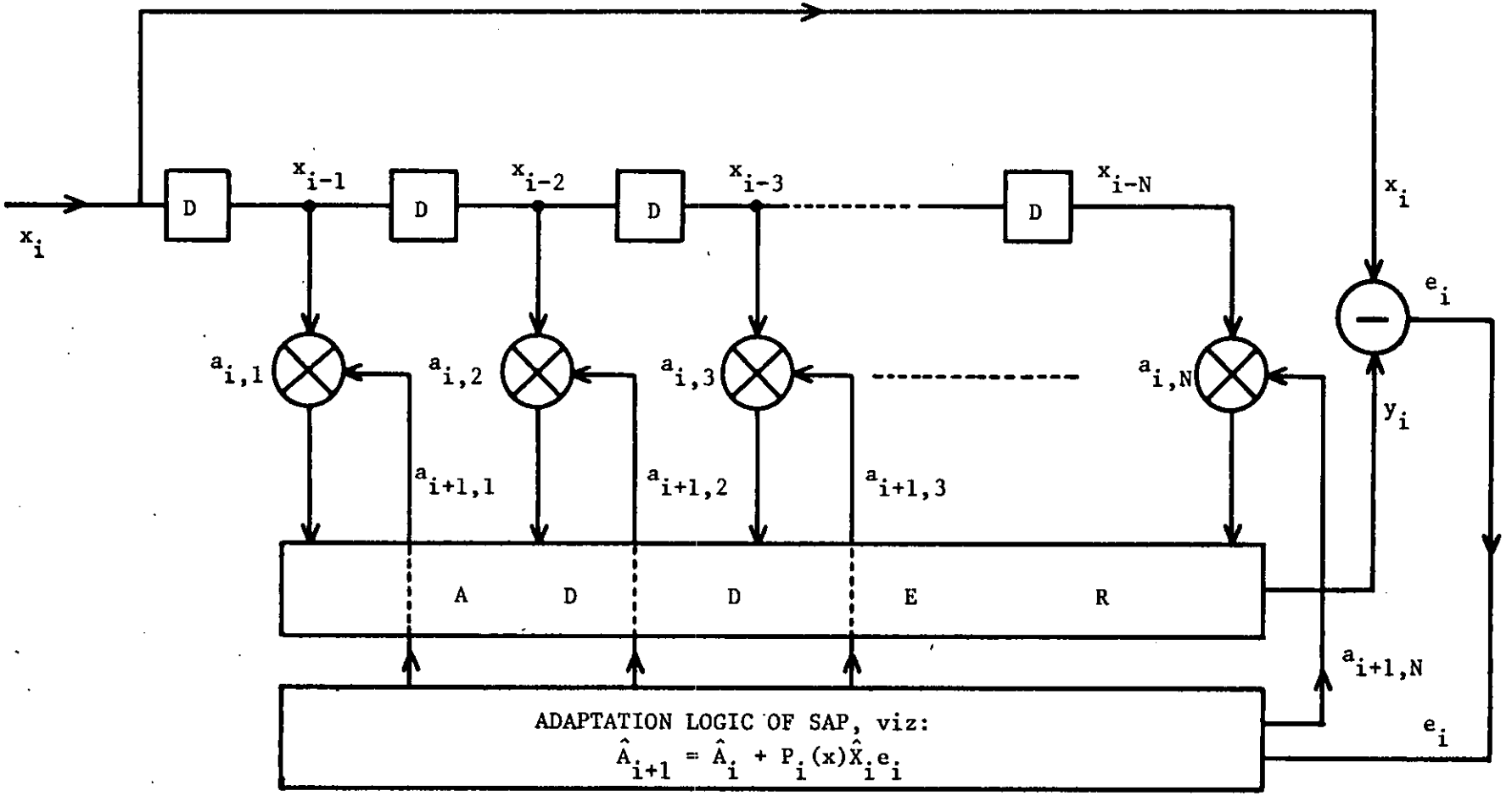


FIGURE 4.2: Operation of SAP Algorithm

4.5 SEQUENTIAL GRADIENT ESTIMATION PREDICTOR, SGEP

In the sequential gradient estimation predictor to be described here, the canonical form specified in Equation (4.9) is again used. The minimization of the mean square prediction error of the correlation or the covariance methods is, however, replaced by a more versatile criterion which attempts in general, to minimize an error function, FU. As an example, this function, FU can be made dependent on the modulus of the prediction error or the square of this error or the cubic function of this error or the differential of the error, etc.

The SGEP algorithm also updates its coefficient according to the general gradient formula given by Equation (4.4), but $g\partial FU/\partial a_{i,k}$ is being replaced by $P_i(x).\Lambda_{i,k}.k^{-\alpha}$, and the k^{th} predictor coefficient at $(i+1)^{th}$ sampling instant is now obtained as

$$a_{i+1,k} = a_{i,k} - P_i(x).\frac{\Lambda_{i,k}}{k^{\alpha}} \quad , k=1,2,\ldots,N, \tag{4.22}$$

or in a vector form,

$$\hat{A}_{i+1} = \hat{A}_i - P_i(x)K\hat{\Lambda}_i \tag{4.23}$$

where K is an (N×N) diagonal matrix, and expressed as

$$K = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & 2^{-\alpha} & & & \vdots \\ \vdots & & \ddots & & \\ 0 & \cdots & & & N^{-\alpha} \end{bmatrix} . \tag{4.24}$$

$P_i(x)$ is again given by Equation (4.21), and is inversely proportional to the power of the speech signal computed over a

duration of M samples. The denominator of $P_i(x)$ behaves as an automatic gain control which tends to equalize the adaptation rate of the prediction algorithm according to the variations in the mean square value, σ_x^2,of the speech sequence computed over the M immediate past samples. The effect of the value of M on the estimation accuracy of the predictor will be discussed in Section 4.7.

The term $k^{-\alpha}$ with $\alpha<1$ provides a smaller modification to the higher order prediction coefficients than to the lower coefficients. This is in agreement with experimental observations which support the importance, in the performance of the algorithm,of the first few prediction coefficients.

The most important factor in controlling the performance of SGEP is $\Lambda_{i,k}$, see Equation (4.22), and is determined for each coefficient. The value of $\Lambda_{i,k}$ used in updating the k^{th} coefficient, at the i^{th} sampling instant, is based on two estimates of the prediction error criterion, FU. For each coefficient, $\Lambda_{i,k}$ is given by

$$\Lambda_{i,k} = FU_{i,2k-1} - FU_{i,2k} \tag{4.25}$$

while the two error functions, FU are formed as follows:

For a particular coefficient a_k its value at the $(i+1)^{th}$ instant, $a_{i+1,k}$ is equal to its previous value minus $\Lambda_{i,k}$ multiplied by a constant. Consider that the first coefficient $a_{i,1}$ in the coefficient vector,

$$\hat{A}_i = [a_{i,1}, a_{i,2}, \ldots, a_{i,N}]^T$$

is increased by a positive number $s_{i,1}$, calculated from

$$s_{i,k} = \frac{1}{Dk^{\beta}} \tag{4.26}$$

where D and β are constants, and D>1, β<1. Having the value of $a_{i,1}$ modified by $s_{i,1}$, a predicted value $y_{i,1}$ of the input sample x_i is obtained. $a_{i,1}$ is then decreased by the same amount, $s_{i,1}$ and another predicted output $y_{i,2}$ is obtained. In the same way, when $a_{i,2}$ in $\hat{A}_i$ is modified by $\mp s_{i,2}$, the $y_{i,3}$ and $y_{i,4}$ estimates of x_i are obtained.

The process of sequentially modifying the term of $\hat{A}_i$ by $\mp s_{i,k}$, k=1,2,...,N, continues resulting in a 2N component predictive sequence $\{y_k\}$. The prediction error between the input sample x_i and each of the 2N predicted values in $\{y_k\}$ is then determined.

The error criterion must now be introduced, i.e., an appropriate FU function must be selected and used to form $A_{i,k}$ according to Equation (4.25). Two error functions were considered in our SGEP investigations and they are:

A. <u>The Absolute Error Criterion</u>: when the absolute value of the prediction error is selected as the error function, i.e.,

$$FU_i = |e_i| \tag{4.27}$$

a sequence $\{FU_{i,k}\}$ having 2N components is formed. That is

$$FU_{i,1} = |x_i - y_{i,1}|$$

$$FU_{i,2} = |x_i - y_{i,2}|$$

⋮ ⋮

$$\begin{aligned} &\vdots \qquad\qquad \vdots \\ FU_{i,2N-1} &= |x_i - y_{i,2N-1}| \\ FU_{i,2N} &= |x_i - y_{i,2N}| \end{aligned} \tag{4.28}$$

B. Mean Square Error Criterion: In this case, the Equation of the error function, FU takes the form

$$\{FU_{i,k}\} = \{e^2_{i,k}\} \tag{4.25}$$

and therefore the 2N components of $\{FU_{i,k}\}$ are

$$\begin{aligned} FU_{i,1} &= (x_i - y_{i,1})^2 \\ FU_{i,2} &= (x_i - y_{i,2})^2 \\ &\vdots \qquad\qquad \vdots \\ FU_{i,2N-1} &= (x_i - y_{i,2N-1})^2 \\ FU_{i,2N} &= (x_i - y_{i,2N})^2 \end{aligned} \tag{4.30}$$

Both error criteria were initially utilized in our experiments. It was found however, that the SNR performance of the SGEP predictor was almost unaffected by using $FU=|e_i|$ or $FU=e_i^2$ and for simplicity it was decided to employ the absolute error criterion throughout the thesis.

Once the $\{FU_{i,k}\}$ sequence is defined the $\Lambda_{i,k}$ elements required for the adaptation of the prediction coefficients are formed using Equation (4.25) and in vector form,

$$\hat{\Lambda}_i = \begin{bmatrix} FU_{i,1} - FU_{i,2} \\ FU_{i,3} - FU_{i,4} \\ \vdots \\ FU_{i,2k-1} - FU_{i,2k} \\ \vdots \\ FU_{i,2N-1} - FU_{i,2N} \end{bmatrix} \tag{4.31}$$

i.e., each element of the $\hat{\Lambda}_i$ vector is the difference between two prediction errors obtained by changing each coefficient first in a positive and then in the negative direction. Now if $FU_{i,2k-1} > FU_{i,2k}$, $y_{i,2k}$ is a better prediction than $y_{i,2k-1}$, see Equation (4.28), $\Lambda_{i,k} > 0$ and consequently $a_{i+1,k}$ should be less than $a_{i,k}$ since, by subtracting a positive constant $s_{i,k}$ from $a_{i,k}$ a better prediction is achieved. In the same way when $FU_{i,2k-1} < FU_{i,2k}$, $a_{i+1,k}$ is increased as $\Lambda_{i,k} < 0$. Thus $\Lambda_{i,k}$ gives the correct direction for the modification in the values of the prediction coefficients, while the actual amount by which the coefficients are incremented or decremented is provided by the term

$$(FU_{i,2k-1} - FU_{i,2k}) P_i(x) k^{-\alpha} . \tag{4.32}$$

After this discussion and having in mind that the coefficients of the proposed linear predictor are Sequentially Updated every sampling instant with the value of each coefficient changed in the correct direction to minimize FU, the reason for naming this scheme SGEP is apparent.

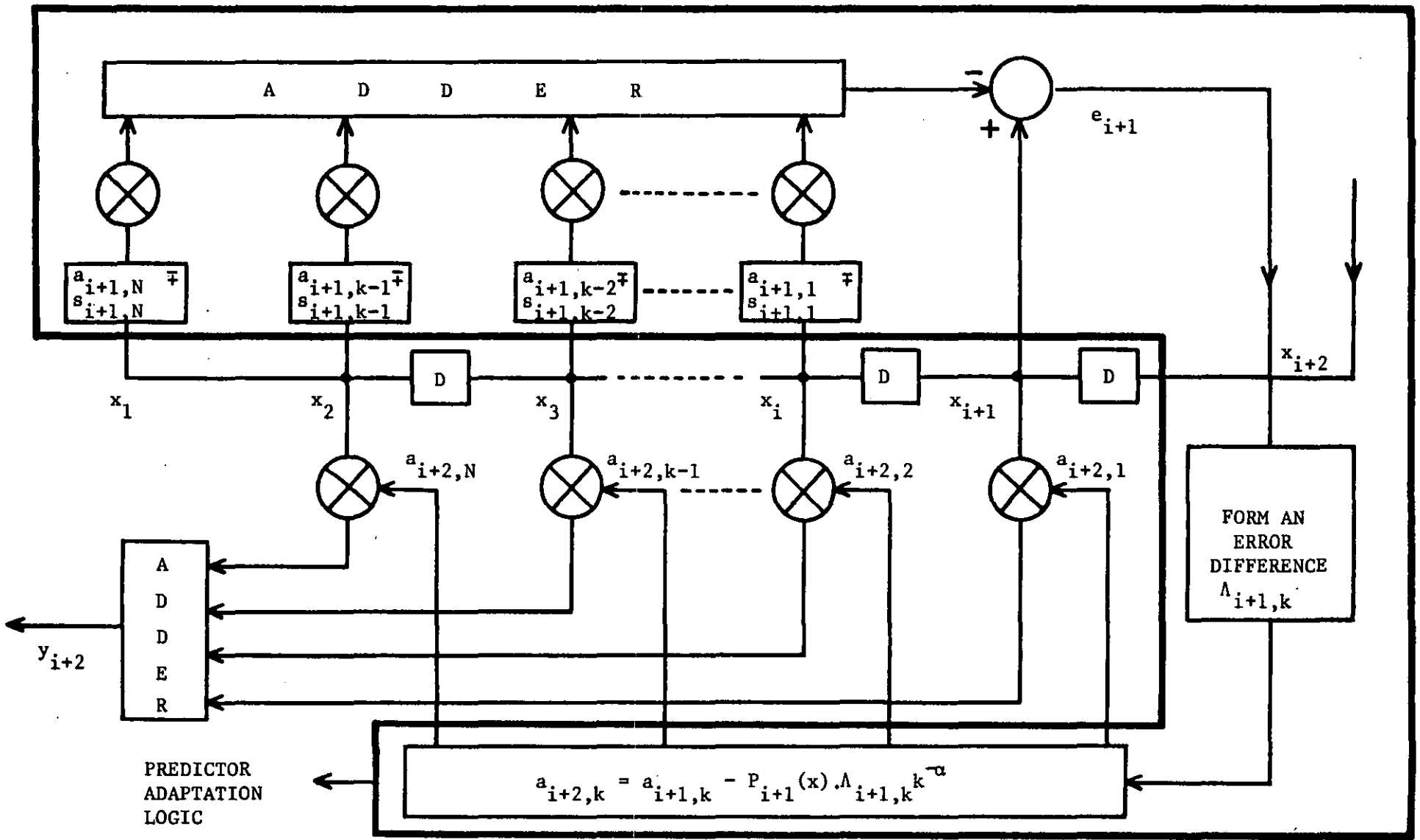


FIGURE 4.3: Operation of SGEP Algorithm at $(i+2)^{th}$ Instant

The block diagram of the SGEP algorithm is presented in Figure 4.3.

4.5.1 Operation of a 4^{th} Order SGEP Predictor

In the preceeding section, the SGEP algorithm was introduced. Before presenting the details of the algorithm's performance and the reasons why SGEP behaves better than the SAP algorithm, we demonstrate the operation of a simple 4^{th}-order predictor whose coefficients are updated using the SGEP procedure.

Suppose that $FU=|e_i|$ and the input sample, x_{i+2}, is to be predicted. The SGEP algorithm forms 8 prediction values, $y_1, y_2, \ldots, y_8$ and 8 error estimates, FU, since two prediction estimates and two error functions are obtained from each modified prediction coefficient. Hence,

$$
\begin{aligned}
FU_{i+1,1} &= |x_{i+1} - \overline{a_{i+1,1} + s_1} x_i - a_{i+1,2} x_{i-1} - a_{i+1,3} x_{i-2} - a_{i+1,4} x_{i-3}| \\
FU_{i+1,2} &= |x_{i+1} - \overline{a_{i+1,1} - s_1} x_i - a_{i+1,2} x_{i-1} - a_{i+1,3} x_{i-2} - a_{i+1,4} x_{i-3}| \\
FU_{i+1,3} &= |x_{i+1} - a_{i+1,1} x_i - \overline{a_{i+1,2} + s_2} x_{i-1} - a_{i+1,3} x_{i-2} - a_{i+1,4} x_{i-3}| \\
FU_{i+1,4} &= |x_{i+1} - a_{i+1,1} x_i - \overline{a_{i+1,2} - s_2} x_{i-1} - a_{i+1,3} x_{i-2} - a_{i+1,4} x_{i-3}| \\
FU_{i+1,5} &= |x_{i+1} - a_{i+1,1} x_i - a_{i+1,2} x_{i-1} - \overline{a_{i+1,3} + s_3} x_{i-2} - a_{i+1,4} x_{i-3}| \\
FU_{i+1,6} &= |x_{i+1} - a_{i+1,1} x_i - a_{i+1,2} x_{i-1} - \overline{a_{i+1,3} - s_3} x_{i-2} - a_{i+1,4} x_{i-3}| \\
FU_{l+1,7} &= |x_{i+1} - a_{i+1,1} x_i - a_{i+1,2} x_{i-1} - a_{i+1,3} x_{i-2} - \overline{a_{i+1,4} + s_4} x_{i-3}| \\
FU_{i+1,8} &= |x_{i+1} - a_{i+1,1} x_i - a_{i+1,2} x_{i-1} - a_{i+1,3} x_{i-2} - \overline{a_{i+1,4} - s_4} x_{i-3}|
\end{aligned}
\tag{4.33}
$$

Once the 8 values of the prediction error function at the $(i+2)^{th}$ sampling interval are computed, it is straightforward to determine $\hat{\Lambda}_{i+1}$, i.e.,

$$\hat{\Lambda}_{i+1} = \begin{bmatrix} FU_{i+1,1} - FU_{i+1,2} \\ FU_{i+1,3} - FU_{i+1,4} \\ FU_{i+1,5} - FU_{i+1,6} \\ FU_{i+1,7} - FU_{i+1,8} \end{bmatrix}$$

Then, knowing $\hat{\Lambda}_{i+1}$, Equation (4.23) is applied to yield the required $\hat{A}_{i+2}$ coefficient vector which can be used to predict the x_{i+2} input sample according to

$$y_{i+2} = \hat{A}_{i+2}^T \hat{X}_{i+2} \tag{4.34}$$

where $$\hat{X}_{i+2} = [x_{i+1}, x_i, x_{i-1}, \ldots, x_{i-N+2}]^T$$

The SGEP algorithm is computationally more complex than SAP. However in handling speech signals, SGEP performs considerably better than SAP as it will shortly be demonstrated in terms of error waveforms and SNR results when both predictors operate free of quantization noise.

The advantage, in prediction accuracy, shown by the SGEP algorithm when compared to SAP, arises as the former actually measures the gradient of the error function and updates its coefficients in the opposite direction. SAP, on the other hand, defines the gradient of the error function as $\hat{X}_i e_i$ with the assumption of $\langle e_i^2 \rangle = e_i^2$, see Equations (4.5)-(4.8) and this seems to deteriorate its performance.

4.6 COMPUTATIONAL REQUIREMENTS OF SAP, SGEP ALGORITHMS

In this section, the computational requirements of SAP and SGEP are discussed and compared with those of the Kalman algorithm. In particular, the number of multiplications and additions required by each algorithm is determined together with the relative processing time required by a digital processor to form an estimate of an input sample when using the above three prediction techniques.

4.6.1 Stochastic Approximation Prediction Algorithm

Suppose that the SAP algorithm is required to update its $\hat{A}_{i+1}$ vector in order to predict x_{i+2}. Initially, $P_{i+1}(x)$ must be computed. The denominator of $P_{i+1}(x)$ in Equation (4.21) is analysed as follows: Suppose M=N, then $\sum_j x_j^2$ requires N multiplications and (N-1) additions. The summation, however, is normalized by a factor N and assuming that the division is computationally equivalent to two multiplications, the number of multiplications becomes (N+2). Further, as the constant B is added to $1/N \sum_j x_j^2$, the number of additions required becomes equal to N. Hence, the computation of the denominator of $P_{i+1}(x)$ requires (N+2) multiplications and N additions. To complete the computation of $P_{i+1}(x)$, one more scalar division is required so a total of (N+4) multiplications and N additions are required to compute $P_{i+1}(x)$.

For the prediction of x_{i+2} in the algorithm, $\hat{A}_{i+2}$ also requires the definition of e_{i+1} which requires N multiplications and N additions

since from Equation (4.34)

$$e_{i+1} = x_{i+1} - \hat{A}_{i+1}^T \hat{X}_{i+1}$$

where

$$\hat{X}_{i+1} = [x_i, x_{i-1}, \ldots, x_{i-N+1}]^T .$$

The term e_{i+1} is then multiplied by $P_{i+1}(x)$ so that a total of (2N+5) multiplications and 2N additions are required for the computation of $P_{i+1}(x).e_{i+1}$. Finally, $P_{i+1}(x).\hat{X}_i$ is formed and $\hat{A}_{i+1}$ is updated in accordance with Equation (4.8) so that the total number of multiplications is (3N+5) while the number of additions is equal to 2N. The above procedure is summarized in Table 4.1.

4.6.2 Sequential Gradient Estimation Predictor Algorithm

Let us suppose that the estimate of x_{i+2} is to be formed using the SGEP algorithm. The $P_{i+1}(x)$ term, see Equation (4.22) is computed as in SAP algorithm and thus (N+4) multiplications and N additions per sample are required.

We can now proceed to determine the computational requirements of $\hat{A}_{i+1,k}$. As it has been shown in Section 4.6, for an N order predictor we must compute 2N components of the error function, FU in order to form the N components of the $\hat{A}_{i+1}$ vector. The term 2N arises since for each coefficient value, the algorithm introduces a positive and negative increment, $\mp s_{i+1,k}$. Note that the values of $s_{i+1,k}$ are

SAP ALGORITHM FOR N ORDER PREDICTOR	NUMBER OF MULTIPLICATIONS PER SAMPLE	NUMBER OF ADDITIONS PER SAMPLE
$P_{i+1}(x)=A/B+\frac{1}{N}\sum_j x_j^2$	N+4	N
$e_{i+1}=x_{i+1}-\hat{A}_{i+1}^T\hat{X}_{i+1}$	N	N
$P_{i+1}(x).e_{i+1}$	1	0
$P_{i+1}(x).e_{i+1}.\hat{X}_i$	N	0
$\hat{A}_{i+2}=\hat{A}_{i+1}+\ldots$	0	N
TOTAL	3N+5	3N

TABLE 4.1: Computational Requirements/Sample for SAP Algorithm

known and fixed. A simplification in forming the $\hat{\Lambda}_{i+1}$ vector comes from the fact that each component of $FU_{i+1,j}$, where j=1,2,...,2N, consists of a fixed and variable term. Let us consider, for example, the case where N=4:

$$FU_{i+1,1} = |x_{i+1} - \overline{a_{i+1,1} + s_{i+1,1}x_i} - a_{i+1,2}x_{i-1} - a_{i+1,3}x_{i-2} - a_{i+1,4}x_{i-3}| \tag{4.35}$$

or

$$FU_{i+1,1} = |\overbrace{x_{i+1} - a_{i+1,1}x_i - a_{i+1,2}x_{i-1} - a_{i+1,3}x_{i-2} - a_{i+1,4}x_{i-3}}^{\text{FIXED TERM, } e_{i+1}} \underbrace{- s_{i+1,1}x_i}_{\substack{\text{VARIABLE TERM} \\ \text{for } j=1,3,\ldots,(2N-1)}}| \tag{4.36}$$

Hence, the 2N components of the error function can be defined as,

$$\begin{aligned} FU_{i+1,1} &= |e_{i+1} - s_{i+1,1}x_i| \\ FU_{i+1,2} &= |e_{i+1} + s_{i+1,1}x_i| \\ &\vdots \qquad\qquad \vdots \\ FU_{i+1,2N-1} &= |e_{i+1} - s_{i+1,N}x_i| \\ FU_{i+1,2N} &= |e_{i+1} + s_{i+1,N}x_i| \end{aligned} \tag{4.37}$$

The computation of e_{i+1} requires N multiplications and N additions and it is performed only once every sampling instant. To determine $\Lambda_{i+1,1}$ say, we multiply $s_{i+1,1}$ by x_i and first add this value to e_{i+1} and then subtract it from e_{i+1}. This means that for each element of $\hat{\Lambda}_{i+1}$ ($\Lambda_{i+1,k}$ is the k^{th} element), we do 3 additions and 1 multiplication

as well as the N additions and N multiplications of e_{i+1}. However, 3 additions and 1 multiplication are repeated for each element of the $\hat{\Lambda}_{i+1}$ vector and therefore the calculation of $\hat{\Lambda}_{i+1}$ requires 3N additions and N multiplications together with N multiplications and N additions for e_{i+1}, totalling 2N multiplications and 4N additions.

Further, $\hat{\Lambda}_{i+1}$ is multiplied by $P_{i+1}(x)$ and matrix, K as defined by Equation (4.24). Consequently, the multiplication of $P_{i+1}(x)$ with every term of $\hat{\Lambda}_{i+1}$ necessitates N more multiplications. Similarly, K also requires extra N multiplications resulting in 2N more multiplications to form $P_{i+1}(x).K.\hat{\Lambda}_{i+1}$. As a result, $P_{i+1}(x).K.\hat{\Lambda}_{i+1}$ requires 4N multiplications and 4N additions in addition to (N+4) multiplications and N additions for the $P_{i+1}(x)$ term. Thus a total of (N+4+4N)=(5N+4) multiplications and (N+4N)=5N additions is necessary in order to finalize the calculation of the term $P_{i+1}(x).K.\hat{\Lambda}_{i+1}$.

To complete the computation of $\hat{A}_{i+2}$, we need the final N additions to form

$$\hat{A}_{i+2} = \hat{A}_{i+1} - P_{i+1}(x).K.\hat{\Lambda}_{i+1}$$

and hence the number of additions becomes 6N.

Table 4.2 summarizes the computational requirements of the SGEP algorithm.

The comparison between the SAP, SGEP and Kalman[113-116] algorithms, in terms of multiplications additions and processing time per sample based on the ICL 1900 computer, is shown in Table 4.3.

SGEP ALGORITHM FOR N ORDER	NUMBER OF MULTIPLICATIONS PER SAMPLE	NUMBER OF ADDITIONS PER SAMPLE
$P_{i+1}(x)=A/B+\frac{1}{N}\sum_j x_j^2$	N+4	N
$\overbrace{\lvert e_{i+1}^{-s_{i+1,k}x_i}\rvert-\lvert e_{i+1}^{+s_{i+1,k}x_i}\rvert}^{\Lambda_{i+1,k}}$	N+1	N+3
$\hat{\Lambda}_{i+1}=[\Lambda_{i+1,1},\Lambda_{i+1,2},\ldots,\Lambda_{i+1,N}]^T$	$\underbrace{N+1.N}_{2N}$	$\underbrace{N+3.N}_{4N}$
$\hat{\Lambda}_{i+1}.P_{i+1}(x)$	N	0
$P_{i+1}(x).K.\hat{\Lambda}_{i+1}$	N	0
$\hat{A}_{i+2}=\hat{A}_{i+1}-P_{i+1}(x)K\hat{\Lambda}_{i+1}$	0	N
TOTAL	5N+4	6N

TABLE 4.2: Computational Requirement/Sample for SGEP Algorithm

FUNCTION	PREDICTOR TYPES			
	GENERAL KALMAN	SIMPLIFIED KALMAN	SAP	SGEP
Multiplications per Sample	$2N^3+3N^2+3N$	$2N^2+4N+1$	3N+5	5N+4
Additions per Sample	$2N^3+3N^2+1$	$2N^2+2$	3N	6N
Multiplications per Sample for N=8	1240	161	29	44
Additions per Sample for N=8	1217	144	24	48
TOTAL TIME (μsec.) REQUIRED/SAMPLE	7394	932	164	272

TABLE 4.3: Comparisons Between Sequential Prediction Algorithms

For N=8, SAP produces the lowest values, and SGEP requires considerably less computational time than the modified Kalman predictor.[113] Table 4.3 is obtained by using the typical computation times of 4 μsecs./multiplication and 2 μsecs./addition of an ICL 1900 processor, resulting in 7394, 932, 164 and 272 μsecs. for N=8 order General Kalman, Simplified Kalman, SAP and SGEP predictors respectively. The corresponding numbers for the second order predictor are 194, 88, 56 and 80 μsecs.

4.7 SIMULATION RESULTS OF ISOLATED SGEP, SAP AND FIRST ORDER (LEAKY) PREDICTORS FOR SPEECH SIGNALS

The sequential gradient estimation predictor SGEP was evaluated using computer simulation. Speech signals band-limited to 3.4 kHz and sampled at 10 kHz provided the input sequence to the predictor. In order to test the effectiveness of the proposed predictor, SGEP, we first compared its error sequence, $\{e_i\}=\{x_i-y_i\}$ against that of the predictor which updates its coefficients using the stochastic approximation method.

The order of predictor was initially selected as N=15. N is an important factor in spectral modelling of speech. As discussed in Chapter II, Section 2.2.5, in general, the minimum number of coefficients is twice the number of formants to be considered.[31] Further from the references (11) and (26), in order to represent the vocal tract adequately, the memory of the predictor must be equal to at least twice the time required for sound waves to travel from the glottis to the lips that is 2×17/34000=1 msec. Consequently, for a f_s, sampling frequency, of 10 kHz, i.e., the sampling period is 0.1 msec., the memory or order of the predictor must be at least 10 for the modelling of vocal tract. However, if the glottal and lip radiation characteristics are accounted for in the model, the above mentioned N, equal to f_s, is usually considered as lower limit for N and 4 or 5 more poles are added to the model.

In summary, for adequate modelling of voiced speech, the order of

predictor should be at least equal to the sampling frequency in kHz. This choice of N can also be taken as an upper bound for the analysis of unvoiced speech. In relation to these upper limits of N, Atal and Hanauer[31] demonstrated that the mean square prediction error for either voiced or unvoiced speech remains almost unchanged when $N \geq 14$ at f_s=10 kHz.

In our experiments, the other parameters of SGEP were A=24, $B=7\times10^{-3}$, $M=10^2$, D=6, $\alpha/\beta=16$, $\beta=1/17$, see Equations (4.22)-(4.26) while SAP had a different value of A ($=10^{-4}$) than SGEP. As noticed here, too many parameters, in addition to N, play an important role in the adaptation rate of SGEP.

The fixed vector, $\hat{S}=[s_1,s_2,\ldots,s_{15}]$ whose elements are given by Equation (4.26), are calculated using the constants D and β. Figure 4.4 presents the variation of these elements with the optimum values of D and β, as quoted above. We have found that the exponential-like decay in Figure 4.4 produces a better performance in coefficient adaptation where a large variation in the lower coefficients is more significant than that in the higher order coefficients.

SNR values of SGEP, SAP and first order predictors, obtained from computer simulation experiments, for two different sets of speech data, at f_s=10 kHz, are illustrated in Table 4.4.

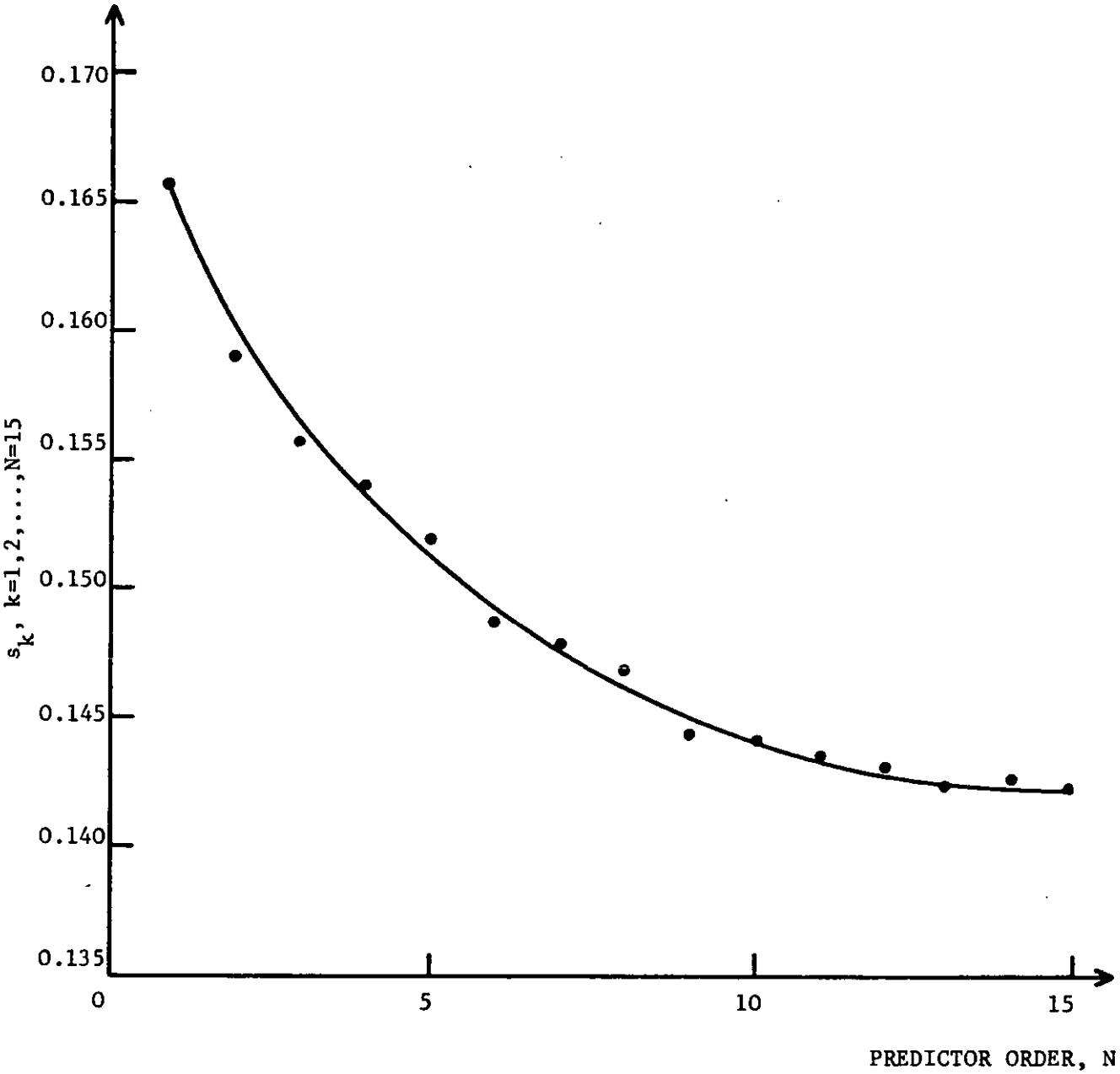


FIGURE 4.4: Variation of s_k with k=1,2,...,N

TYPE OF PREDICTION ALGORITHM	SNR in dB	
	DATA I	DATA II
First Order Predictor (Leaky Integrator a_1=0.94)	8.95	11.92
SAP (N=15)	12.67	13.04
SGEP (N=15)	15.99	17.93

TABLE 4.4: Typical SNR Values of SGEP, SAP and Leaky for Two sets of Data

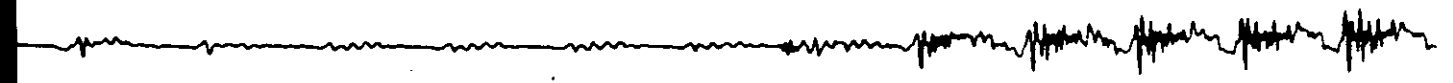
It can be seen that the prediction error power for SGEP is typically 3-4 dB lower than that obtained using the stochastic approximation predictor SAP. Figure 4.5 shows an arbitrary voiced speech segment, and the prediction error waveforms of a first order predictor having a coefficient of 0.94, a SAP predictor and finally SGEP. The prediction error of these predictors for another segment of speech containing unvoiced and voiced parts, is shown in Figure 4.6. In both Figures 4.5 and 4.6, SGEP has the smallest prediction error.

Figures 4.7 and 4.8 show the variation of SNR as a function of input speech power for SAP and SGEP. The performance of both prediction algorithms depends on the value of M, i.e. the number of samples used to estimate the power of the input signal, see Equation (4.21) with N=M.

(i)

(ii)

(iii)



(iv)

FIGURE 4.5: Prediction Error Waveforms for DATA I
(i) Input, (ii) Leaky (a_1=0.94),
(iii) SAP, (iv) SGEP, N=15

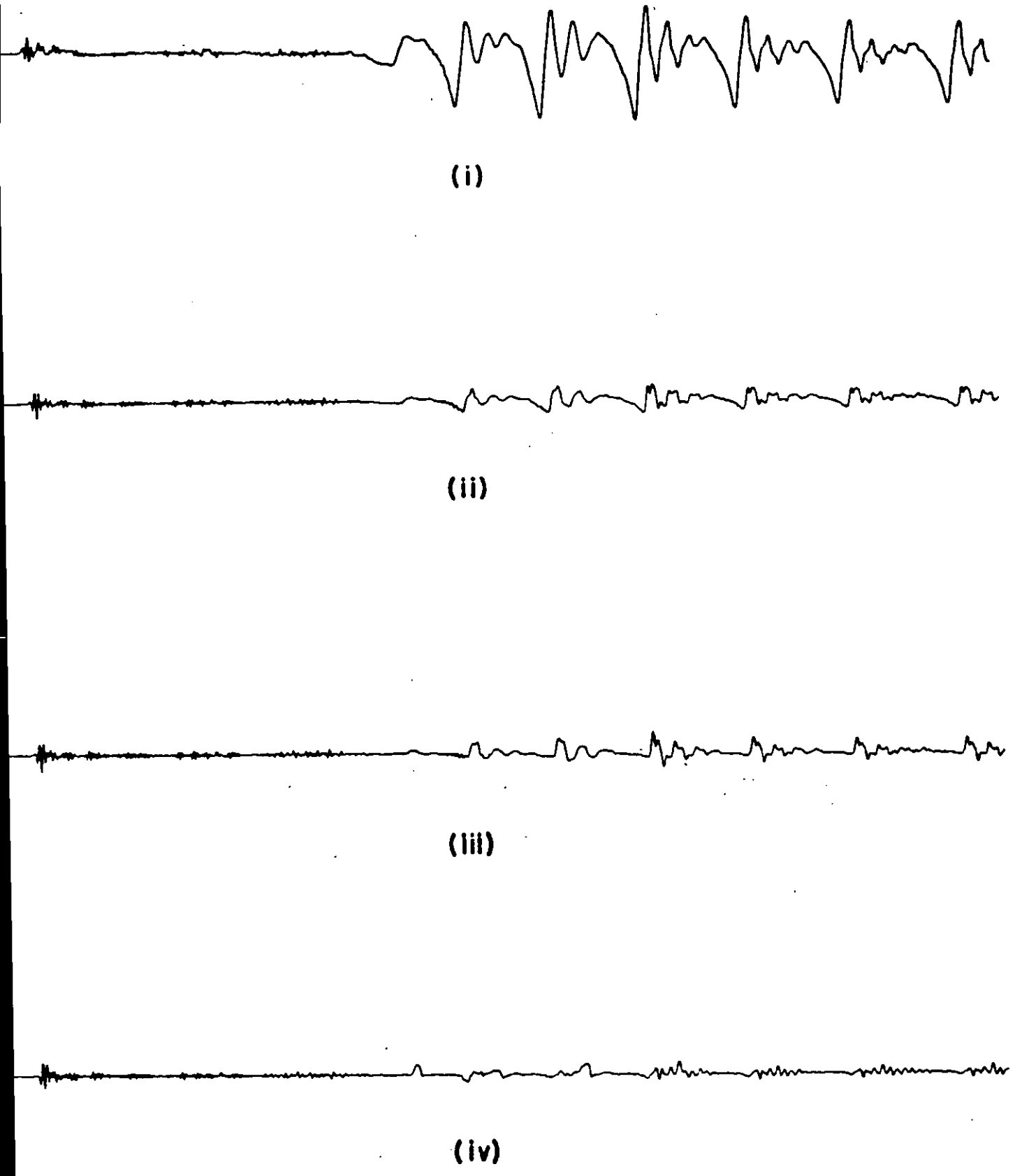


FIGURE 4.6: Prediction Error Waveforms for DATA II
(i) Input, (ii) Leaky (a_1=0.94),
(iii) SAP, (iv) SGEP, N=15

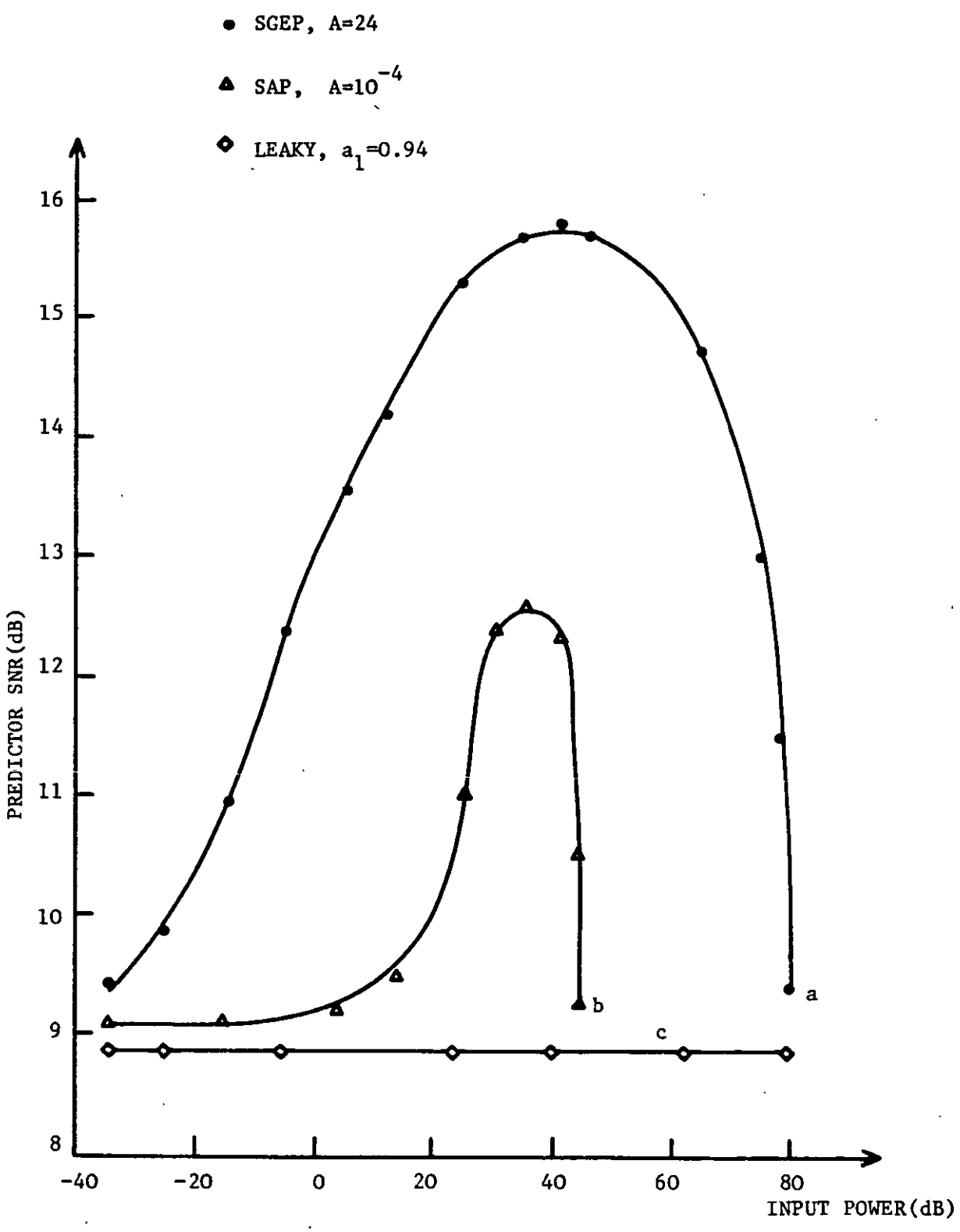


FIGURE 4.7: SNR vs. Input Power, $M=M_p=100$

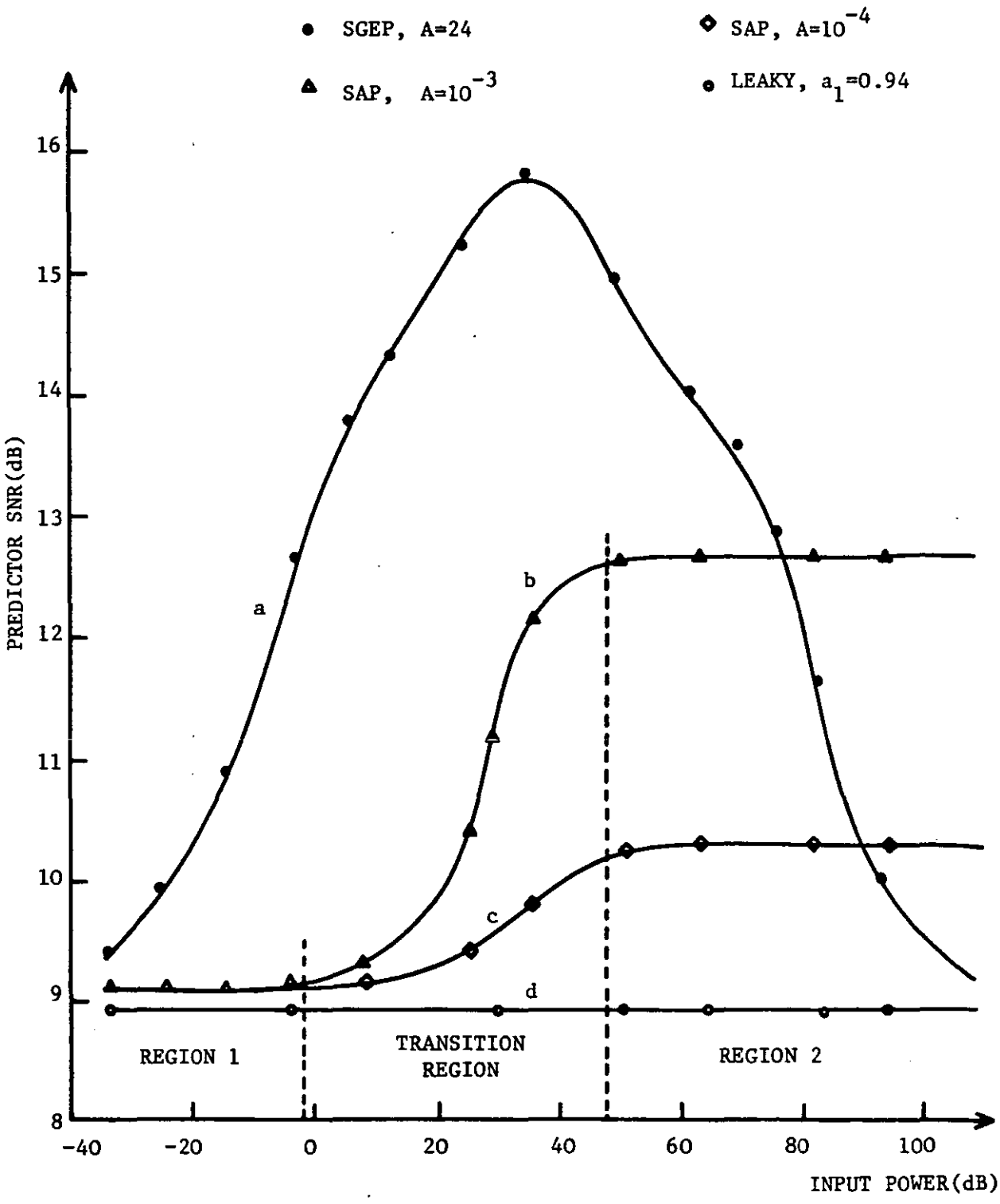


FIGURE 4.8: SNR vs. Input Power, M=N=15

For $M=M_p=100$, the power of the speech signal is measured over a duration commensurate with a typical pitch period, M_p, and this results in SAP having a restricted dynamic range as shown in Figure 4.7. This value of M is satisfactory for SGEP which has a significant SNR improvement compared to SAP. When the value of M is reduced and made equal to N, the order of the predictor, and the constant A remains at 10^{-4}, the SNR for SAP is reduced by 2 dB, but the dynamic range is improved, see Figure 4.8. When the value of A is again optimized to 10^{-3}, SAP increases its peak SNR to approximately that value obtained when $M=M_p=100$. The effect of reducing M from 100 to 15 in SGEP is to cause some deterioration in the SNR curves.

When the power of the speech signal, computed over M samples, is small compared to the value of B used in Equation (4.21) for both SAP and SGEP, the denominator of $P_i(x)$ is approximately equal to B and the predictions are less accurate. This situation is worse for SAP algorithms as shown in Figure 4.8, see curve (b) in REGION 1. However as we increase the power of the speech signal, we notice that SNR of SAP experiences a TRANSITION REGION and then remains unchanged for the high power of the speech signal, see REGION 2. The cause of this behaviour can be explained as follows:

If the $P_i(x)$ term of SAP is constant, then the algorithm, $\hat{A}_{i+1}=\hat{A}_i+P_i(x)e_i\hat{X}_i$ becomes signal dependent. In other words, reasonable prediction results are obtained only over a restricted range of input power levels. In order to improve the scheme and make it independent

of the speech signal, first $P_i(x)$ term is expressed as an inverse function of the input power, i.e., $P_i(x)=A/\frac{1}{N}\sum_{j=i-1-N}^{i-1} x_j^2$. The resulting SNR values turn out to be almost constant having a value of REGION 3, 12.67 dB, over the wide range of speech power.

When, however, unvoiced speech or silence intervals occur, the $P_i(x)$ approaches infinity since $\langle x_j^2 \rangle$ goes to zero. That is the reason why the constant B in the denominator is introduced. Thus B maintains a finite value for $P_i(x)$ during the lower power levels of speech signal, and prevents the occurrence of large prediction errors.

The low SNR value in REGION 1 is therefore due to the effect of the constant B. That is, at low power levels, $P_i(x).e_i.\hat{X}_i$ remains almost at zero, hence $\hat{A}_{i+1}$ approaches $\hat{A}_i$ and SAP behaves as a leaky integrator provided that the initial vector is in the form of $\hat{A}_0=[a_1,0,0,\ldots,0]^T$.

For the power levels of the TRANSITION REGION, $P_i(x).e_i.\hat{X}_i$ is varying considerably as the predictor is trying to adapt to the characteristics of the input signal. As soon as curve (b) approaches REGION 2, an observation of the values of $P_i(x).e_i.\hat{X}_i$ reveals that while $\langle P_i(x) \rangle$ is reduced, and both $\langle e_i \rangle$ and $\langle \hat{X}_i \rangle$ are increased, hence $\langle P_i(x).e_i.\hat{X}_i \rangle$ is nearly constant. Consequently, the adaptation rate of the SAP algorithm remains constant, at high power levels of speech.

Figures 4.9 (a,b) and 4.10(a,b) illustrate a segment of the error waveform, $\{e_i\}=\{x_i-y_i\}$ and the corresponding power spectral when SAP

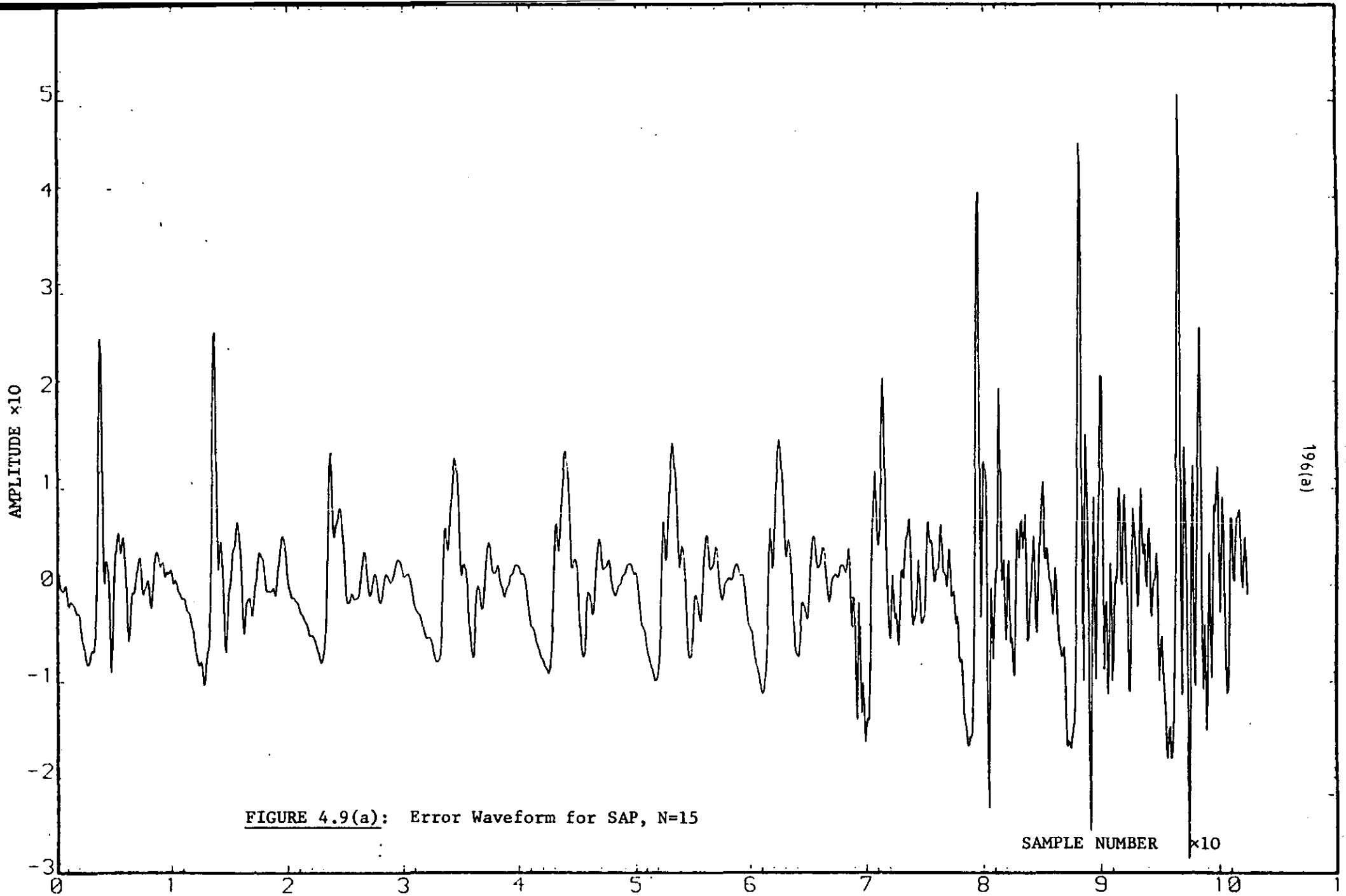


FIGURE 4.9(a): Error Waveform for SAP, N=15

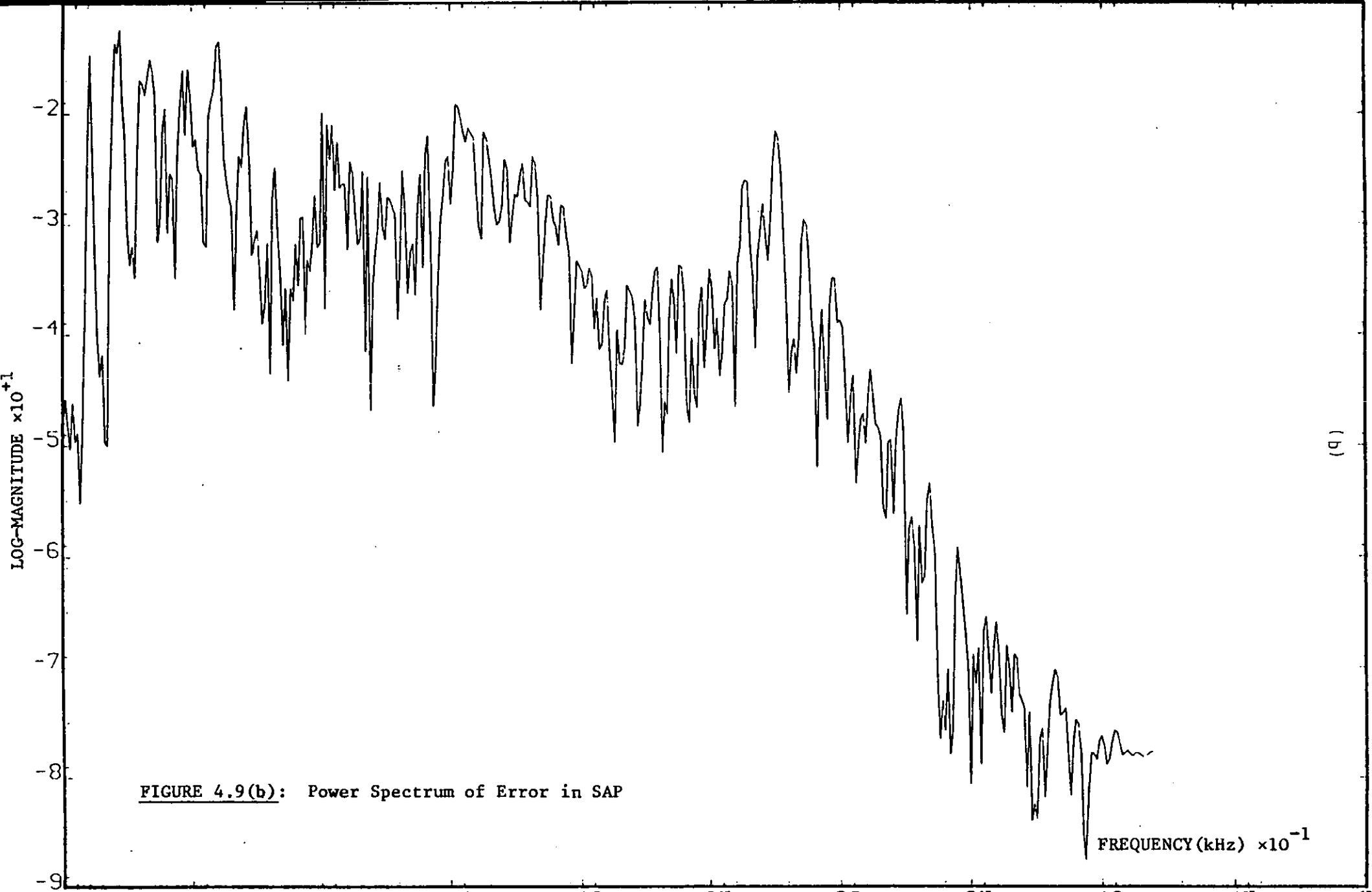


FIGURE 4.9(b): Power Spectrum of Error in SAP

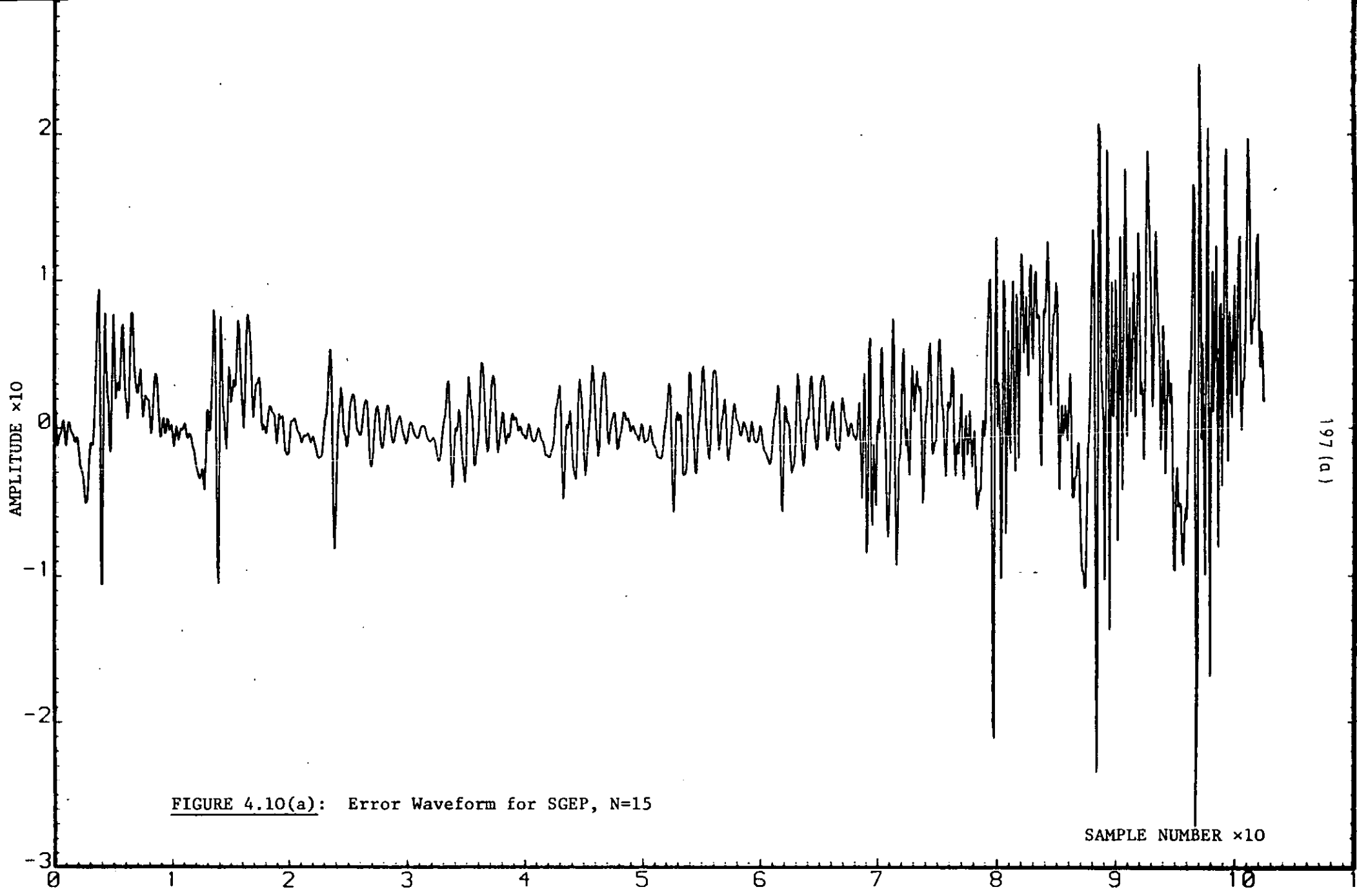


FIGURE 4.10(a): Error Waveform for SGEP, N=15

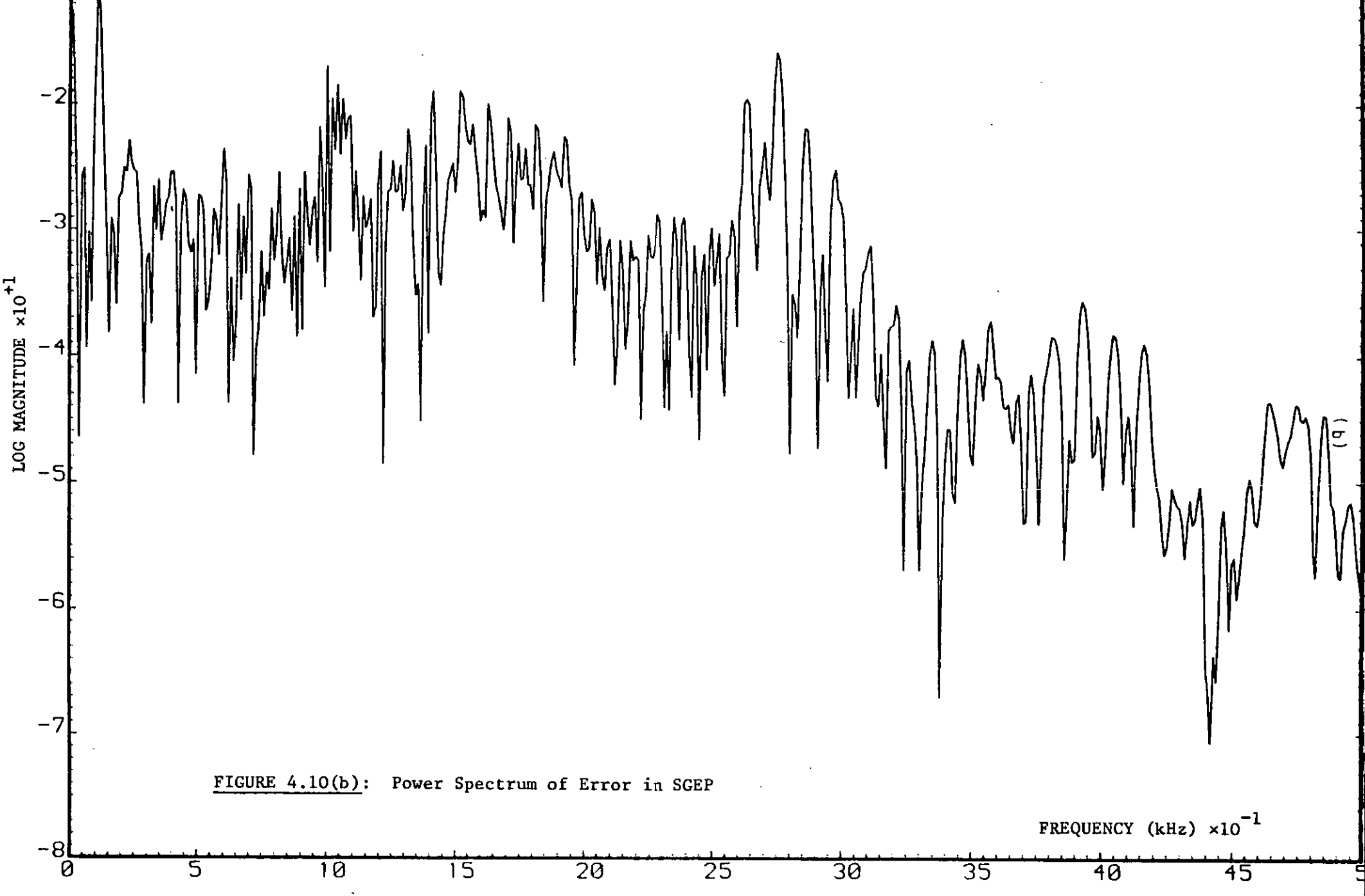


FIGURE 4.10(b): Power Spectrum of Error in SGEP

and SGEP are used, respectively. These are obtained at the peak SNR values of Figure 4.8(a,b) and N=15. These figures while demonstrating the ability of SGEP to provide an error signal whose power is 3-4 dB's less than that of SAP, they also show the advantage of SGEP in removing redundancy, i.e., in decorrelating the error signal and producing a flatter power spectrum.

The variation of SNR as a function of predictor order, N, is shown in Figure 4.11 (M=N). The SNR for SAP, is always lower than that of SGEP and falls off rapidly at lower values of N. Even with N=2, SGEP can be operated satisfactorily. Note that the gain in SGEP's SNR when N varies from 2 to 15 is only 0.40 dB. Thus the ability of SGEP to operate with N=2 reduces significantly its complexity and processing time.

4.8 NOTE ON PUBLICATION[116]

A paper entitled *"Sequential Gradient Estimation Predictor for Speech Signals"*, in co-authorship with Dr. R. Steele and Dr. C.S. Xydeas, has been published in the IEEE, International Conference Proceedings on Acoustic, Speech and Signal processing, ICASSP 79, pp.723-726, Washington D.C., April 1979.

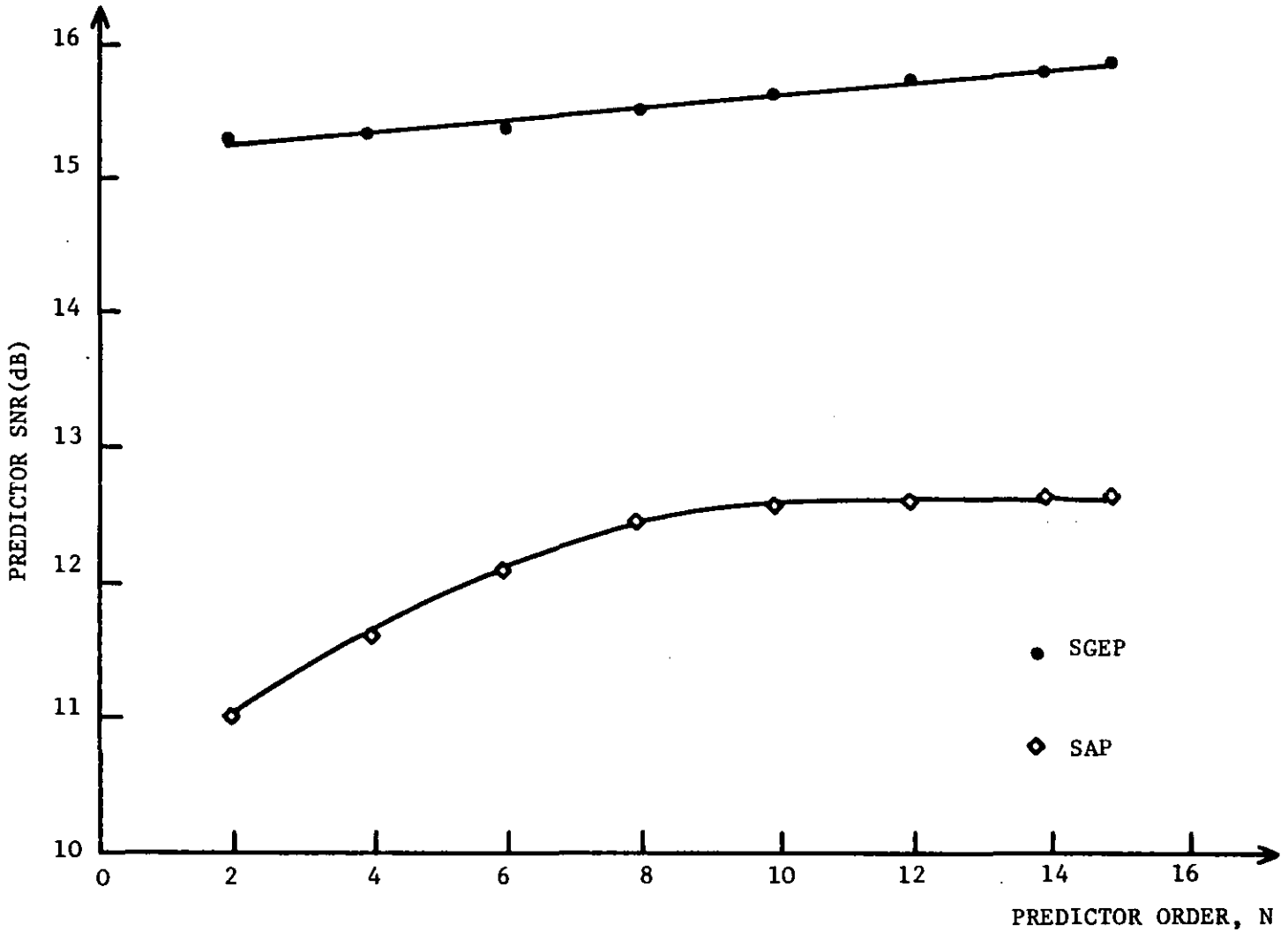


FIGURE 4.11: Variation of Predictor SNR as a Function of Predictor Order, N and M=N

4.9 CONVERGENCE OF THE "SAP" and "SGEP" ALGORITHMS

In this section the sequentially updated SAP and SGEP predictors are compared with a modified autocorrelation predictor (MAP). MAP is defined by considering a sliding-block autocorrelation predictor (SBAP) whose coefficients are re-calculated from a block of W samples every sampling instant using the autocorrelation method. The term sliding-block is introduced, because the block moves forward one sample every sampling period, i.e., there is one new speech sample in the block every time the coefficients are recalculated. Although this (SBAP) algorithm can not be used in a practical system because of the amount of side-information that must be transmitted, it does achieve more accurate predictions than those of SAP or SGEP, and here it is used as a bench-mark. First, the mathematical analysis of the convergence of SAP and SGEP is presented and then, experimental results of the variation of the prediction coefficients with time for the SBAP algorithm, SAP and SGEP are provided.

4.9.1 Convergence of the SAP Algorithm

At the i^{th} sampling instant, SBAP has a set of coefficients, $\hat{A}_s$, and a prediction error, $e_{i,s}$, while SAP, see Equation (4.8) with represented by

$$\hat{A}_{i+1} = \hat{A}_i + Pe_i\hat{X}_i \quad . \tag{4.38}$$

Comparing Equation (4.38) with that of (4.8), it is noted that P is

the convergence parameter, $P_i(x)$, which will be assumed constant over a small number of sampling intervals. The hat (^) above the symbol means the symbol is a vector, viz:

$$\hat{A}_k^T = [a_1, a_2, a_3, \ldots, a_N] \tag{4.39}$$

$$\hat{X}_i^T = [x_{i-1}, x_{i-2}, \ldots, x_{i-N}] \tag{4.40}$$

where the raised T implies transpose of the vector.

The predicted output of Equation (4.9) can be written in vector form as:

$$y_i = \hat{A}_i^T \hat{X}_i \tag{4.41}$$

the product of a (1×N) and (N×1) matrix, and the prediction error is

$$e_i = x_i - y_i .$$

For SBAP

$$e_{i,s} = x_i - \hat{A}_s^T \hat{X}_i ,$$

and

$$e_{i,s} - e_i = -(\hat{A}_s - \hat{A}_i)^T \hat{X}_i . \tag{4.42}$$

Now, let $\hat{\gamma}_i$ be as a difference vector,

$$\hat{\gamma}_i = \hat{A}_i - \hat{A}_s \tag{4.43}$$

and write Equation (4.38) as

$$\hat{A}_{i+1} - \hat{A}_s = \hat{A}_i - \hat{A}_s + P\hat{X}_i e_i$$

or

$$\hat{\gamma}_{i+1} = \hat{\gamma}_i + P\hat{X}_i e_i . \tag{4.44}$$

Implicit in Equation (4.44) is the assumption that $\hat{A}_s$ is unchanged, and a more accurate representation than $\hat{A}_i$.

From Equations (4.42) and (4.43)

$$e_i = e_{i,s} - \hat{\gamma}_i^T \hat{X}_i$$

and substituting e_i into Equation (4.44) yields

$$\hat{\gamma}_{i+1} = \hat{\gamma}_i - P[\hat{\gamma}_i^T \hat{X}_i^2 - e_{i,s}\hat{X}_i]. \tag{4.45}$$

The norm of $\hat{\gamma}_{i+1}$ is the square root of its product, viz:

$$||\hat{\gamma}_{i+1}||^2 = <\gamma_{i+1}\gamma_{i+1}> \tag{4.46}$$

From Equation (4.45)

$$||\hat{\gamma}_{i+1}||^2 = ||\hat{\gamma}_i||^2 - 2P<\hat{\gamma}_i\hat{\gamma}_i^T\hat{X}_i^2 - e_{i,s}\hat{\gamma}_i^T\hat{X}_i> + P^2<\{(\hat{\gamma}_i^T\hat{X}_i)^2 + e_{i,s}^2 - 2\hat{\gamma}_i^T\hat{X}_i e_{i,s}\}\hat{X}_i^2> . \tag{4.47}$$

Now, P<<1, and for a sufficiently small prediction error ($e_{i,s} \to 0$),

$$||\hat{\gamma}_{i+1}||^2 \approx ||\hat{\gamma}_i||^2 - 2P<\hat{\gamma}_i\hat{\gamma}_i^T\hat{X}_i^2> \tag{4.48}$$

and as $2P<\hat{\gamma}_i\hat{\gamma}_i^T\hat{X}_i^2> = 2P<(\hat{\gamma}_i^T\hat{X}_i)^2>$, i.e., always a positive quantity,

$$||\hat{\gamma}_{i+1}||^2 < ||\gamma_i||^2 . \tag{4.49}$$

Consequently the SAP algorithm changes its prediction coefficients $\hat{A}_i$ to ensure the convergence towards $\hat{A}_s$.

The above analysis is based on the minimization of mean square error, see Equation (4.8). In a similar way, convergence of Cummiskey's[62] algorithm employing the minimization of absolute error criterion, (see Appendix E), can be proved, viz:

$$\hat{A}_{i+1} - \hat{A}_s = \hat{A}_i - \hat{A}_s + P_a \hat{X}_i \operatorname{sgn}(e_i) \tag{4.50}$$

P_a, instead of P, is used to distinguish the absolute error criterion from the m.s.e. one. Hence,

$$\hat{\gamma}_{i+1} = \hat{\gamma}_i + P_a \hat{X}_i \operatorname{sgn}(e_i) \ . \tag{4.51}$$

$\hat{A}_s$ and $\hat{\gamma}_i$ are the set of coefficients of SBAP and the difference vector at the i^{th} instant respectively. Then,

$$e_i = e_{i,s} - \hat{\gamma}_i^T \hat{X}_i \tag{4.52}$$

and substitution of Equation (4.52) in Equation (4.51) yields

$$\hat{\gamma}_{i+1} = \hat{\gamma}_i + P_a \hat{X}_i \operatorname{sgn}(e_{i,s} - \hat{\gamma}_i^T \hat{X}_i) \ . \tag{4.53}$$

The norm of Equation (4.53) is

$$||\hat{\gamma}_{i+1}||^2 = ||\hat{\gamma}_i||^2 + 2P_a \langle \hat{\gamma}_i^T \hat{X}_i \operatorname{sgn}(e_{i,s} - \hat{\gamma}_i^T \hat{X}_i) \rangle + P_a^2 \langle \hat{X}_i^2 \{\operatorname{sgn}(e_i)\}^2 \rangle \ . \tag{4.54}$$

Again, convergence depends on the value of P_a and for a sufficiently small P_a ($P_a << 1$), Equation (4.54) may be written

$$||\hat{\gamma}_{i+1}||^2 \simeq ||\hat{\gamma}_i||^2 + 2P_a \langle \hat{\gamma}_i^T \hat{X}_i \operatorname{sgn}(e_{i,s} - \hat{\gamma}_i^T \hat{X}_i) \rangle \ . \tag{4.55}$$

For $e_{i,s} < \hat{\gamma}_i^T \hat{X}_i$, the coefficients of SAP are not near optimum, Equation (4.55) becomes,

$$||\hat{\gamma}_{i+1}||^2 \simeq ||\hat{\gamma}_i||^2 + 2P_a \hat{\gamma}_i^T \hat{X}_i \operatorname{sgn}(-\hat{\gamma}_i^T \hat{X}_i) \ . \tag{4.56}$$

Since,
$$z.\operatorname{sgn}(z) = |z| \tag{4.57}$$

Equation (4.56) is rewritten as,

$$||\hat{\gamma}_{i+1}||^2 \approx ||\hat{\gamma}_i||^2 - 2P_a|\hat{\gamma}_i^T\hat{X}_i| \quad . \tag{4.58}$$

Finally, it is obvious that $2P_a|\hat{\gamma}_i^T\hat{X}_i|$ is always positive, hence

$$||\hat{\gamma}_{i+1}||^2 < ||\hat{\gamma}_i||^2 \quad . \tag{4.59}$$

Consequently, Cummiskey's algorithm, obeying the minimization of absolute error criterion, also changes its prediction coefficients $\hat{A}_i$ to ensure the convergence towards $\hat{A}_s$. Also the convergence is slowed to stop when

$$|\hat{\gamma}_i^T\hat{X}_i| \rightarrow |e_{i,s}| \tag{4.60}$$

i.e., for Equation (4.60), Equation (4.55) becomes

$$||\hat{\gamma}_{i+1}||^2 = ||\hat{\gamma}_i||^2 \quad . \tag{4.61}$$

4.9.2 Convergence of the SGEP Algorithm

The SGEP updates its coefficients according to Equation (4.22). As before, $P_i(x)$ is assumed to be a constant over a small number of sampling intervals and the optimizing term $k^{-\alpha}$ is ignored. Hence,

$$\hat{A}_{i+1} = \hat{A}_i - P\hat{\Lambda}_i \tag{4.62}$$

where $\hat{\Lambda}_i$ is the vector representation of Equation (4.25), $\hat{\gamma}_i$ is the difference vector as defined before, see Equation (4.43), and a difference matrix Γ_i as

$$\Gamma_i = A_i - A_{i,s} \tag{4.63}$$

where column matrices A_i, $A_{i,s}$ and Γ_i have N identical elements of $\hat{A}_i, \hat{A}_{i,s}$ and $\hat{\gamma}_i$ respectively.

From Section 4.5, at every sampling instant all the coefficients are up-dated. To do this, each coefficient is increased by s_k, k=1,2,...,N, see Equation (4.26), while the other coefficients are unchanged. Thus we are dealing with N vectors, $\hat{A}'_{i,k}$, k=1,2,...,N, from which N predictions are made and N errors $e'_{i,k}$, k=1,2,...,N are obtained. The procedure is repeated by subtracting s_k, k=1,2,3,...,N, from each coefficient in turn, and obtain a second set of errors $e''_{i,k}$, k=1,2,...,N. Hence,

$$\hat{A}'_{i,1} = \begin{bmatrix} a_1+s_1 \\ a_2 \\ \vdots \\ a_N \end{bmatrix} \quad , \quad \hat{A}'_{i,2} = \begin{bmatrix} a_1 \\ a_2+s_2 \\ a_3 \\ \vdots \\ a_N \end{bmatrix} \quad , \quad \hat{A}'_{i,N} = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_{N-1} \\ a_N+s_N \end{bmatrix} \tag{4.64}$$

and

$$\hat{A}''_{i,1} = \begin{bmatrix} a_1-s_1 \\ a_2 \\ \vdots \\ a_N \end{bmatrix} \quad , \quad \hat{A}''_{i,2} = \begin{bmatrix} a_1 \\ a_2-s_2 \\ a_3 \\ \vdots \\ a_N \end{bmatrix} \quad , \quad \hat{A}''_{i,N} = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_{N-1} \\ a_N-s_N \end{bmatrix} \tag{4.65}$$

The error vectors are

$$\hat{e}'_i = \begin{bmatrix} e'_{i,1} \\ e'_{i,2} \\ \vdots \\ e'_{i,N} \end{bmatrix} \quad , \quad \hat{e}''_i = \begin{bmatrix} e''_{i,1} \\ e''_{i,2} \\ \vdots \\ e''_{i,N} \end{bmatrix} \tag{4.66}$$

The elements of the vectors are

$$e'_{i,k} = x_i - \hat{A}'^{T}_{i,k}\hat{X}_i \quad , \; k=1,2,\ldots,N \tag{4.67}$$

$$e''_{i,k} = x_i - \hat{A}''^{T}_{i,k}\hat{X}_i \quad , \; k=1,2,\ldots,N. \tag{4.68}$$

Now, the two matrices are formed, viz:

$$A'_i = \begin{bmatrix} \hat{A}'^{T}_{i,1} \\ \hat{A}'^{T}_{i,2} \\ \vdots \\ \hat{A}'^{T}_{i,N} \end{bmatrix} \quad , \qquad A''_i = \begin{bmatrix} \hat{A}''^{T}_{i,1} \\ \hat{A}''^{T}_{i,2} \\ \vdots \\ \hat{A}''^{T}_{i,N} \end{bmatrix} \tag{4.69}$$

and express Equation (4.66) in the form

$$\hat{e}'_i = X_i - A'_i\hat{X}_i \tag{4.70}$$

and

$$\hat{e}''_i = X_i - A''_i\hat{X}_i \tag{4.71}$$

where

$$X_i = \begin{bmatrix} x_i \\ x_i \\ \vdots \\ x_i \end{bmatrix} \quad , \text{ and } \quad \hat{X}_i = \begin{bmatrix} x_{i-1} \\ x_{i-2} \\ \vdots \\ x_{i-N} \end{bmatrix} \tag{4.72}$$

Consider now the error vector $\hat{e}_{i,s}$ associated with SBAP,

$$\hat{e}_{i,s} = X_i - A_{i,s}\hat{X}_i \tag{4.73}$$

and

$$\hat{e}_{i,s} = \begin{bmatrix} e_{i,s} \\ e_{i,s} \\ \vdots \\ e_{i,s} \end{bmatrix} \tag{4.74}$$

Proceeding to find the error vectors, from Equations (4.70) and (4.73)

$$\hat{e}_i' - \hat{e}_{i,s} = -(A_i' - A_{i,s})\hat{X}_i \quad . \tag{4.75}$$

Now,

$$A_i' = A_i + S_i \tag{4.76}$$

where matrix S_i is given by

$$S_i = \begin{bmatrix} s_{i,1} & 0 & 0 & \cdots & 0 \\ 0 & s_{i,2} & 0 & \cdots & 0 \\ \vdots & & & & \vdots \\ 0 & \cdots & & & s_{i,N} \end{bmatrix} \tag{4.77}$$

and matrix S_i is the same as matrix S since it is a fixed, given matrix.

Substituting Equation (4.76), in Equation (4.75) and with the aid of Equation (4.63),

$$\hat{e}_i' = -(\Gamma_i + S_i)\hat{X}_i + \hat{e}_{i,s} \quad . \tag{4.78}$$

Similarly,

$$\hat{e}_i'' = -(A_i'' - A_{i,s})\hat{X}_i + \hat{e}_{i,s}$$

and because

$$A_i'' = A_i - S_i \tag{4.79}$$

$$\hat{e}_i'' = -(\Gamma_i - S_i)\hat{X}_i + \hat{e}_{i,s} \quad . \tag{4.80}$$

In order to demonstrate the inate ability of SGEP to converge faster than SAP, we first choose the same error, i.e., mean square error criterion as used in the simulation of SAP. From Equation (4.62),

$$\hat{A}_{i+1} - \hat{A}_s = \hat{A}_i - \hat{A}_s - P\hat{\Lambda}_i \tag{4.81}$$

or

$$\hat{\gamma}_{i+1} = \hat{\gamma}_i - P\hat{\Lambda}_i \tag{4.82}$$

where

$$\hat{\Lambda}_i = \begin{bmatrix} e'^2_{i,1} - e''^2_{i,1} \\ e'^2_{i,2} - e''^2_{i,2} \\ \vdots \\ e'^2_{i,N} - e''^2_{i,N} \end{bmatrix} \tag{4.83}$$

Each element in the vector $\hat{\gamma}_{i+1}$,

$$\begin{aligned} \gamma_{i+1,1} &= \gamma_{i,1} - P(e'^2_{i,1} - e''^2_{i,1}) \\ \gamma_{i+1,2} &= \gamma_{i,2} - P(e'^2_{i,2} - e''^2_{i,2}) \\ &\vdots \\ \gamma_{i+1,N} &= \gamma_{i,N} - P(e'^2_{i,N} - e''^2_{i,N}). \end{aligned} \tag{4.84}$$

The elements of $\hat{\Lambda}_i$ are found with the aid of Equations (4.78) and (4.80), viz:

$$e'_{i,k} = -(\hat{\gamma}_i^T + \hat{S}_k)\hat{X}_i + e_{i,s} \tag{4.85}$$

and

$$e''_{i,k} = -(\hat{\gamma}_i^T - \hat{S}_k)\hat{X}_i + e_{i,s} \tag{4.86}$$

where $k=1,2,\ldots,N$ and $\hat{S}_k=[0,0,\ldots,0,s_k,0,\ldots,0]$.

Thus having found

$$\gamma_{i+1,k} = \gamma_{i,k} - P(e'^2_{i,k} - e''^2_{i,k}) \qquad k=1,2,3,\ldots,N \tag{4.87}$$

and from Equations (4.85) and (4.86),

$$\begin{aligned} e'^2_{i,k} - e''^2_{i,k} &= (e'_{i,k} + e''_{i,k})(e'_{i,k} - e''_{i,k}) \\ &= 4\hat{S}_k\,\hat{\gamma}_i^T(\hat{X}_i)^2 - 4\hat{S}_k\hat{X}_i e_{i,s} \end{aligned} \tag{4.88}$$

and because the $e_{i,s}$ is small in comparison with other errors

$$e'^{2}_{i,k} - e''^{2}_{i,k} \simeq 4\hat{S}_k \hat{Y}_i^T (\hat{X}_i)^2$$

$$k=1,2,3,\ldots,N \qquad (4.89)$$

Equation (4.87) can therefore be written in vector form as

$$\underset{(N\times 1)}{\hat{Y}_{i+1}} = \underset{(N\times 1)}{\hat{Y}_i} - \underbrace{4P\hat{S}_{\hat{Y}_i}(\hat{X}_i)^2}_{(N\times 1)} . \qquad (4.90)$$

Taking the norm of Equation (4.90)

$$||\hat{Y}_{i+1}||^2 = ||\hat{Y}_i||^2 - 8P\langle \hat{Y}_i^T \hat{S}_{\hat{Y}_i}(\hat{X}_i)^2 \rangle + 16P^2 \langle \{(\hat{S}_{\hat{Y}_i})^2 (\hat{X}_i)^4\} \rangle \qquad (4.91)$$

and for P sufficiently small,

$$||\hat{Y}_{i+1}||^2 \simeq ||\hat{Y}_i||^2 - 8P\langle \hat{Y}_i^T \hat{S}_{\hat{Y}_i}(\hat{X}_i)^2 \rangle \qquad (4.92)$$

as $8P\langle \hat{Y}_i^T \hat{S}_{\hat{Y}_i}(\hat{X}_i)^2 \rangle$ is always positive,

$$||\hat{Y}_{i+1}||^2 < ||\hat{Y}_i||^2 \qquad (4.93)$$

and hence the algorithm SGEP converges.

In a similar way convergence of SGEP minimizing absolute error can be proved, viz:

Equation (4.82) still holds but the vector $\hat{\Lambda}_i$ is given by

$$\hat{\Lambda}_i = \begin{bmatrix} |e'_{i,1}| - |e''_{i,1}| \\ |e'_{i,2}| - |e''_{i,2}| \\ \vdots \\ |e'_{i,N}| - |e''_{i,N}| \end{bmatrix} \qquad (4.94)$$

Each element in the vector $\hat{Y}_{i+1}$ is

$$\begin{aligned}
\gamma_{i+1,1} &= \gamma_{i,1} - P(|e'_{i,1}| - |e''_{i,1}|) \\
\gamma_{i+1,2} &= \gamma_{i,2} - P(|e'_{i,2}| - |e''_{i,2}|) \\
&\vdots \qquad\qquad \vdots \qquad\qquad\qquad \vdots \\
\gamma_{i+1,N} &= \gamma_{i,N} - P(|e'_{i,N}| - |e''_{i,N}|).
\end{aligned} \tag{4.95}$$

The elements of $\hat{\Lambda}_i$ again are found with the aid of Equations (4.85) and (4.86), viz.

$$\gamma_{i+1,k} = \gamma_{i,k} - P\{|-(\hat{\gamma}_i^T+\hat{S}_k)\hat{X}_i + e_{i,s}| - |-(\hat{\gamma}_i^T-\hat{S}_k)\hat{X}_i + e_{i,s}|\}$$

$$k=1,2,3,\ldots,N. \tag{4.96}$$

or simply,

$$\gamma_{i+1,k} = \gamma_{i,k} - P\{|w-v| - |w+v|\} \tag{4.97}$$

where $w = -\hat{\gamma}_i^T\hat{X}_i + e_{i,s}$

$v = \hat{S}_k\hat{X}_i$.

Also from algebra,

$$|w-v| - |w+v| = \begin{cases} -2w\,\mathrm{sgn}(v), & |v|>|w| \\ -2v\,\mathrm{sgn}(w), & |w|>|v|. \end{cases} \tag{4.98}$$

Hence, Equation (4.97) is examined for the two cases as follows:

CASE I: i.e., $|\hat{S}_k\hat{X}_i| > |\hat{\gamma}_i^T\hat{X}_i + e_{i,s}|$

with the aid of Equation (4.98), the elements given by Equation (4.95) can be expressed as, $(e_{i,s} \to 0)$

$$\begin{aligned}
\gamma_{i+1,1} &\simeq \gamma_{i,1} - 2P\hat{\gamma}_i^T\hat{X}_i.\mathrm{sgn}(s_1x_{i-1}) \\
\gamma_{i+1,2} &\simeq \gamma_{i,2} - 2P\hat{\gamma}_i^T\hat{X}_i.\mathrm{sgn}(x_2x_{i-2}) \\
&\vdots \qquad\qquad \vdots \qquad\qquad\qquad \vdots \\
\gamma_{i+1,k} &\simeq \gamma_{i,k} - 2P\hat{\gamma}_i^T\hat{X}_i.\mathrm{sgn}(s_kx_{i-k}) \\
&\vdots \qquad\qquad \vdots \qquad\qquad\qquad \vdots \\
\gamma_{i+1,N} &\simeq \gamma_{i,N} - 2P\hat{\gamma}_i^T\hat{X}_i.\mathrm{sgn}(s_Nx_{i-N})
\end{aligned} \tag{4.99}$$

In vector form,

$$\underset{(N\times 1)}{\hat{\gamma}_{i+1}} \simeq \underset{(N\times 1)}{\hat{\gamma}_i} - \underbrace{2P\hat{\gamma}_i^T\hat{X}_i.\mathrm{sgn}(S\hat{X}_i)}_{(N\times 1)} \tag{4.100}$$

since the elements of matrix S are always positive, $\mathrm{sgn}(S\hat{X}_i)$ is calculated only by the sign of $\hat{X}_i$ and hence,

$$\hat{\gamma}_{i+1} \simeq \hat{\gamma}_i - 2P\hat{\gamma}_i^T\hat{X}_i.\mathrm{sgn}(\hat{X}_i) \tag{4.101}$$

For a sufficiently small P (P<<1), the norm of Equation (4.101), together with the aid of Equation (4.57) yields,

$$||\hat{\gamma}_{i+1}||^2 \simeq ||\hat{\gamma}_i||^2 - 4P(\hat{\gamma}_i)^2(\hat{\hat{X}}_i\hat{X}_i^T) \tag{4.102}$$

where

$$\hat{\hat{X}}_i = [\mathrm{sgn}(x_{i-1}),\mathrm{sgn}(x_{i-2}),\ldots,\mathrm{sgn}(x_{i-N})].$$

In Equation (4.102) as $4P(\hat{\gamma}_i^2)(\hat{\hat{X}}_i\hat{X}_i^T)$ is always positive, i.e., $\mathrm{sgn}(x_{i-1})$. $x_{i-1}=|x_{i-1}|$, and

$$||\hat{\gamma}_{i+1}||^2 < ||\hat{\gamma}_i||^2 . \tag{4.103}$$

Therefore, the algorithm satisfying the first condition, see Equation (4.98), converges.

CASE II: $\quad |-\hat{\gamma}_i^T\hat{X}_i+e_{i,s}|>|\hat{S}_k\hat{X}_i|$.

For this case, as $e_{i,s}\to 0$, the elements given by Equation (4.95) are expressed as,

$$\begin{aligned}
\gamma_{i+1,1} &\simeq \gamma_{i,1} - 2P.s_1x_{i-1}\mathrm{sgn}(\hat{\gamma}_i^T\hat{X}_i)\\
\gamma_{i+1,2} &\simeq \gamma_{i,2} - 2P.s_2x_{i-2}\mathrm{sgn}(\hat{\gamma}_i^T\hat{X}_i)\\
&\vdots\\
\gamma_{i+1,k} &\simeq \gamma_{i,k} - 2P.s_kx_{i-k}\mathrm{sgn}(\hat{\gamma}_i^T\hat{X}_i)\\
&\vdots\\
\gamma_{i+1,N} &\simeq \gamma_{i,N} - 2P.s_Nx_{i-N}\mathrm{sgn}(\hat{\gamma}_i^T\hat{X}_i) \quad , k=1,2,3,\ldots,N,
\end{aligned} \tag{4.104}$$

and in vector form,

$$\hat{\gamma}_{i+1} \simeq \hat{\gamma}_i - 2P\hat{X}_i \operatorname{sgn}(\hat{\gamma}_i^T \hat{X}_i) \tag{4.105}$$

The norm of Equation (4.105) for a sufficiently small P (P<<1) yields

$$||\hat{\gamma}_{i+1}||^2 = ||\gamma_i||^2 - 4P\left\{ \sum_{k=1}^{N} s_k \hat{\gamma}_i^T \hat{X}_i \operatorname{sgn}(\hat{\gamma}_i^T \hat{X}_i) \right\} \tag{4.106}$$

and from Equation (4.57)

$$||\hat{\gamma}_{i+1}||^2 \simeq ||\hat{\gamma}_i||^2 - 4P\left\{ \sum_{k=1}^{N} s_k |\hat{\gamma}_i^T \hat{X}_i| \right\} \tag{4.107}$$

In Equation (4.107) as $|\hat{\gamma}_i^T \hat{X}_i|$ and s_k are positive

$$||\hat{\gamma}_{i+1}||^2 < ||\hat{\gamma}_i||^2 \tag{4.108}$$

and hence, the algorithm converges when satisfying the second condition, see Equation (4.98).

4.9.3 Experimental Results for Convergence of the SAP and SGEP Algorithms

In order to support the computer simulation SNR results which indicate that SGEP converges faster than SAP towards an optimum solution, the sequentially formed prediction coefficients of the two algorithms are compared with those of the SBAP algorithm since the values of its coefficients are regarded as close to ideal. A block of 128 samples was employed to compute the SBAP coefficients and this block was shifted sequentially sample-by-sample to scan the input speech.

Consider the case when each predictor has only one coefficient a_1. Figure 4.12(a) displays a segment of speech, while (b) shows how the

SBAP coefficient adapts to the variations in the speech, particularly the large amplitude changes in the speech during the sampling intervals 200 to 600, 600 to 1100, 1100 to 2200. Although the SAP algorithm updates every sampling instant, the Figure 4.12(c) shows that, except for the initial period, it converges to the average value of the SBAP coefficient, i.e., 0.94, making only small variations about this value. This is not the case for SGEP, which is a truly instantaneously adaptive algorithm as seen in Figure 4.12(d), where its coefficient variations correspond more closely to SBAP and the statistical variations of the speech signal.

Figure 4.13 contains variations of the SBAP, SAP and SGEP predictor coefficients where each has two coefficients, and the predictions are made on the same speech signal of Figure 4.12(a). The variation of the first coefficient a_1 with the number of samples (or with time) for SBAP, SAP and SGEP is shown in Figure 4.13(a),(b) and (c) respectively. As expected, the structure of the a_1 function varies for the different segments of the input speech, i.e., in Figure 4.13(a), there are rapid variations in a_1 as the SBAP adapts to the changes in the speech signal. SAP with its poor rate of adaptation produces a maximum a_1 of 1.15, see Figure 4.13(b), after processing the entire speech segment, a value significantly below the average value of a_1=1.60 for SBAP. Like SAP, the coefficient a_1 for SGEP has an initial value of unity, but unlike the former it does not decrease significantly before its upward climb. After 400 samples, a_1 for SGEP in Figure 4.13(c) is close to the a_1 values for SBAP, although with smaller variations, and near the end of the segment it reaches an average level close to a_1 of SBAP. Similarly, the variations of the second coefficient a_2 are shown in Figure 4.13(d),

(e) and (f) for SBAP, SAP and SGEP, respectively. Once more it can be observed that the convergence of SGEP is superior to SAP. Note that the second coefficients in SBAP, SAP and SGEP have negative values as a_1 values were positive, i.e., the average values of a_1 being larger than unity are compensated by the second coefficients.

The variation of coefficient values with time for high order predictors were also examined and found consistent in the ability of SGEP to converge faster and more accurately than SAP. As an example, the variations of the first two coefficients of a fourth-order predictor employing SBAP, SAP and SGEP algorithms are presented in Figures 4.14(a)-(f) for ascending order, N.

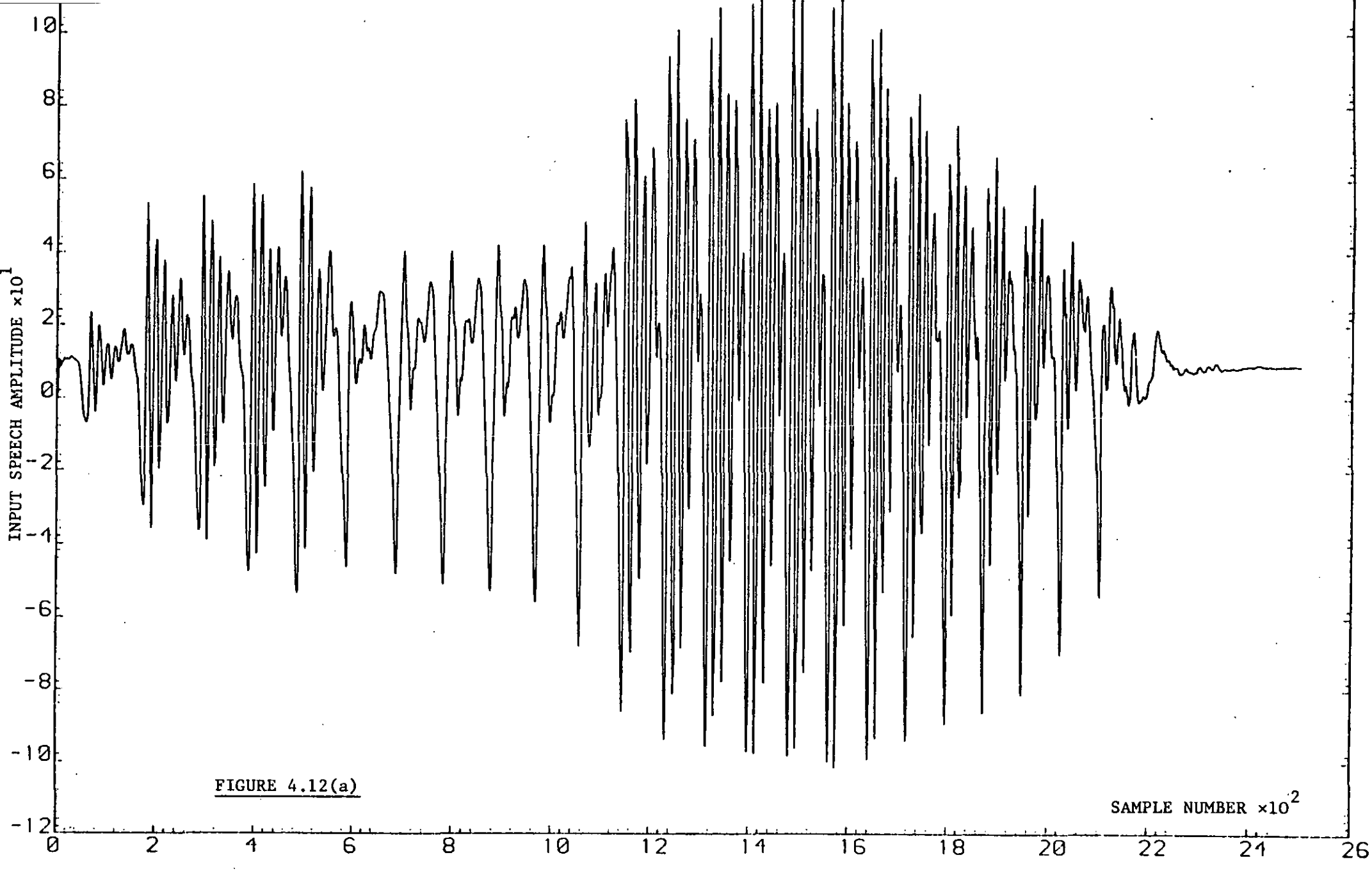

INPUT SPEECH AMPLITUDE ×10^1
10
8
6
4
2
0
-2
-4
-6
-8
-10
-12
0
2
4
6
8
10
12
14
16
18
20
22
24
26
SAMPLE NUMBER ×10^2

FIGURE 4.12(a)

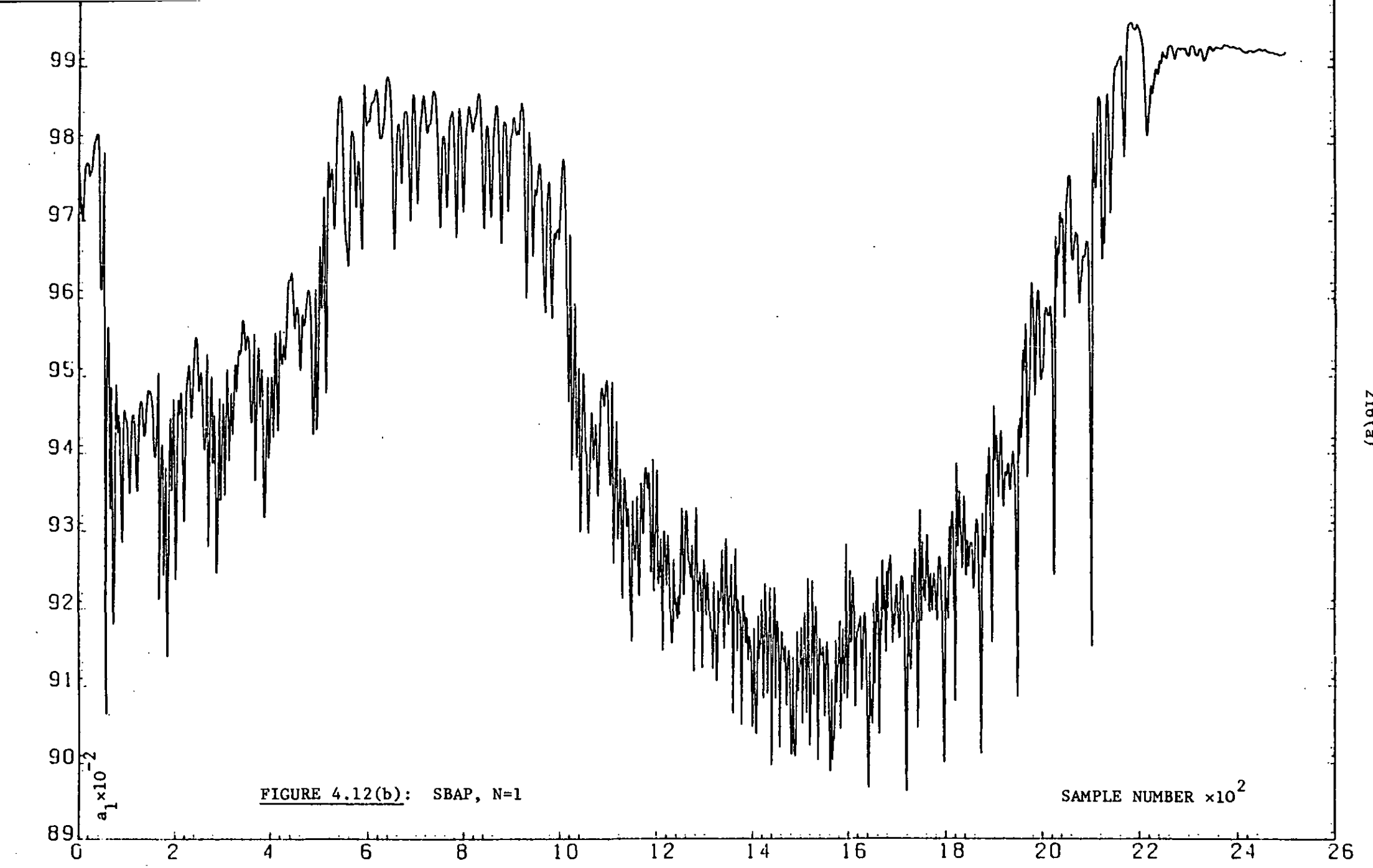


FIGURE 4.12(b): SBAP, N=1

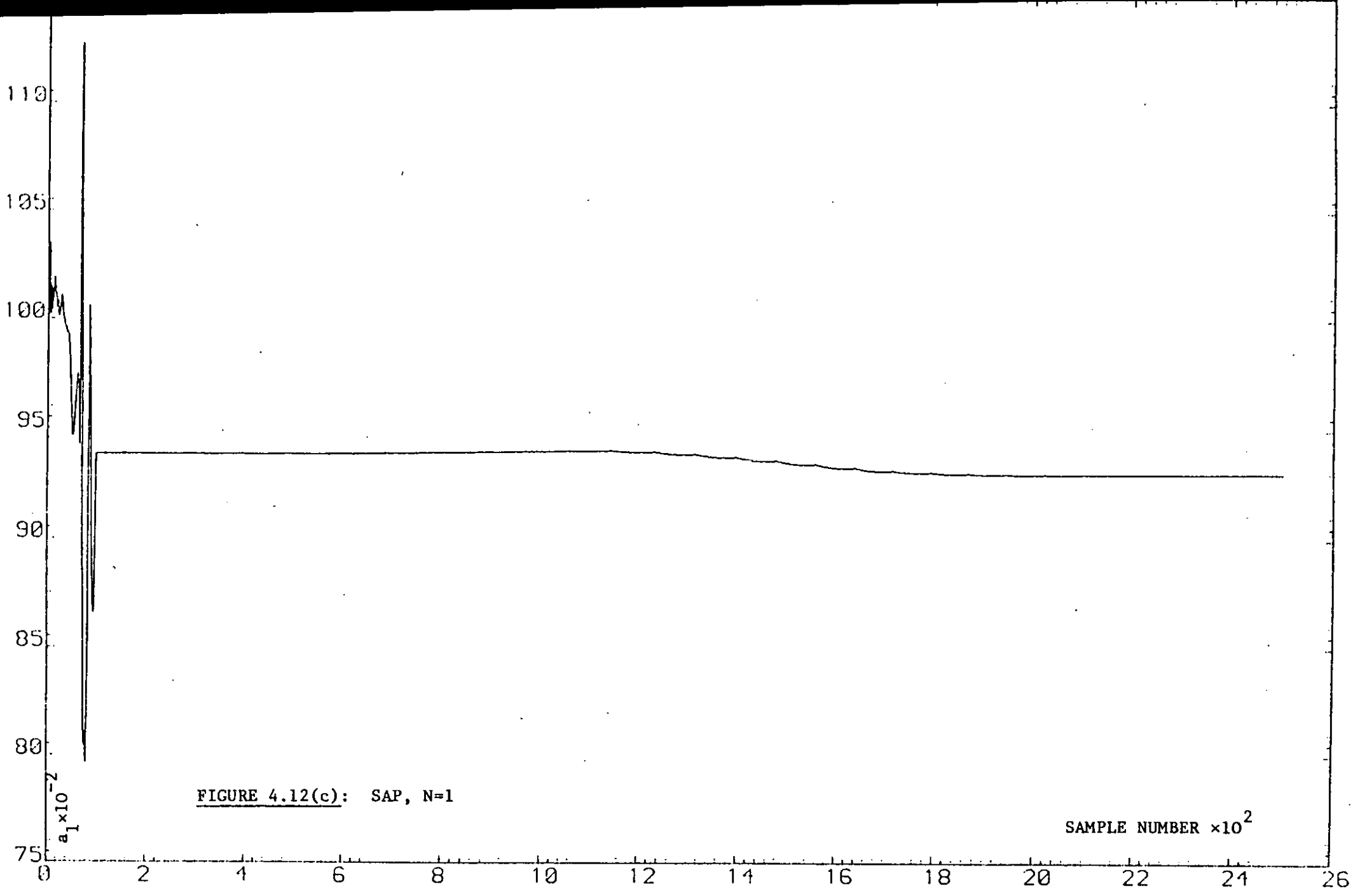


FIGURE 4.12(c): SAP, N=1

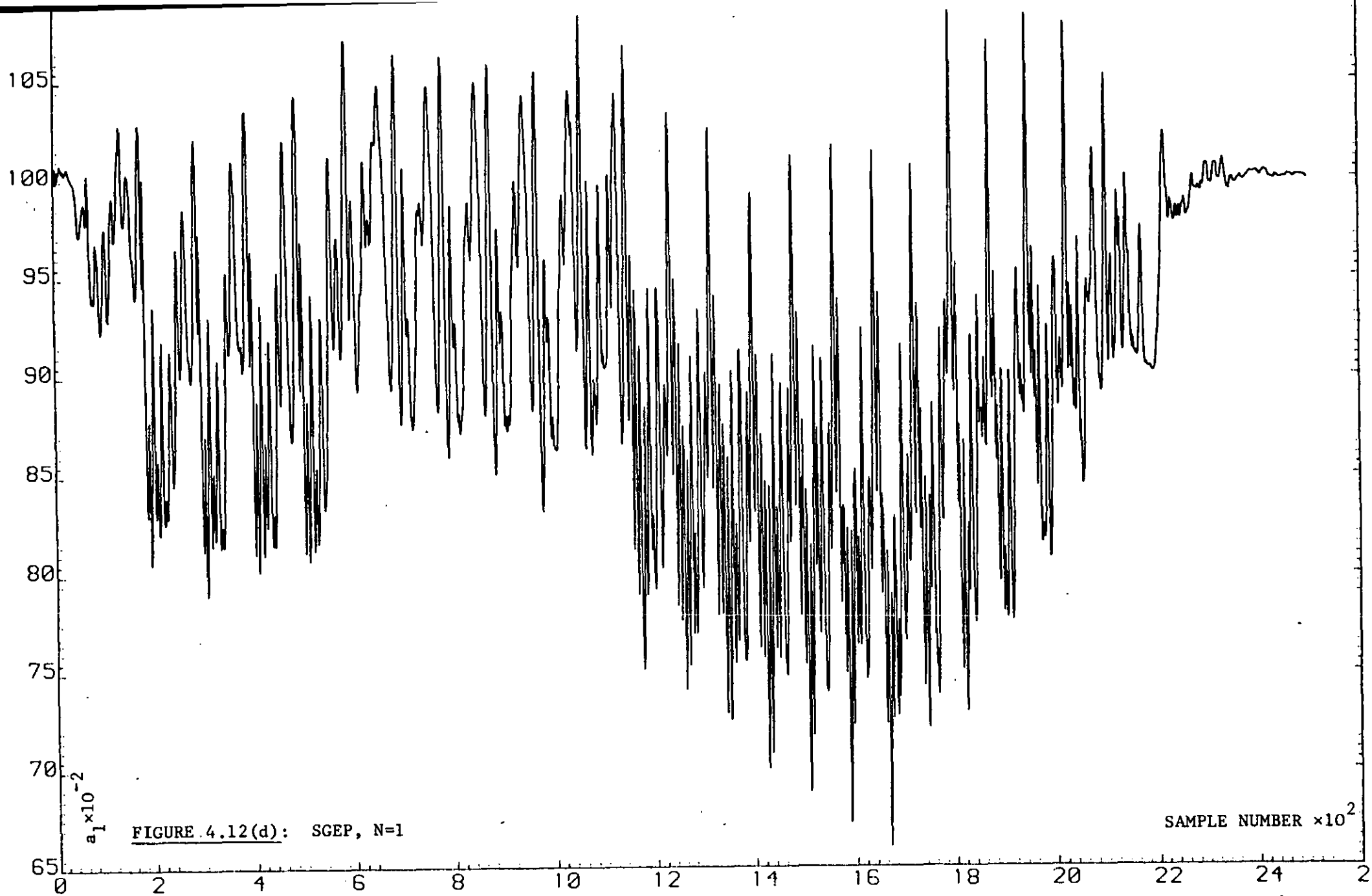


FIGURE 4.12(d): SGEP, N=1

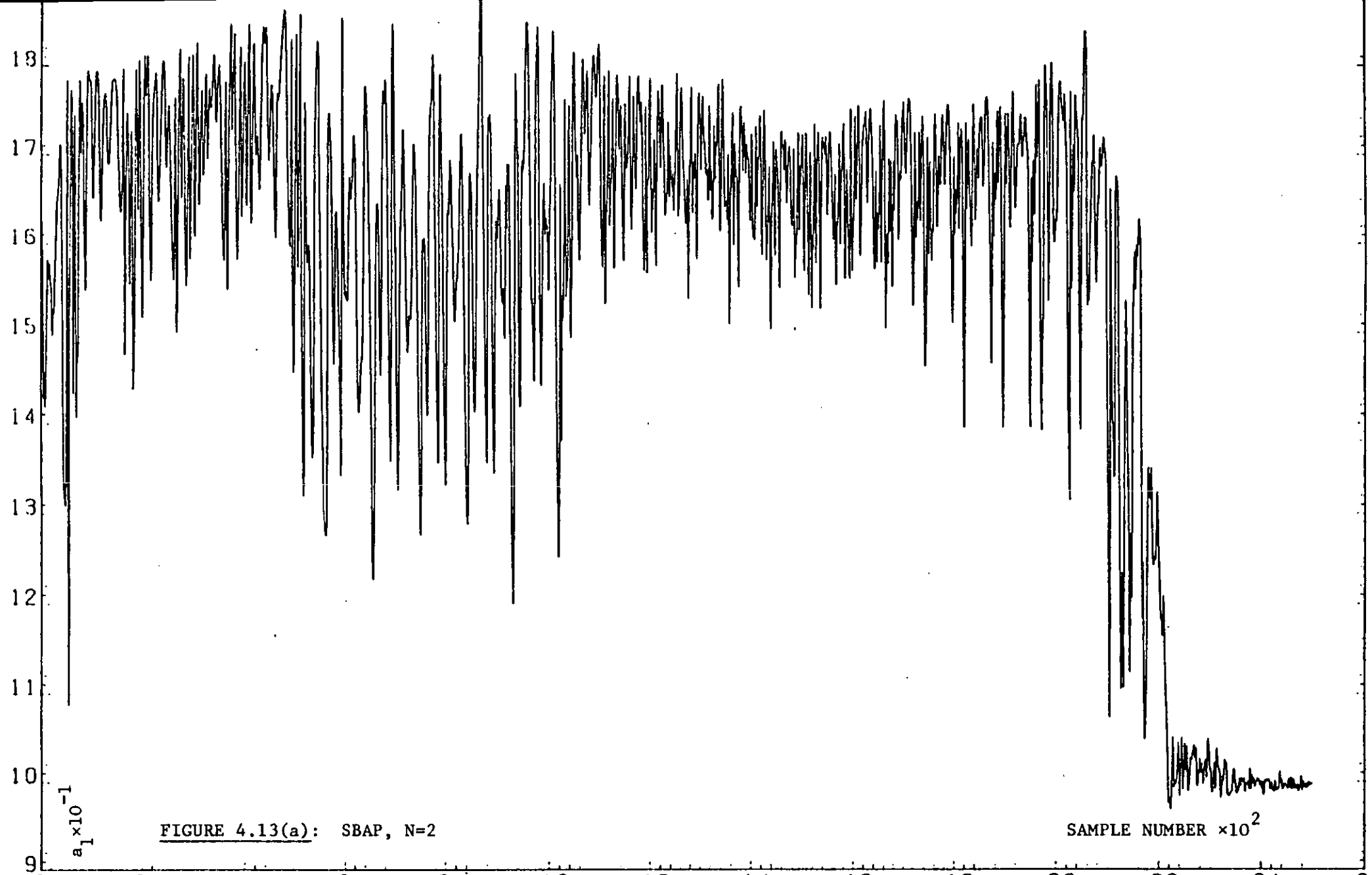


FIGURE 4.13(a): SBAP, N=2

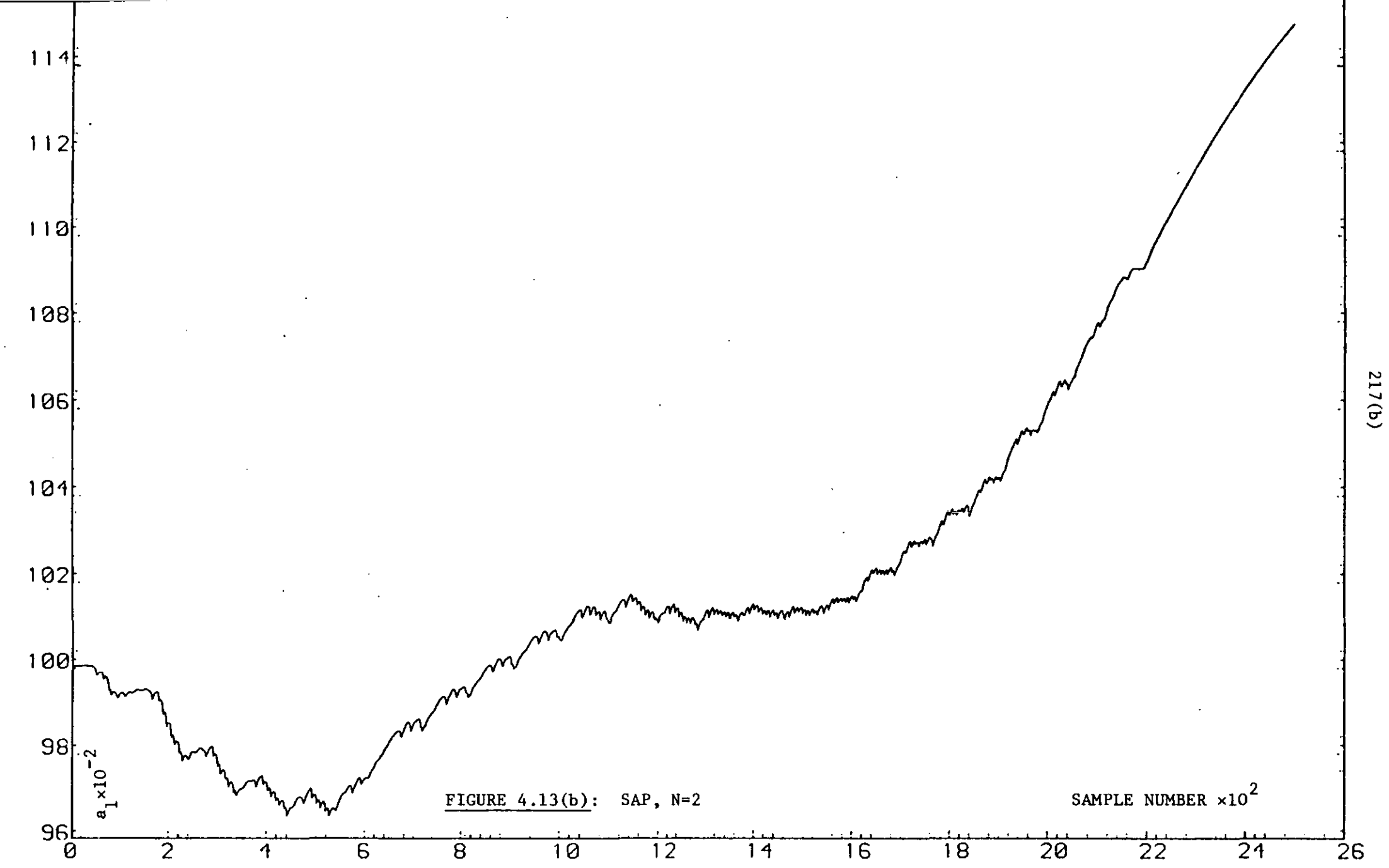


FIGURE 4.13(b): SAP, N=2

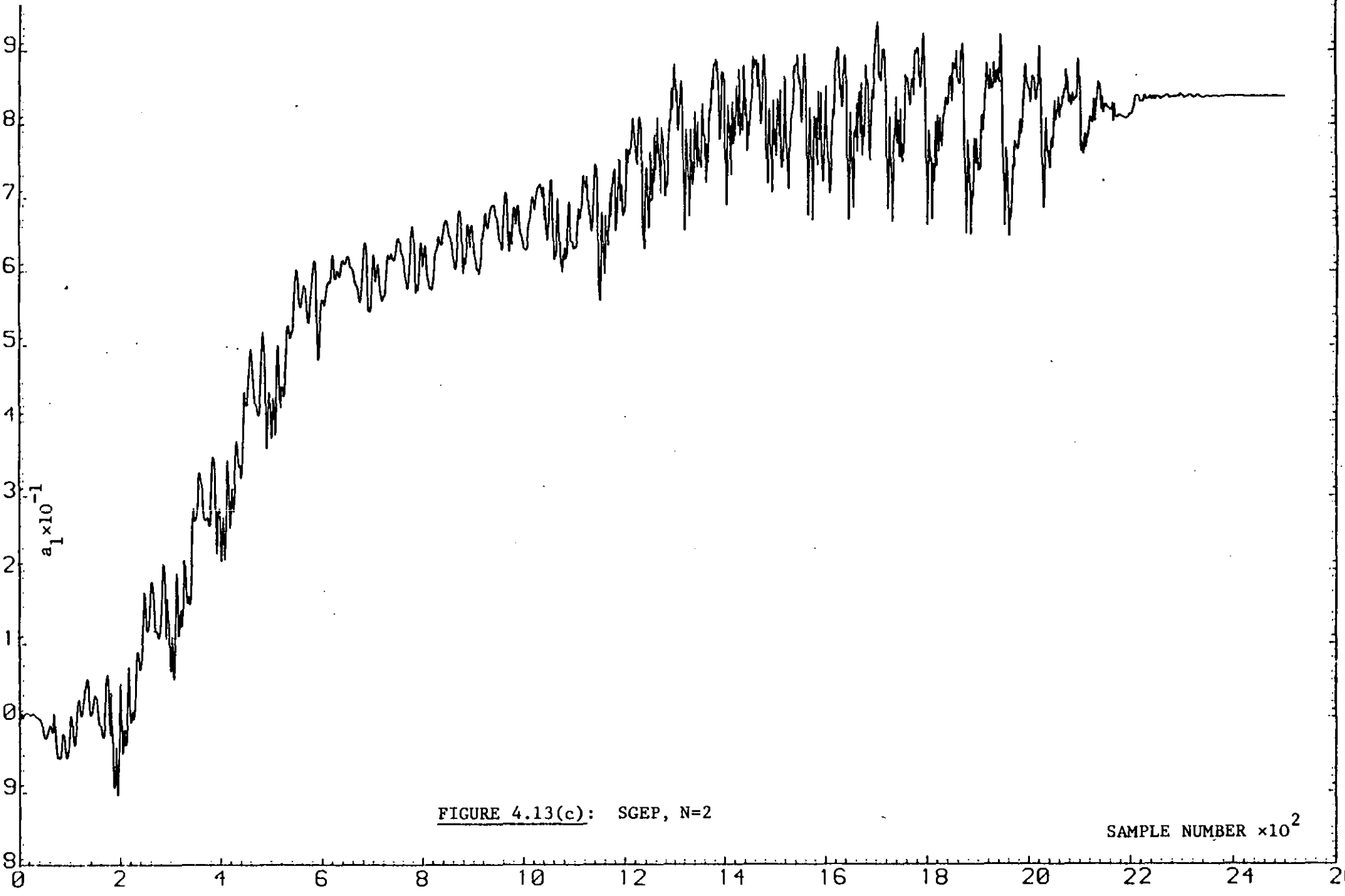


FIGURE 4.13(c): SGEP, N=2

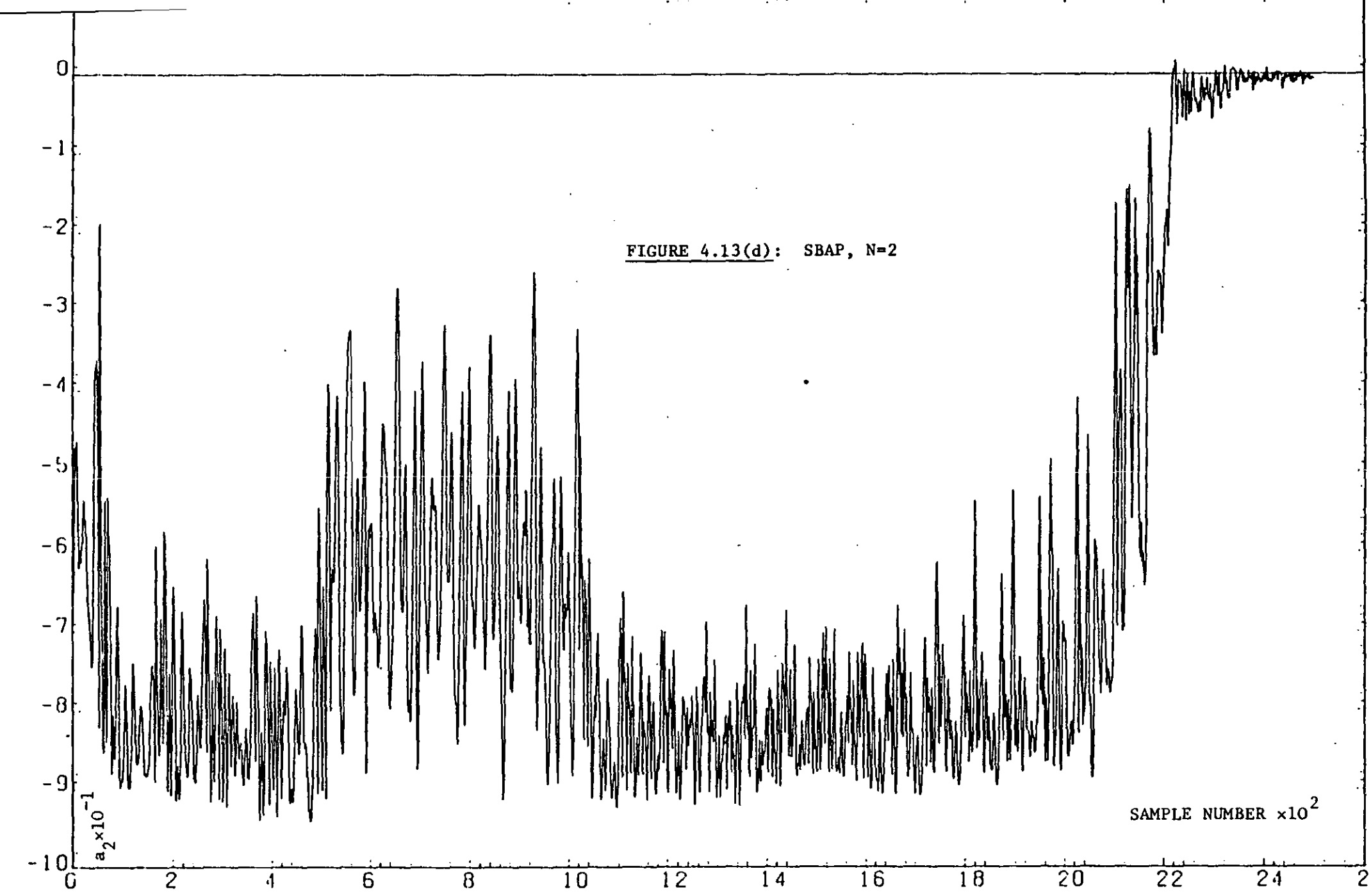


FIGURE 4.13(d): SBAP, N=2

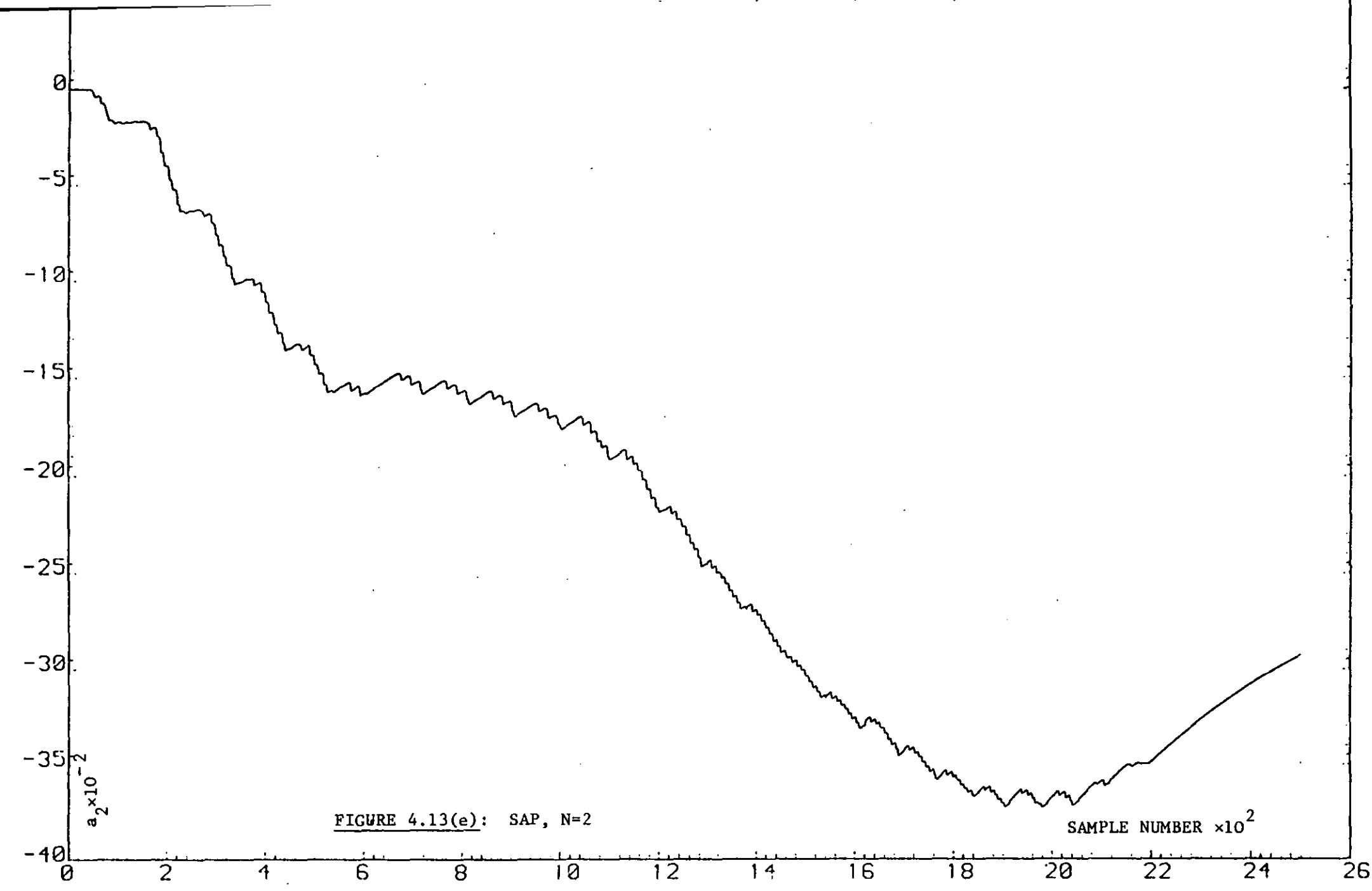


FIGURE 4.13(e): SAP, N=2

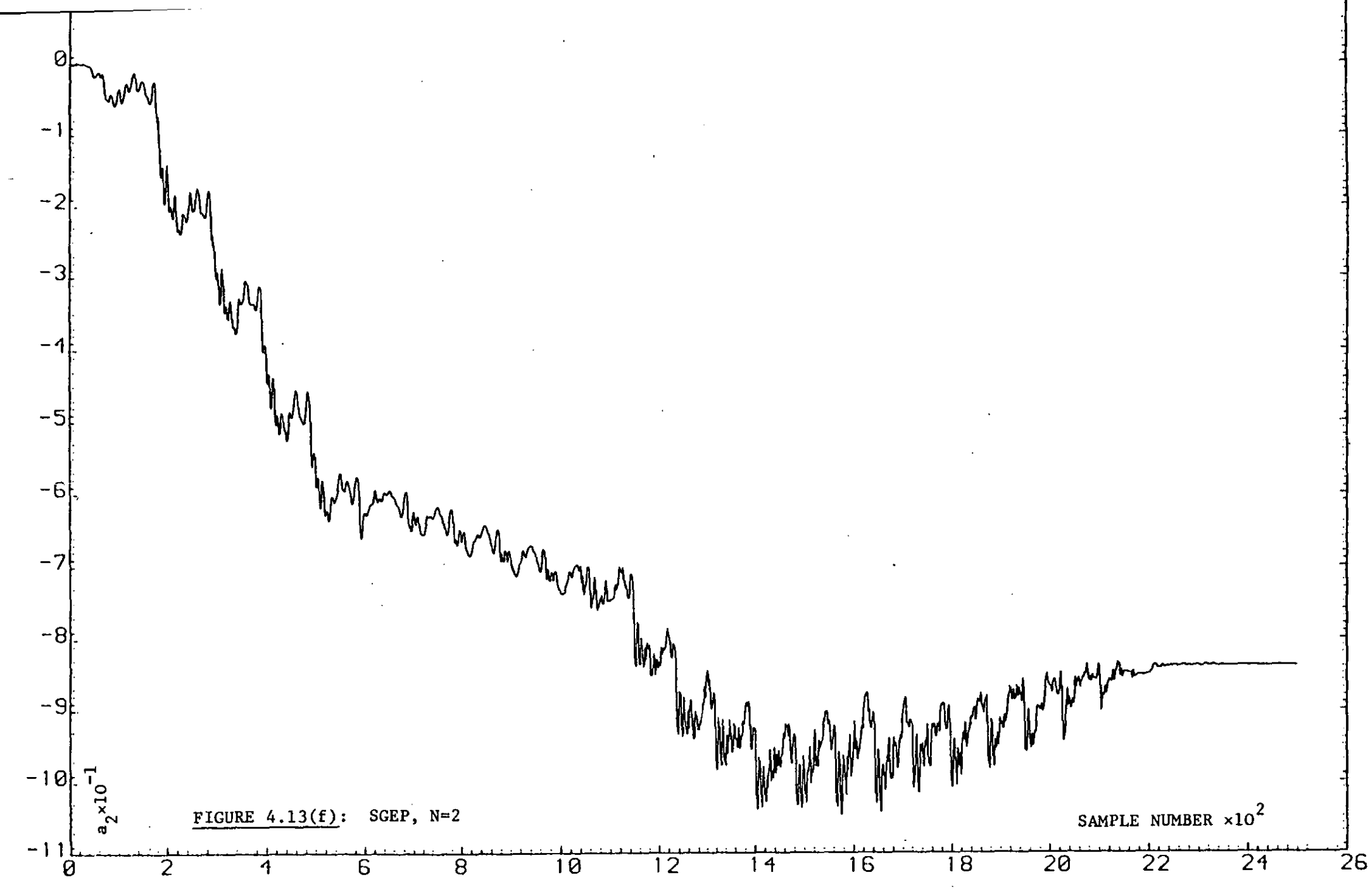


FIGURE 4.13(f): SGEP, N=2

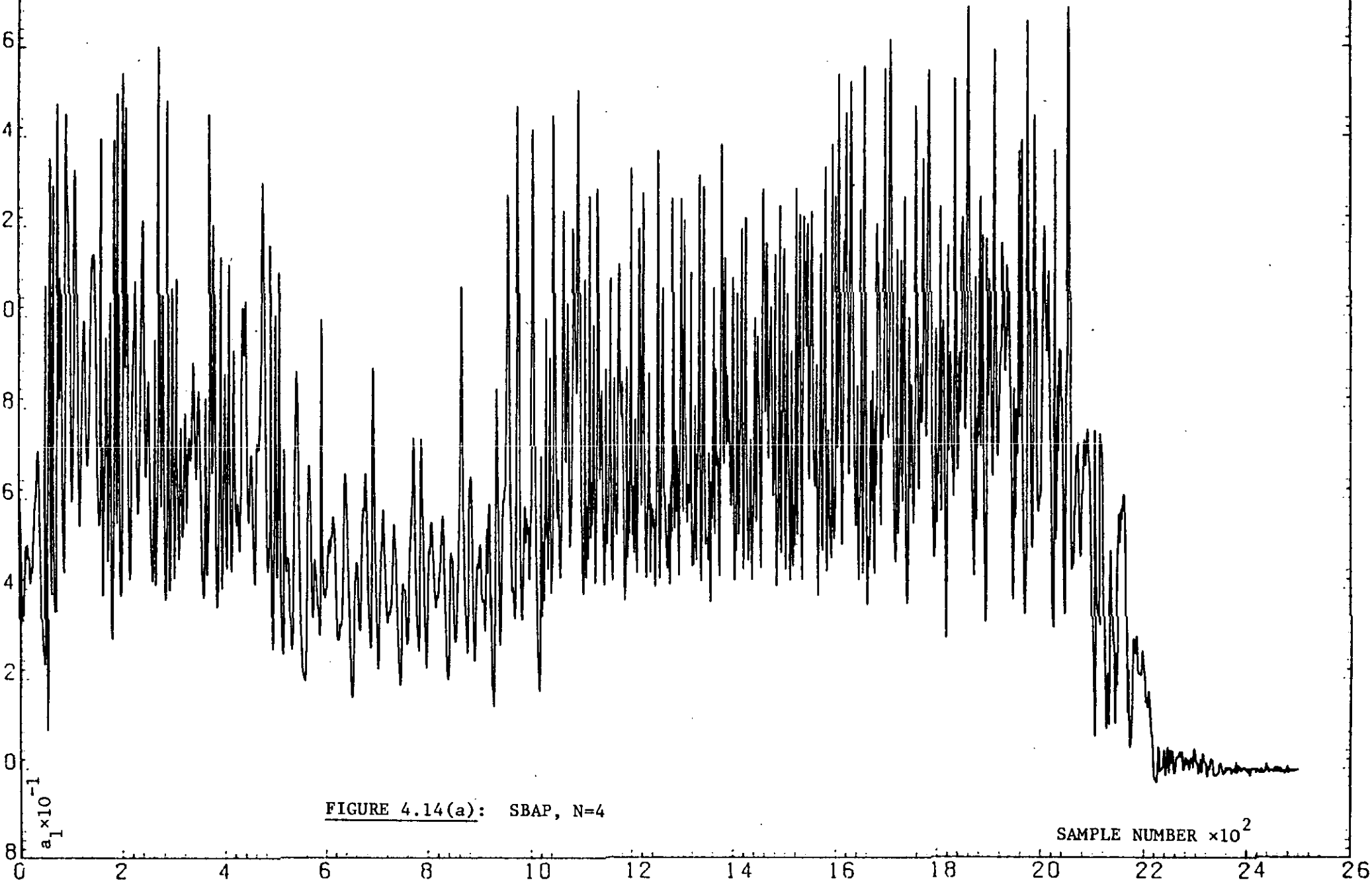


FIGURE 4.14(a): SBAP, N=4

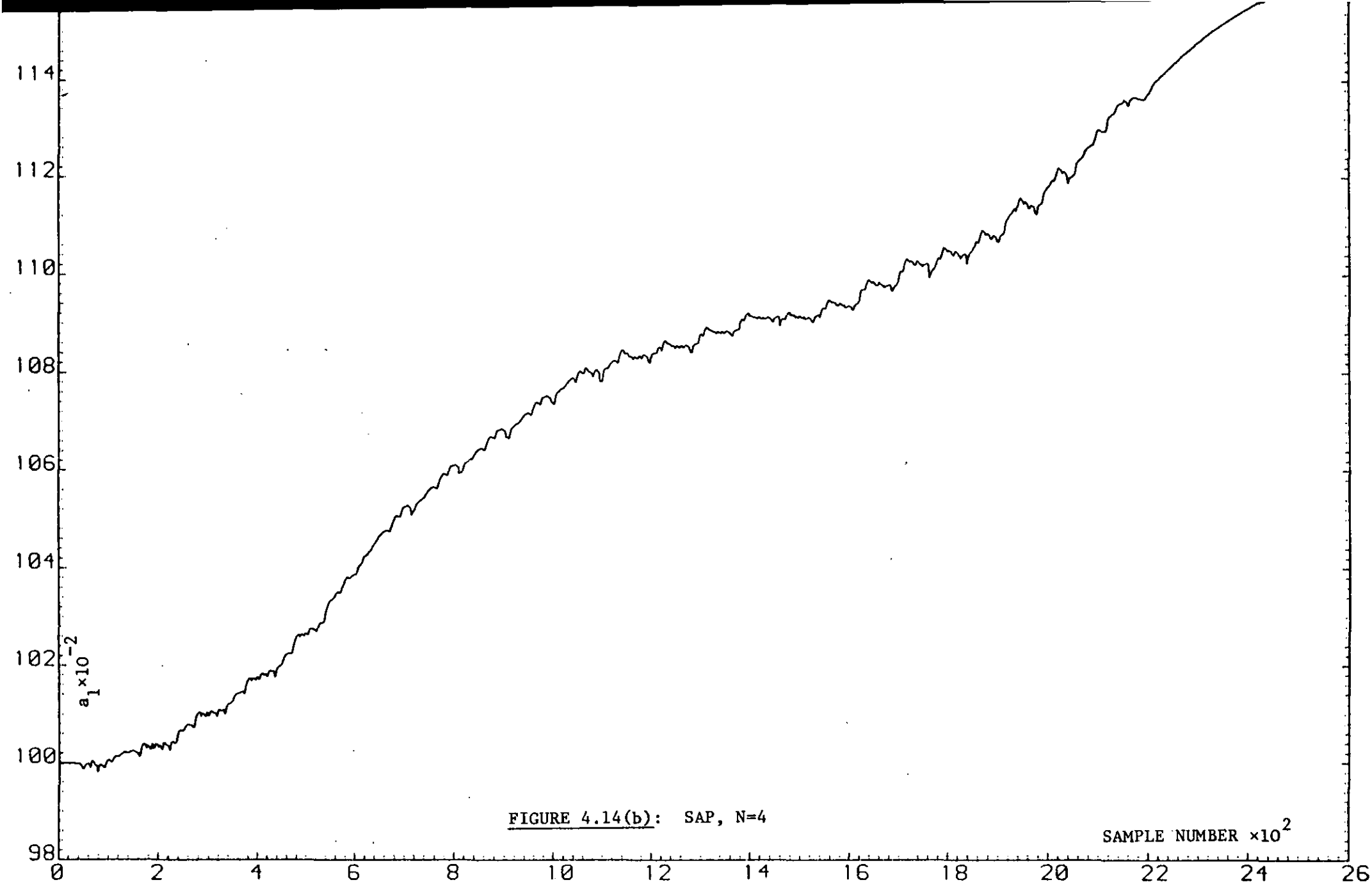


FIGURE 4.14(b): SAP, N=4

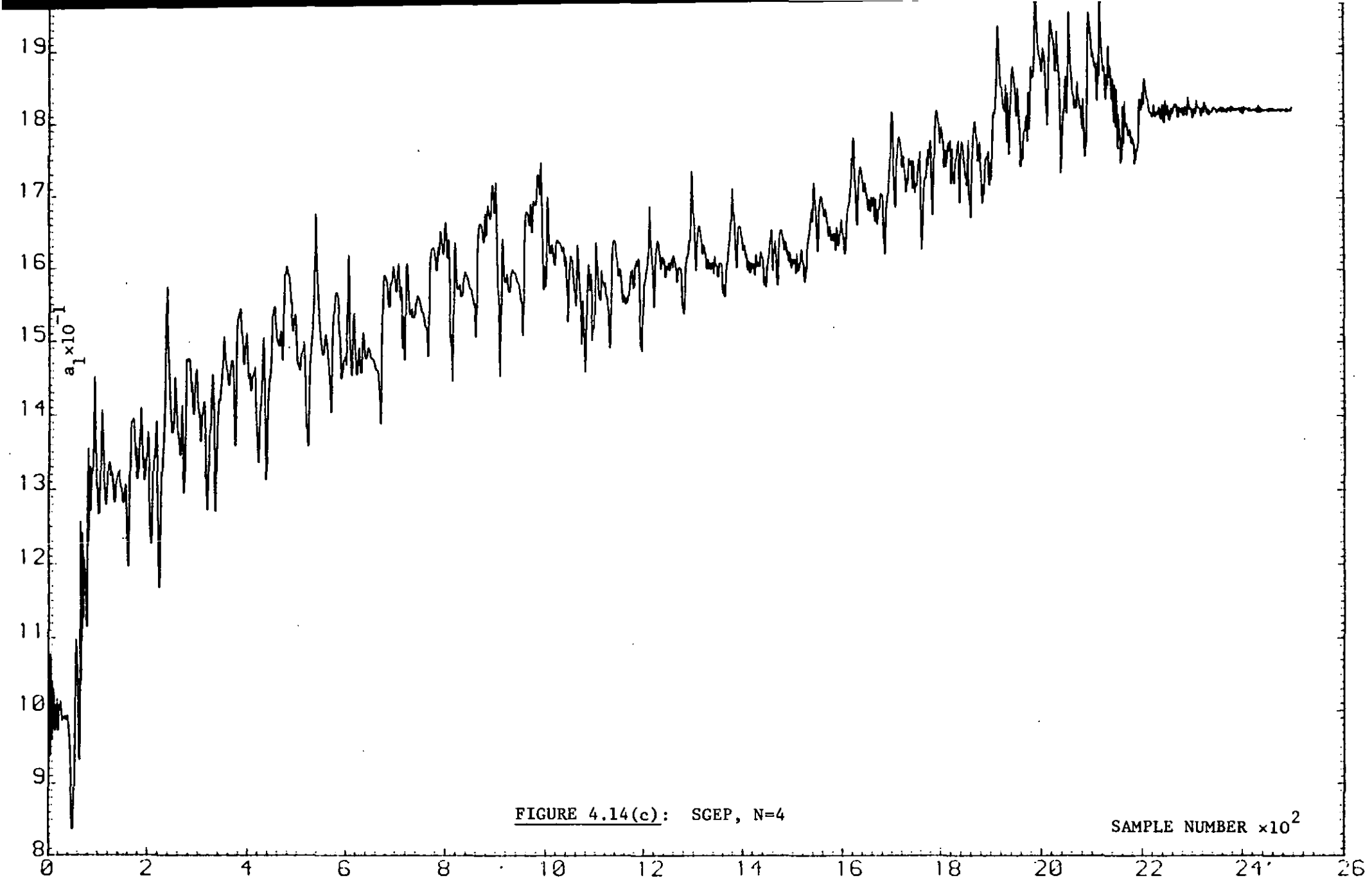


FIGURE 4.14(c): SGEP, N=4

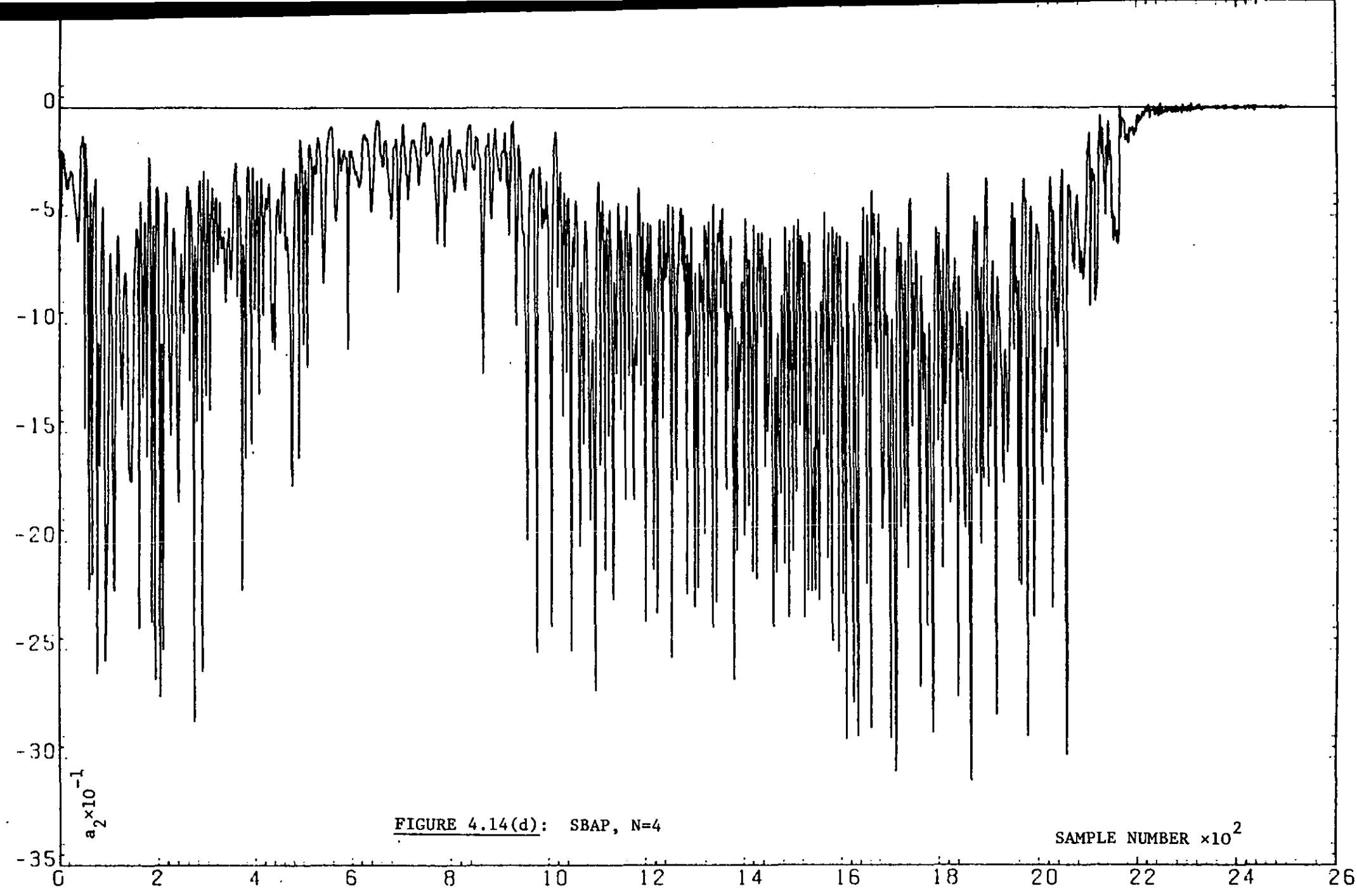


FIGURE 4.14(d): SBAP, N=4

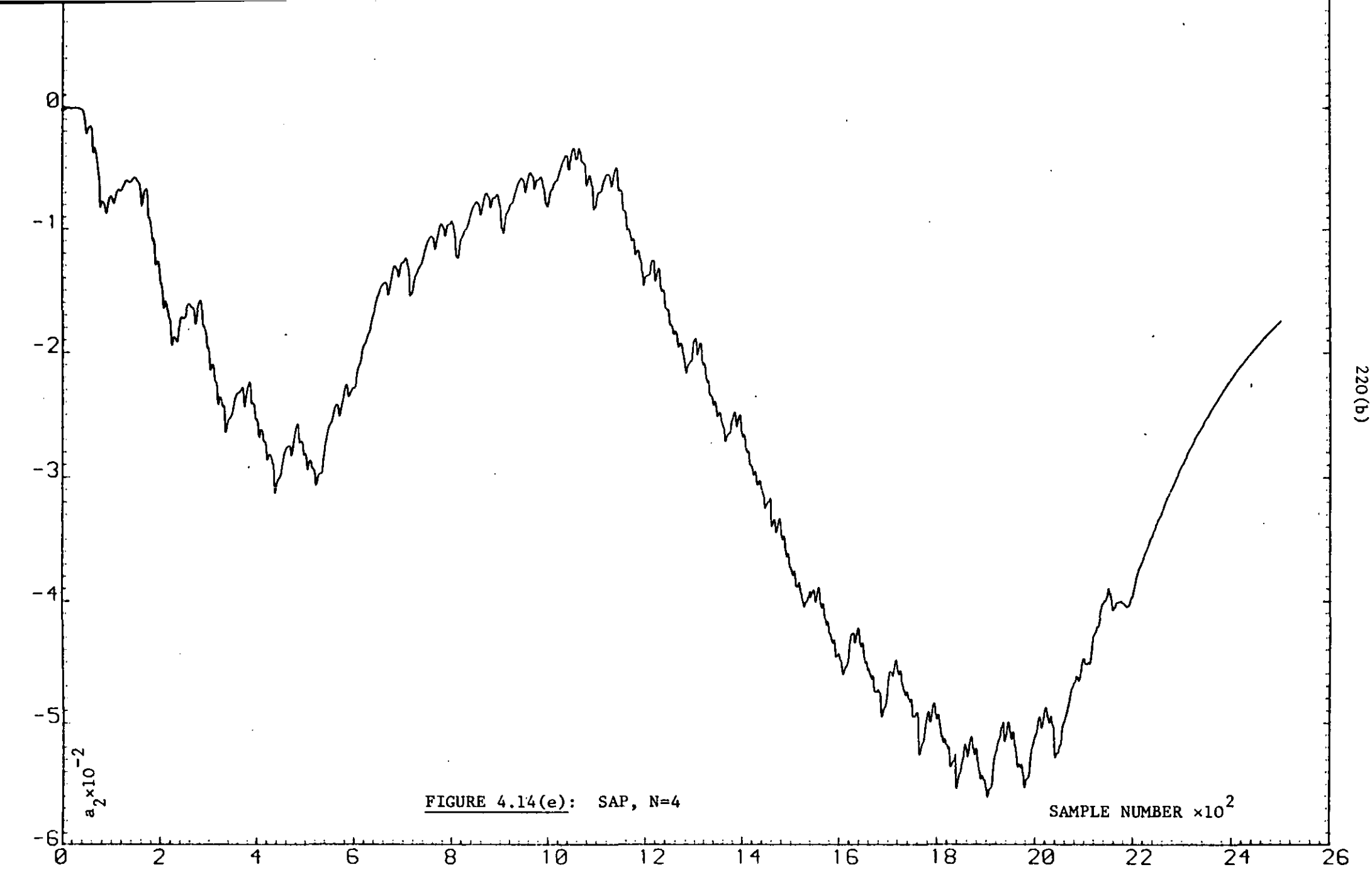


FIGURE 4.14(e): SAP, N=4

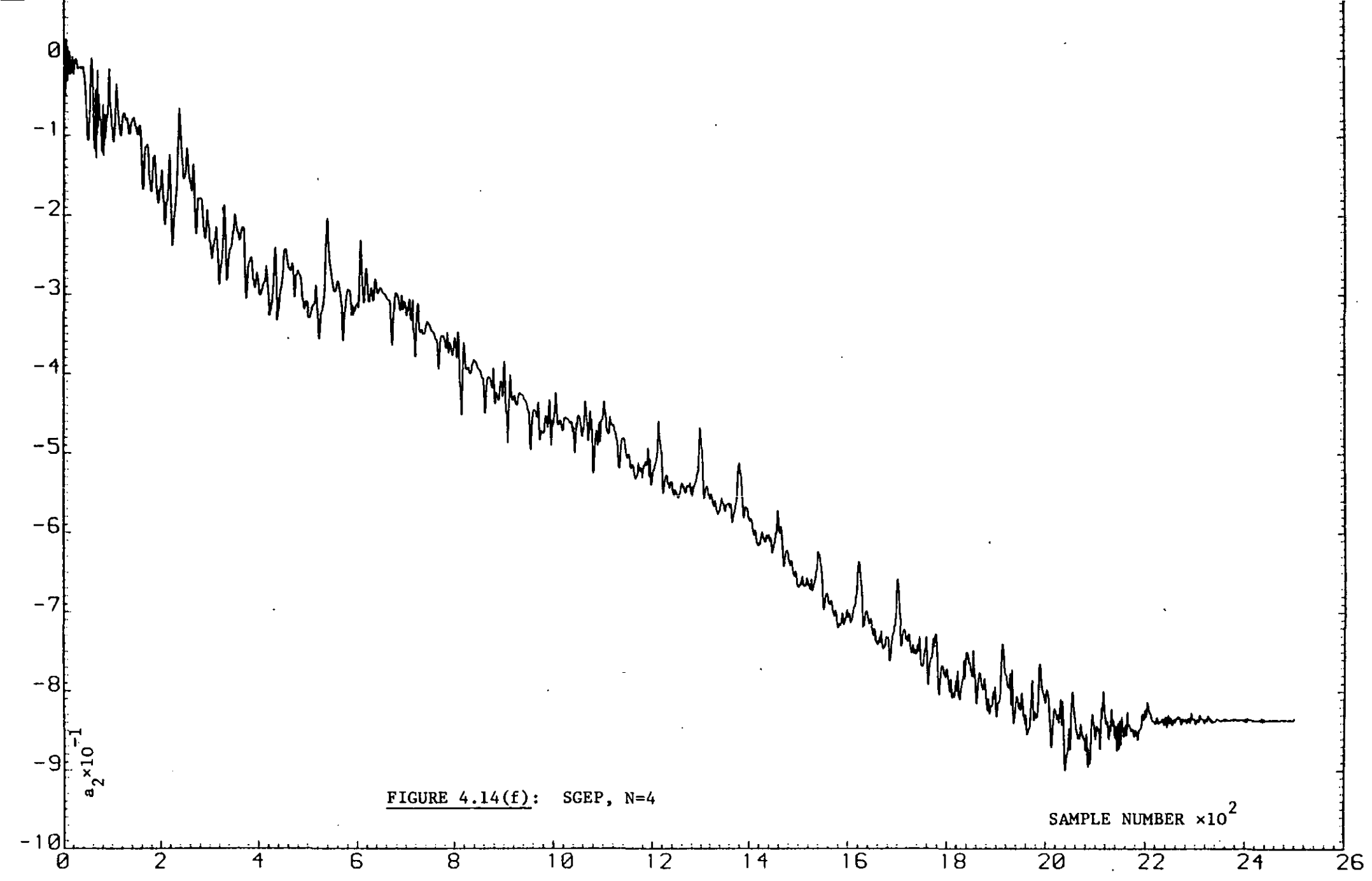


FIGURE 4.14(f): SGEP, N=4

4.10 FURTHER EXPERIMENTATIONS USING THE SAP, SGEP ALGORITHMS

In previous sections, the adaptive linear predictors, namely SAP and SGEP were examined in detail, and their performances were compared on the basis of SNR values and convergence behaviour. Both adaptive algorithms compute their coefficients iteratively, in an attempt to approach the optimum solution. As such, the matrix inversion required per sample basis, see SBAP, using Widrow's algorithm is eliminated.[111]

The sequentially adaptive predictors can be capitalized fully in the field of speech communication. Our main intention here is to apply these algorithms in DPCM encoded speech, as it will be investigated in the next chapter. However, in the present chapter, we make a digression to give briefly the outcomes of further experimentations employing such predictors. First we consider various parallel SAP/SGEP and SGEP/SGEP, configurations in an attempt to improve previously reported results in terms of SNR and dynamic range, DR. Then, we describe briefly the performance of SAP and SGEP in association with adaptive noise canceller.

4.10.1 Parallel-Predictor Structures

Figure 4.8 shows that although SGEP has faster adaptation rate, its dynamic range, DR, is smaller compared to the dynamic range of the SAP algorithm. A predictor having both the fast convergence properties of the SGEP algorithm and the wide DR of SAP will be therefore desirable. Such a predictor has been designed and it is shown in Figure 4.15. It

consists of the SAP and SGEP predictors both operating on the same speech sample, followed by a second modified SGEP having only two coefficients. Thus, the prediction outputs obtained from SAP and SGEP are again weighted by the second SGEP ($SGEP_2$), whose parameters are independent from the first SGEP ($SGEP_1$).

The proposed system of Figure 4.15 can be described as follows:

Suppose y'_i and y''_i are the prediction outputs, obtained at the i^{th} sampling instant, from the SAP and SGEP predictors respectively. Now, the coefficients of the second SGEP, at the same sampling instant, are $\theta_{i,1}$ and $\theta_{i,2}$, and they are given by,

$$\theta_{i,1} = \theta_{i-1,1} - \Lambda_{i-1,1}\ell_{i-1} \tag{4.109}$$

$$\theta_{i,2} = \theta_{i-1,2} - \Lambda_{i-1,2}\ell_{i-1} \cdot \tag{4.110}$$

$\Lambda_{i-1,1}$ and $\Lambda_{i-1,2}$ are formed as

$$\begin{aligned} \Lambda_{i-1,1} &= |e'_{i-1,1}| - |e'_{i-1,2}| \\ \Lambda_{i-1,2} &= |e'_{i-1,3}| - |e'_{i-1,4}| \end{aligned} \tag{4.111}$$

where

$$\begin{aligned} e'_{i-1,1} &= x_{i-1} - \{(\theta_{i-1,1}+s_1)y'_{i-1} + \theta_{i-1,2}y''_{i-1}\} \\ e'_{i-1,2} &= x_{i-1} - \{(\theta_{i-1,1}-s_1)y'_{i-1} + \theta_{i-1,2}y''_{i-1}\} \\ e'_{i-1,3} &= x_{i-1} - \{\theta_{i-1}.y'_{i-1} + (\theta_{i-1,2}+s_2)y''_{i-1}\} \\ e'_{i-1,4} &= x_{i-1} - \{\theta_{i-1}.y'_{i-1} + (\theta_{i-1,2}-s_2)y''_{i-1}\} \end{aligned} \tag{4.112}$$

and y'_{i-1} and y''_{i-1} are the prediction outputs at the $(i-1)^{th}$ sampling instant.

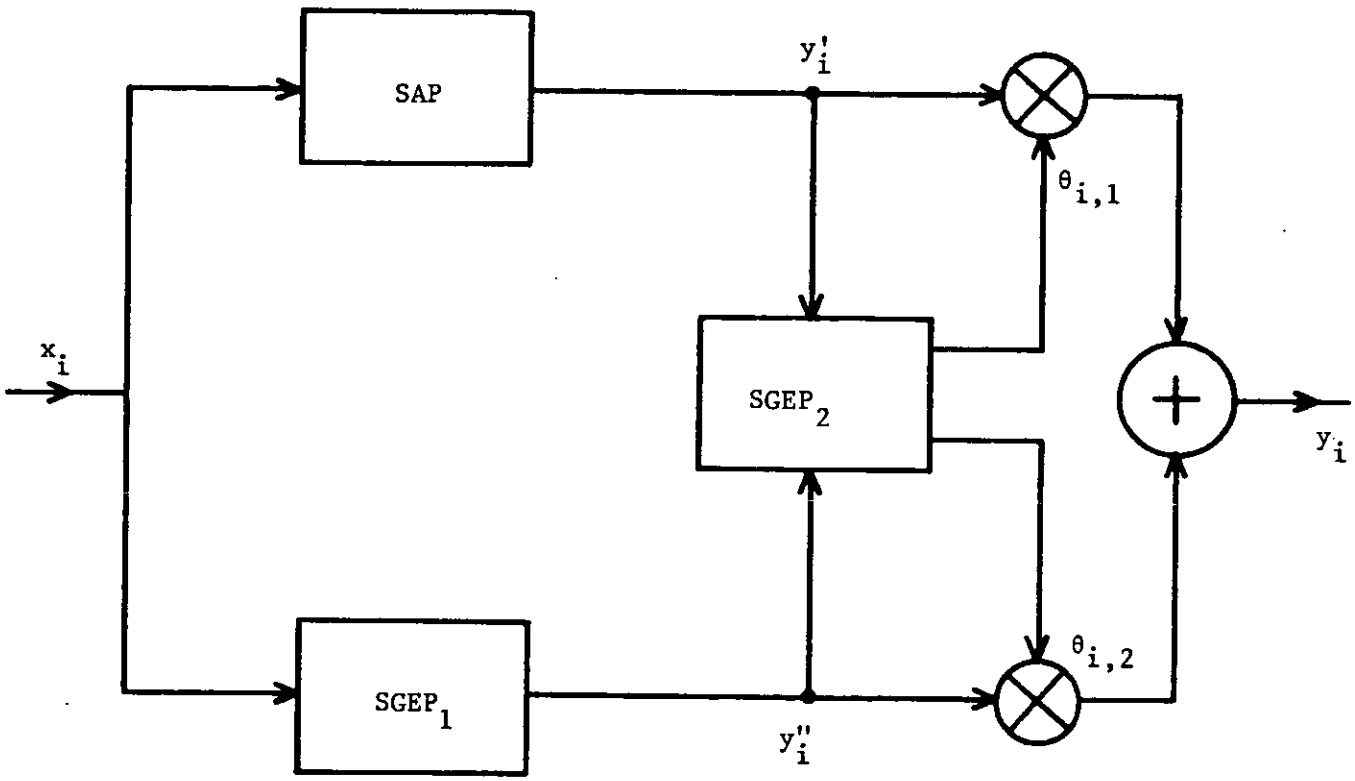


FIGURE 4.15: Parallel-Predictor (SAP/SGEP) Structure

Once the $SGEP_2$ coefficients are computed at the i^{th} instant, the final output is calculated as the linear combination of the y_i' and y_i'' samples, i.e.,

$$y_i = \theta_{i,1} \cdot y_i' + \theta_{i,2} y_i'' \tag{4.113}$$

where y_i is the prediction value of x_i. The parameter ℓ_{i-1} is determined according to

$$\ell_{i-1} = \frac{A_1}{B_1 + \frac{1}{NN} \sum_{j=1}^{NN} \left\{ \frac{y'_{i-j} + y''_{i-j}}{2} \right\}^2} \tag{4.114}$$

where A_1 and B_1 are constants.

Computer simulation experiments of this predictor were carried out with the parameters of SAP and $SGEP_1$ remaining unchanged (see Section 4.7) and with the constants of the $SGEP_2$ algorithm properly optimized. These are $\hat{\theta}_0=[1,0]^T$, $A_1=8.0$, $B_1=100$ and $NN=30$. The graphs of SNR as a function of input power levels for SGEP,SAP and the proposed combined structure are shown in Figure 4.16. We observe that the peak SNR of this *"Parallel prediction"* scheme is slightly reduced compared to the peak SNR of the SGEP algorithm, but at the same time, its dynamic range has been significantly improved.

Another system configuration considered in our experiments is shown in Figure 4.17. This is a ladder structure in which the error sequence generated by a single SGEP is aimed a. to be reduced in terms of its power and b. to be further randomized.

For simplicity, simulation was carried out using only one second

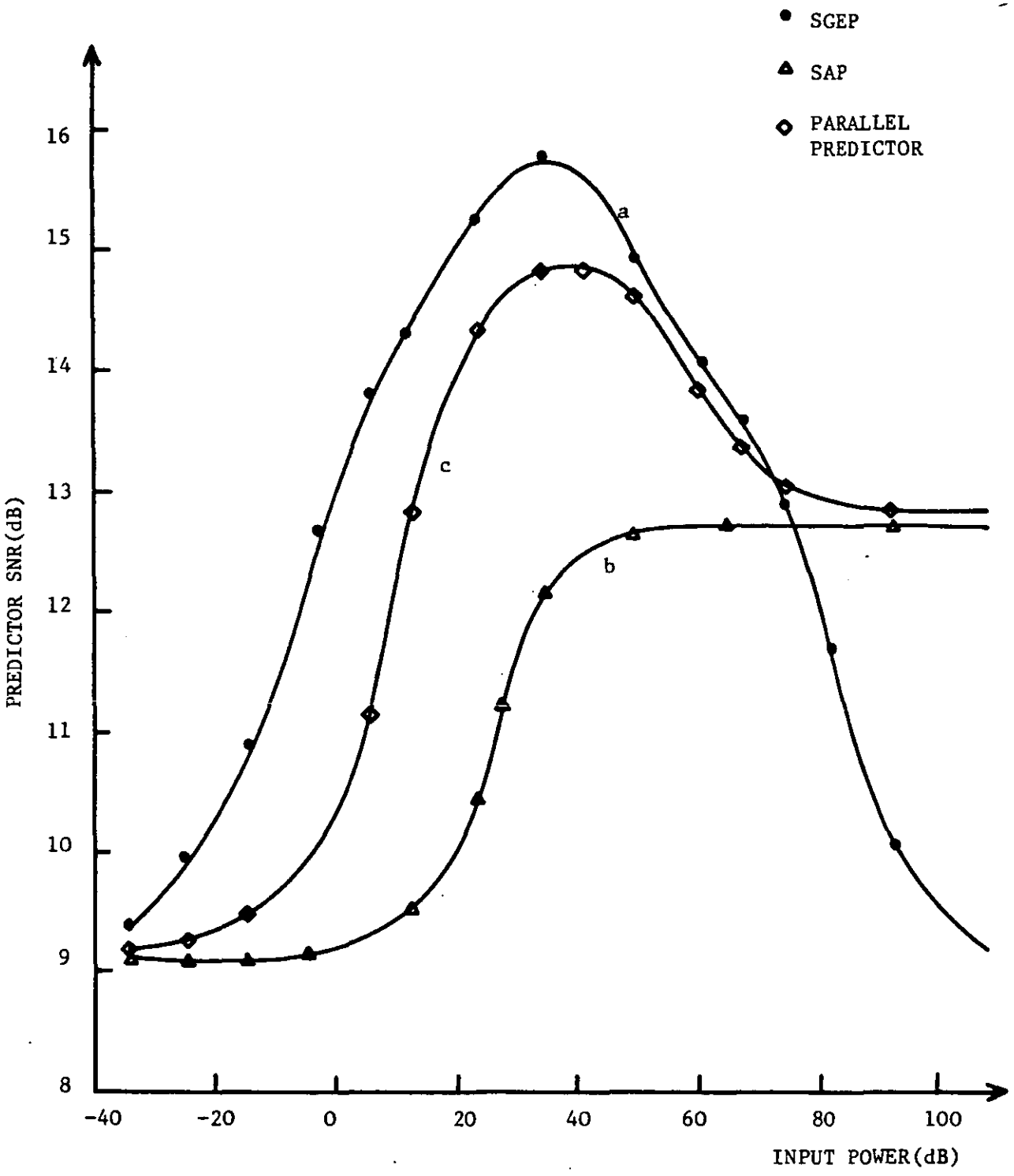


FIGURE 4.16: SNR Versus Input Power, M=N=15

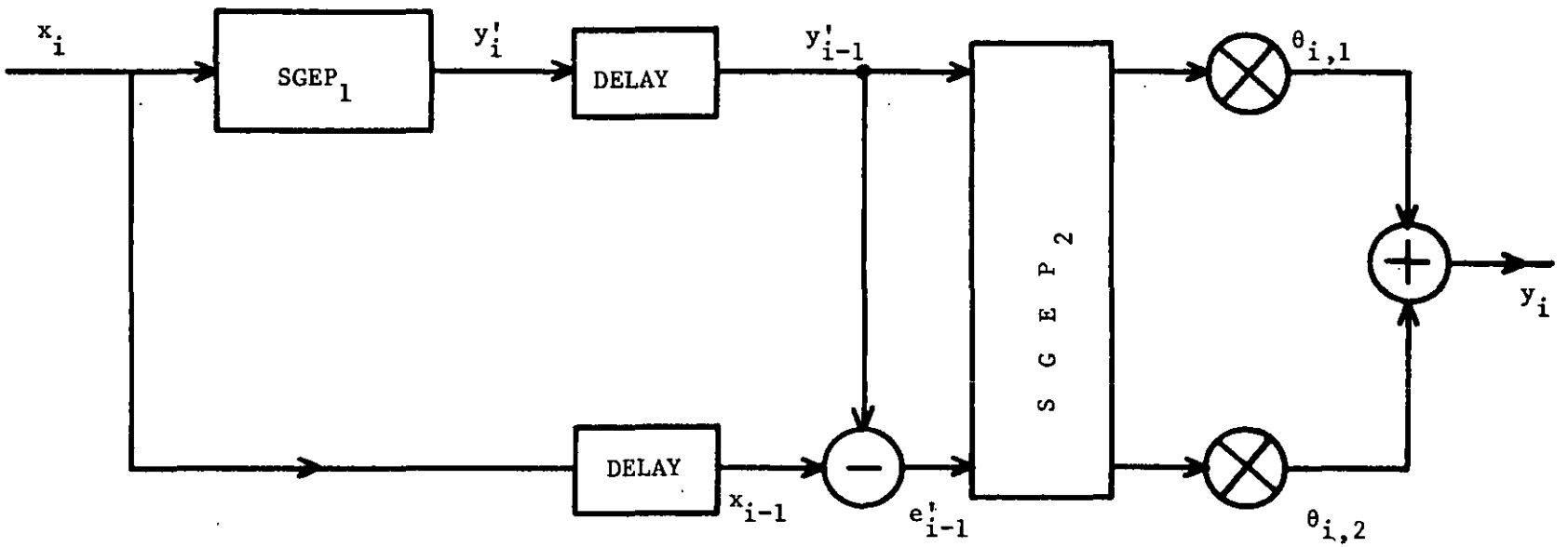


FIGURE 4.17: Two-Stage Ladder Structure

order, $SGEP_2$, predictor which operates on the output of $SGEP_1$, and its error sample delayed by one sampling period. The mathematical description of Figure 4.17 is the same as in Equations (4.109) to (4.113) with y''_{i-1} being replaced by e'_{i-1}. Hence,

$$y_i = \theta_{i,1} y'_i + \theta_{i,2} e'_{i-1} \tag{4.115}$$

and

$$\ell_{i-1} = \frac{A_2}{B_2 + \frac{1}{NN} \sum_{j=1}^{NN} (y'_{i-j})^2} \tag{4.116}$$

where A_2 and B_2 are constants.

The performance of this predictor together with optimally selected parameters, i.e., $\hat{\theta}_0=[1,0]$, N=15, NN=30, A_2=10 and B_2=100, are shown in Figure 4.18. Although there are no sufficient results to support the performance of the ladder structure, Figure 4.18(b) seems to suggest that the amplitude range of the second error signal is reduced and also that contains higher frequencies.

A better judgement of parallel configurations may be attained when they are associated with DPCM predictors and speech modelling processes. However, no further research was pursued towards this direction.

4.10.2 The SGEP and SAP Algorithms in Reducing the Acoustic Noise in Speech

In recent years, Adaptive Noise Cancelling, ANC,[117] has received a lot of attention in various aspects of signal processing. Fundamentally,

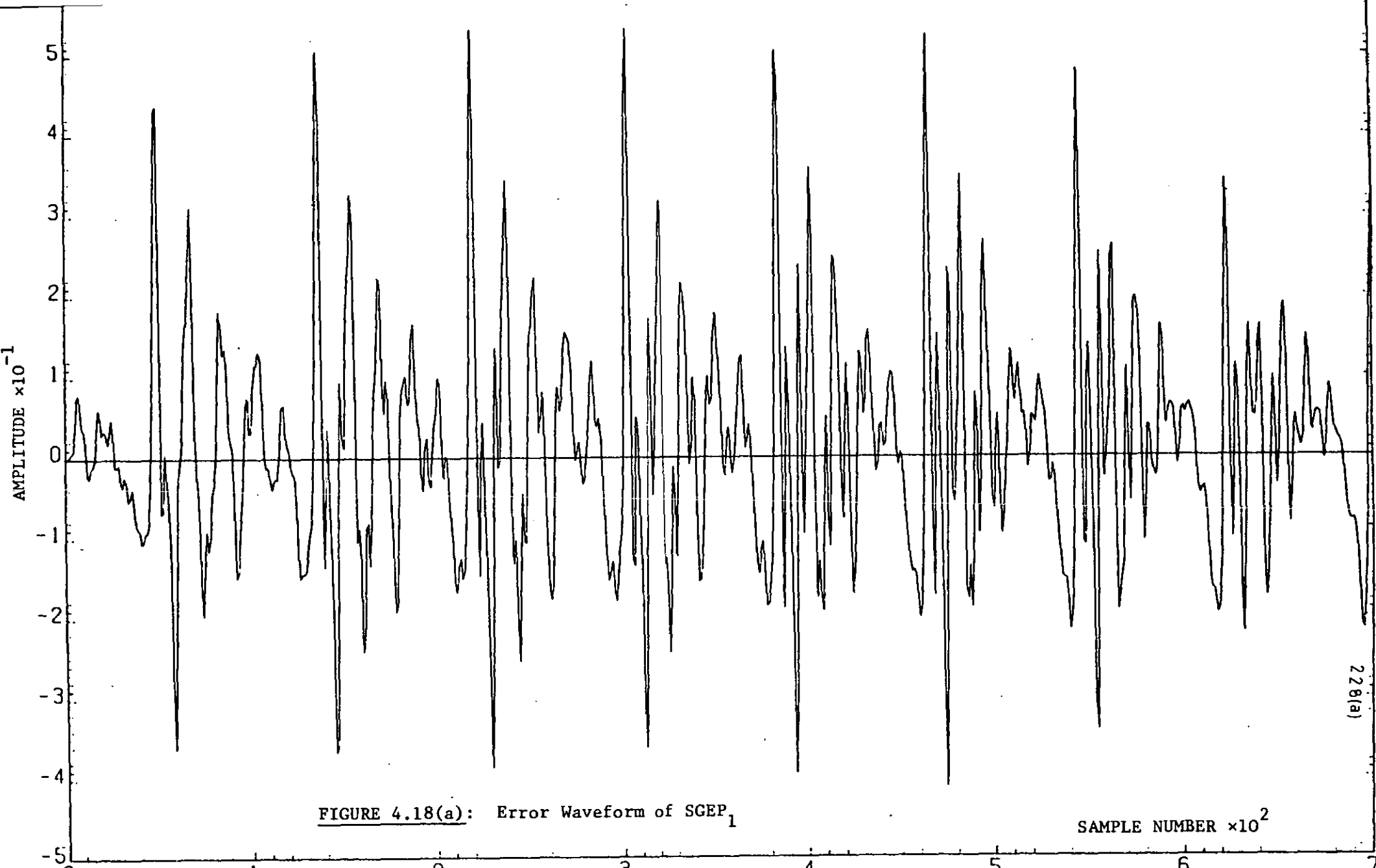


FIGURE 4.18(a): Error Waveform of $SGEP_1$

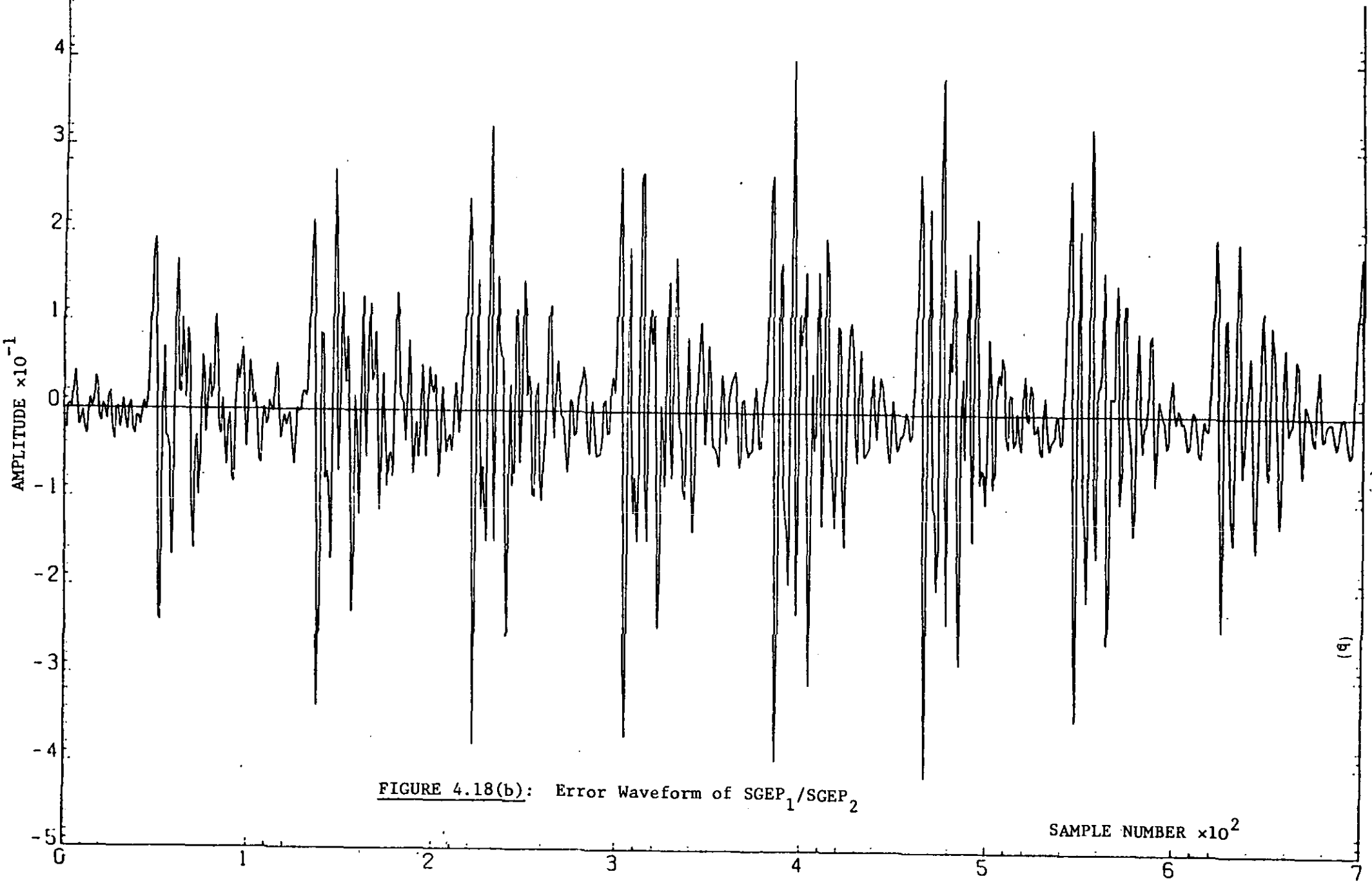


FIGURE 4.18(b): Error Waveform of $SGEP_1/SGEP_2$

it is based on a convenient technique for estimating the additive noise waveform, present in a signal. An adaptive predictor can be employed for this purpose so that it inherently incorporates a self-adjusting capability, see Figure 4.19.

We now, briefly present the performance results of SAP and SGEP in ANC, in terms of SNRSEG and subjective tests, obtained at this department.[118] The block diagram of ANC system used in the computer simulation experiments is shown in Figure 4.19. A random noise generator with a uniform p.d.f. was used to produce the reference noise signal, $\{XQ_i\}$. $\{XQ_i\}$ was also low pass filtered to provide the $\{XQF_i\}$ sequence, used to corrupt the speech samples $\{X_i\}$.

For these experiments SGEP and SAP were modified to efficiently model the transfer function between the source and the reference inputs. Thus the adaptation of the prediction coefficients was performed according to:

SAP:
$$a_{i+1,k} = a_{i,k} + P_i(x).XO_i.XN_{i-k} \tag{4.117}$$

SGEP:
$$a_{i+1,k} = a_{i,k} - \Lambda_{i,k}.P_i(x).\frac{1}{(N+1-k)^{\alpha}} \tag{4.118}$$

where
$$P_i(x) = \frac{A}{\frac{1}{N}\sum_{j=i-N-1}^{i-1}(XN)_j^2} \tag{4.119}$$

and
$$s_k = \frac{1}{D(N+1-k)^{\beta}} \tag{4.120}$$

$$k = 1,2,\ldots,N,$$

α,β,D are constants (see Section 4.7).

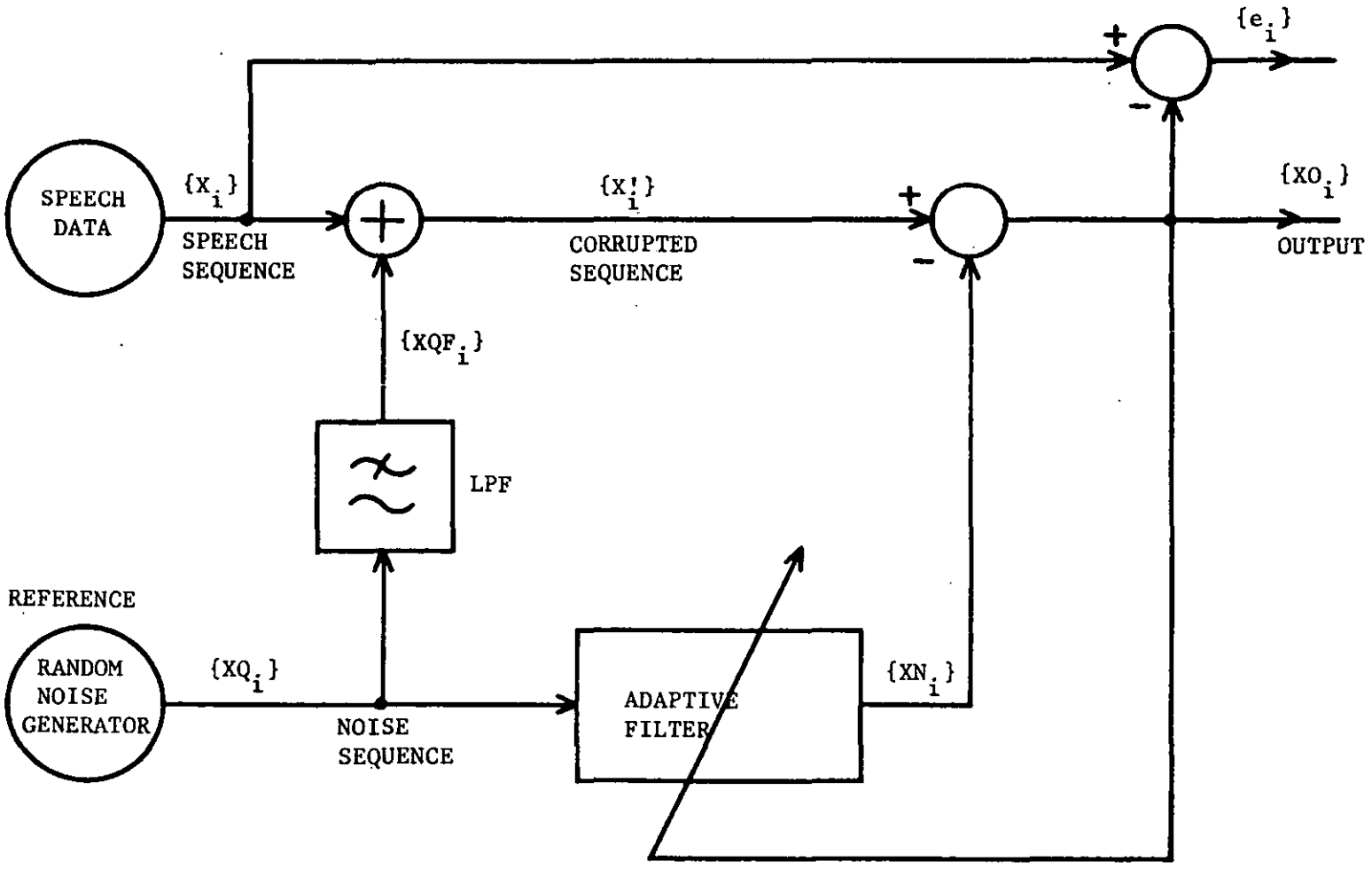


FIGURE 4.19: A Block Diagram of ANC for Simulation Purposes

Notice that, the constant B in the denominator of Equation (4.119) is eliminated. The reasons why are simply that there exists no undue oscillations here, unlike the speech during silence and the noise-waveform does not have any periodicity, respectively.

The SNRSEG improvement factor, SNRSEGIM was employed as performance criterion and is defined as,

$$\text{SNRSEGIM(dB)} = \text{SNRSEG(dB)}_{\text{output}} - \text{SNRSEG(dB)}_{\text{input}}$$

$$= \frac{1}{NB} \sum_{m=1}^{NB} \left[10 \log_{10} \left(\frac{X_i^2}{e_i^2} \right)_m - 10 \log_{10} \left(\frac{X_i^2}{XQF_i^2} \right)_m \right]$$

or

$$= \frac{1}{NB} \sum_{m=1}^{NB} 10 \log_{10} \left(\frac{XQF_i^2}{e_i^2} \right)_m \qquad (4.121)$$

where NB is the total number of blocks and m corresponds to the m^{th} block of samples.

The input data used in the experiments was speech of duration 5 sec., sampled at 4.8 kHz, and low pass filtered to 2.2 kHz. The results of using a 16^{th} order, N=16, filter can be described briefly as follows:[118]

a. At high level of interference, SNRSEG(dB)$_{\text{input}}$=-7 dB's, SNRSEGIM tends to grow fast. During the initial cycles of adaptation, the system with SAP yields an advantage of at least 1 dB over that using SGEP. The "A" values in Equation (4.119) for SAP and SGEP were 10^{-2} and 40, respectively. SGEP constants, α,β,D were 0.5, 0.3 and 10 respectively. Measurements taken after about 4 sec. of adaptation show that the SAP had lost 2-7 dB to the SGEP.

b. For lower levels of noise, SNRSEG(dB)=24 dB's, the system employing SGEP consistently gives better SNRSEGIM than the one with SAP. About 5 dB's of advantage is gained after 1 sec. of adaptation time and about 6 dB's at the end of 4 sec.

Subjective tests show, however, the system using SGEP produces a lower quality speech than that using SAP. Thus, although SGEP offers a faster adaptation response than SAP, when a steady state condition has reached in the modelling of the transfer function between the source and the reference inputs, SAP is able to oscillate with a smaller '*step size*' about the "*optimum*" model compared to SGEP which distorts the speech signal during the attempt to remove the noise.

4.11 DISCUSSION AND CONCLUSIONS

In this chapter various prediction techniques as applied to speech signals have been treated. The emphasis was directed to sequential prediction algorithms, that is to methods of solving the Wiener equation[101] in an iterative way. In particular, the SAP and SGEP techniques were examined in detail and SNR curves, error waveforms, rate of convergence, and computational time requirements, were considered for the performance evaluation of these systems.

In the proposed SGEP algorithm, each prediction coefficient is in turn increased and decreased in value, at every sampling instant, by a prescribed amount, while the other coefficients are kept constant. Predicitions are then made and the algorithm enables the coefficients to be modified by a certain amount, towards the correct direction. The SGEP method extends Sakrison's[119] sequential predictor, used for handling statistically stationary signals, so that non-stationary speech signals can be processed.

When compared to SAP, SGEP has an improved performance with a prediction error power of typically 3 dB's lower than that obtained from SAP, as can be seen in Figures 4.5 to 4.11.

The SNR improvement of SGEP over SAP is attributed to the faster convergence ability of the SGEP coefficients towards their optimum value. The mathematical treatment of the convergence of both SAP and SGEP is also provided for minimizing two error criteria, namely, the mean-square error and absolute error functions. Section 4.9 shows that

both algorithms converge to the optimum set of coefficients, $\hat{A}_{opt}$ (in this section, it is assumed that $\hat{A}_{opt}=\hat{A}_{s}$) where $\hat{A}_{opt}$ is defined from one sample sliding block autocorrelation method. Figures 4.12 and 4.14 illustrating the variations, with time, of the prediction coefficients of SAP, SGEP and the optimum block scheme, clearly support that SGEP converges faster than SAP towards the optimum solution.

In terms of the number of computations,the General Kalman and Simplified Kalman require $(2N^3+3N^2+3N)$ and $(2N^2+4N+1)$ multiplications per sample while the numbers of additions/subtractions per sample are $(2N^3+3N^2+1)$ and $(2N^2+2)$, respectively, where N is the number of weights in the predictor. However, SAP and SGEP require (3N+5), (5N+4) multiplications per sample, and 3N, 6N additions per sample respectively. As far as the processing time per sample based on the ICL 1900 computer is concerned, for N=8, SAP produces the lowest values, and SGEP requires considerably less computational time than the modified-Simplified Kalman predictor, see Table 4.3.

Further, the comparison of the sequential techniques with a fixed coefficient predictor shows that the latter having one coefficient (leaky integrator, $a_1=0.94$) is inferior. This becomes evident when the SNR of a leaky integrator is found to be of the order of 8.8 dB,see Figures 4.7(c) and 4.8(d). When the number of prediction coefficients of a fixed predictor increases to four, the SNR shows variations with different speech sentences, used as an input signal. The maximum SNR obtained for a 4^{th}-order fixed predictor having McDonald's average coefficients[55] is of the order of 12 dB's, see Chapter III, Figure 3.9(b).

The same figure also indicates that the use of a higher order predictor gives no advantage in terms of SNR. Hence, comparison of both Figure 3.9(b) and Figure 4.11 reveals that the adaptive scheme employing SGEP out-performs the fixed predictor using average coefficients.

In Section 4.10 a few applications of adaptive prediction techniques have been mentioned. Specifically, the problem of the cancellation of noise from speech has been addressed.

In the following chapter, we direct our research efforts towards the development of DPCM speech digitizer employing sequential predictors, namely SAP and SGEP.

CHAPTER V

DPCM EMPLOYING SEQUENTIAL PREDICTORS

5.1 INTRODUCTION

Adaptive DPCM is more efficient method of encoding commercial quality speech than log-PCM.[5] However, it is unlikely to dislodge the entrenched position of log-PCM in the commercial telephone network because of the high capital investment, unless trunk channel capacity becomes scarce. There may be a role for ADPCM to play in the local network, and more particularly in mobile radio, where its robustness to transmission errors is a valuable asset.

DPCM codecs having adaptive quantizers, but non-adaptive predictors, DPCM-AQJ or DPCM-AQF, have a superior performance to non-adaptive DPCM systems.[48] It is presumed that by using both adaptive quantizer and adaptive predictor, a large improvement in SNR would inevitably ensue. As this is generally not so, due to the predictors operating on sequences corrupted by quantization noise, our efforts were directed towards the investigation of adaptive predictors that are capable of achieving substantial gains in codec performance when operating with one of two types of well-known adaptive quantizers, namely AQJ and AQF.[48,58,64,70,120,121,122]

This chapter is therefore to examine the performance of ADPCM codecs employing the sequentially adaptive predictors presented in the previous chapter. We commence with a resume of the main elements of the ADPCM codecs used in our computer simulation experiments.

In Section 5.3, the performance of DPCM having one of the following three predictors, i.e., Fixed First Order, FFOP, SAP, SGEP, together with

Jayant's Adaptive Quantizer, AQJ, is presented for transmission bit rates of 16-40 Kb/sec. These codecs are abbreviated as DPCM-AQJ-FFOP, ADPCM-AQJ-SAP and ADPCM-AQJ-SGEP, respectively.

Jayant's quantizer is then replaced by one with forward step size transmission and computer simulation results of DPCM-AQF employing FFOP, SAP or SGEP are presented with and without additive noise in the transmission path. Similarly, these codecs are abbreviated as DPCM-AQF-FFOP, ADPCM-AQF-SAP and ADPCM-AQF-SGEP.

Comparison of these techniques are based on SNR(dB) and SNRSEG(dB) measurements, waveforms of the reconstructed signal as well as informal listening tests.

5.2 DPCM

DPCM has been described extensively in previous chapters and we will therefore confine ourselves to mentioning only its salient features.

Given that the predictors operate on the locally decoded sequence $\{\hat{X}_i\}$, let us exacerbate the prediction difficulties by reducing the number of quantization levels in order to reduce the transmitted bit-rate. Having fewer levels leads us to consider the design of the quantizer more carefully. As the power in the error signal rises when voiced speech is present, compared to unvoiced speech, the quantizer must be capable of extending and contracting the position of its levels in order to achieve a nearly constant SNR. Consequently, we will discuss, in the next sections, the quantization and prediction schemes used in our DPCM studies.

5.2.1 Quantizers

Jayant's Adaptive Quantizer, $AQJ^{(51,64,120-122)}$ has been used extensively in DPCM systems. This quantizer produces at each sampling instant an output quantization level, $\hat{e}_i$ and a quantization level number R_i where the sub-script i refers to the i^{th} sampling instant. R_i has two components, sign and magnitude. The magnitude component of R_i increases progressively from the centre of the quantizer. This level number R_i having values

$$R_i = \pm\frac{(2k-1)}{2} \qquad k=1,2,\ldots,2^{b-1} \tag{5.1}$$

is represented by a b-bit word and transmitted. R_i is also operated on locally. Using one sample delay and a look-up table, a multiplier number

is formed. This multiplier at the $(i-1)^{th}$ instant is $M(|R_{i-1}|)$, and is used to calculate the uniform adaptive quantization step size of i^{th} instant:

$$\Delta_i = M(|R_{i-1}|)\Delta_{i-1} \quad . \tag{5.2}$$

The output quantization level $\hat{e}_i$ is therefore

$$\hat{e}_i = \Delta_i R_i \quad . \tag{5.3}$$

In this way, AQJ accepts the error sample e_i and forms R_i and $\hat{e}_i$. R_i is transmitted as a binary code and also facilitates the formulation of the multiplier constant and hence the next step size. $\hat{e}_i$ is generated both locally and at the receiver. The step size in AQJ, given by Equation (5.2) expands and contracts the quantizer range like an accordian in an attempt to confine the components in the input error sequence $\{e_i\}$ to within the range of the quantizer.

Although AQJ, described in Equation (5.2), performs well for ideal channels, it is extremely susceptible to transmission errors. Goodman et al[67] proposed a robust to noise, AQJ algorithm where a *"leakage"* constant β_q, $0 \leq \beta_q \leq 1$, is introduced so that

$$\Delta_i = M(|R_{i-1}|)\Delta_{i-1}^{\beta_q} \quad . \tag{5.4}$$

The scheme presented by Equation (5.4) has been studied in depth by Mitra[123] and Einarsson.[124]

In practice, Δ_i is constrained to

$$\Delta_{min} \leq \Delta_i \leq \Delta_{max} \tag{5.5}$$

resulting in the DPCM encoder of having a dynamic range of approximately,

$$DR(dB) = 20 \log_{10}\left\{ \frac{\Delta_{max}}{\Delta_{min}} \right\} \qquad (5.6)$$

Figure 5.1 shows the block diagram of DPCM-AQJ codec used in our computer simulation experiments.

Another adaptive quantizer, frequently employed in DPCM is that with forward step-size transmission, AQF. The word forward is used to imply that the step size, Δ, of this uniform adaptive quantizer is evaluated every W input speech samples and is transmitted every W/f_s secs. as side information. As a consequence, the received speech is delayed by W sampling periods. Jayant[110] used DPCM-AQF for digital transmission of speech through noisy channels, such as those encountered in mobile radio. The main reason in using AQF rather than sequentially updated AQJ-quantizer is its robust performance in the presence of channel errors provided that Δ is protected and correctly received. Also, the AQF quantizer can be easily implemented. Figure 5.2 shows the block diagram of the DPCM-AQF codec.

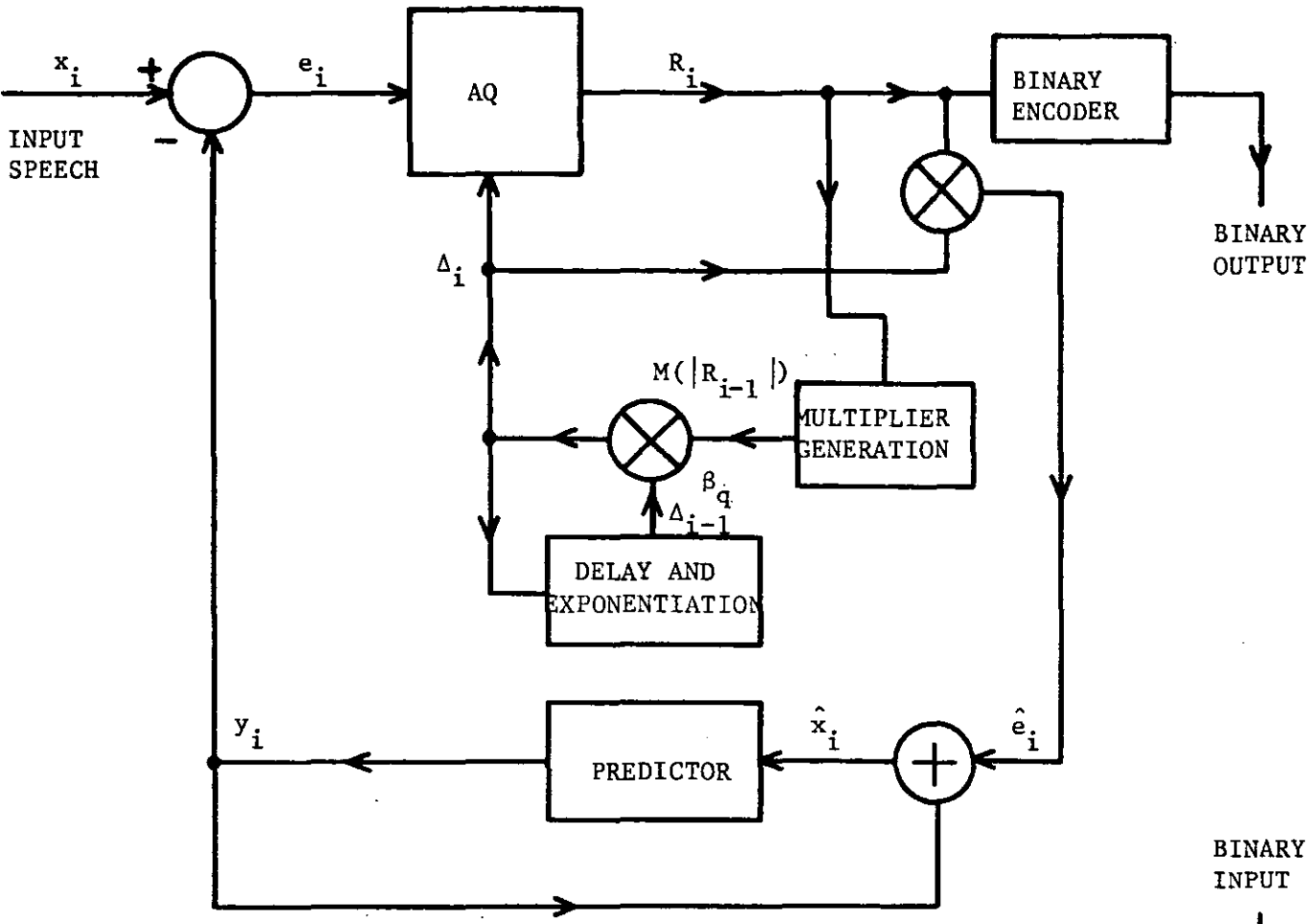
To define Δ, the rms value σ_d of the difference between adjacent samples in a block of W samples is calculated as

$$\sigma_d = \sqrt{\frac{1}{W} \sum_{i=2}^{W} (x_i - x_{i-1})^2} \; . \qquad (5.7)$$

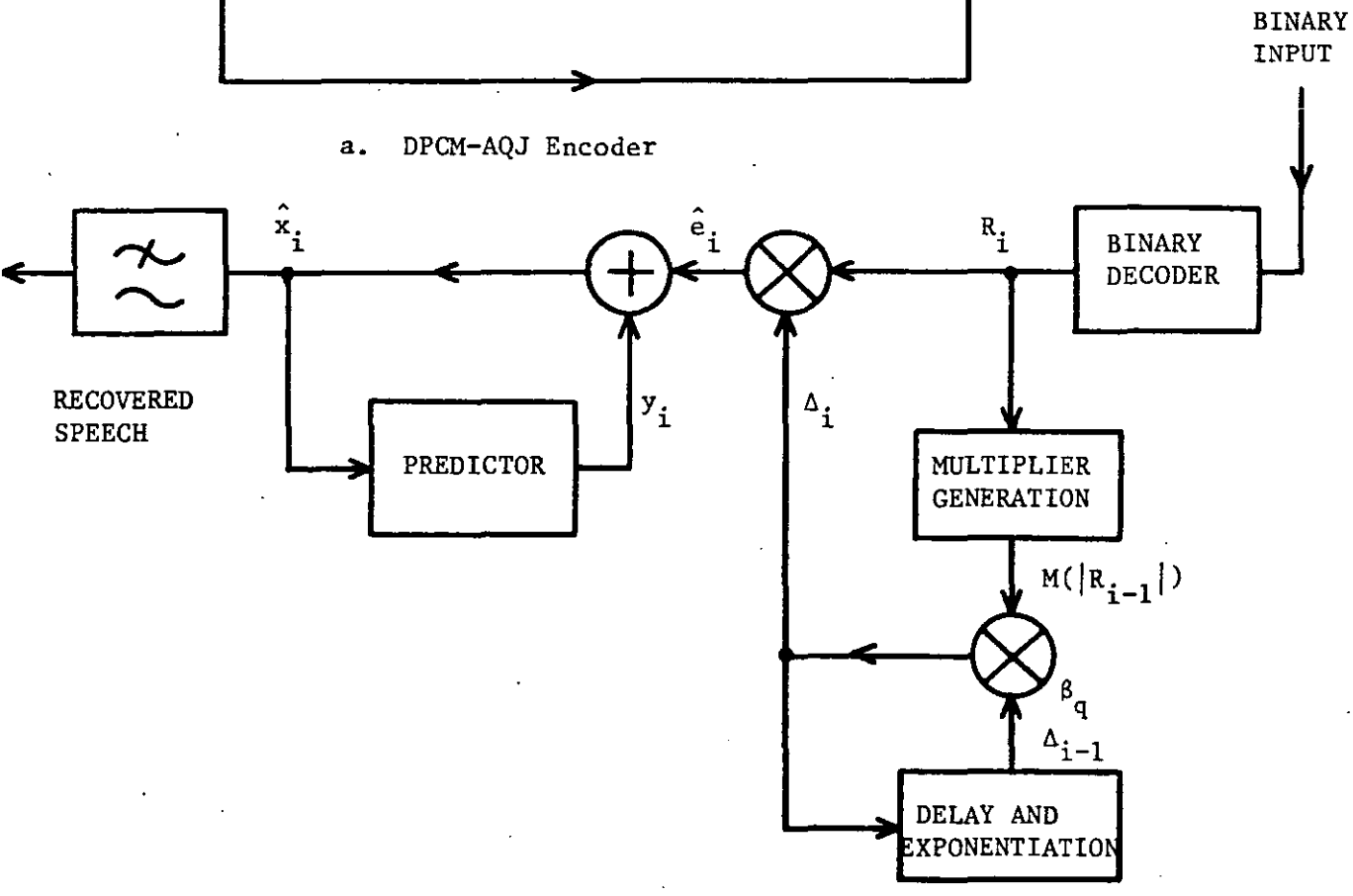
Δ is then formed at the output of the QS quantizer as

$$\Delta = Q\{(\alpha_q \sigma_d)\} \qquad (5.8)$$

where $Q\{(.)\}$ means the quantization of (.), see Figure 5.2. α_q is a step size optimizing coefficient whose value depends on the p.d.f. of the error signal, the number of quantization levels and the channel bit error rate, (BER).



a. DPCM-AQJ Encoder



b. DPCM-AQJ Decoder

FIGURE 5.1: DPCM-AQJ Codec used in simulation experiments

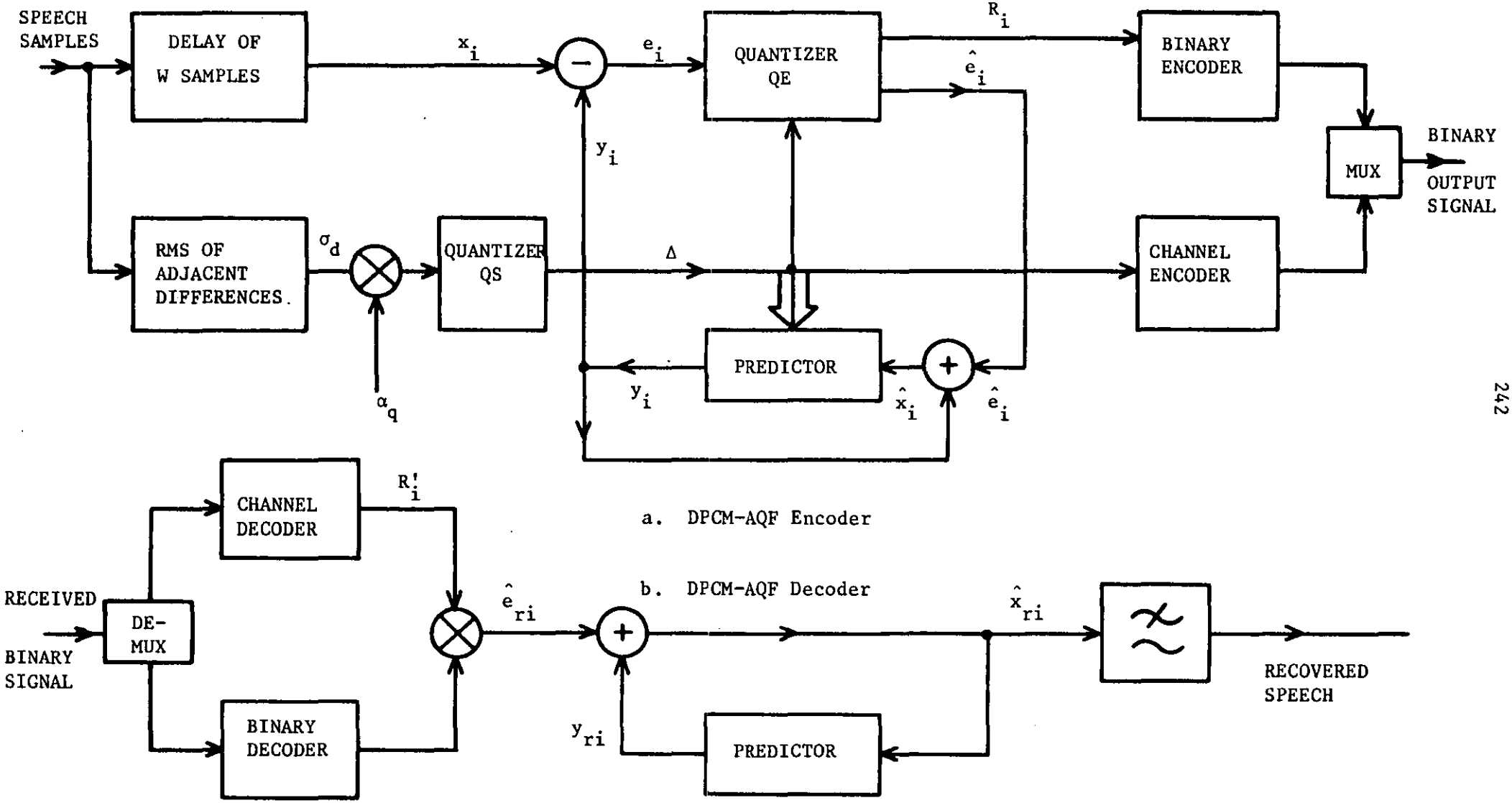


FIGURE 5.2: DPCM-AQF Codec used in Simulation Experiments

The DPCM quantizer QE (see Figure 5.2) accepts the sample (x_i-y_i) to be quantized, and the step size Δ, and produces a level number R_i and a quantization level $\hat{e}_i$, where the i sub-script means the i^{th} sampling instant in a particular block of samples, i.e.,

$$\hat{e}_i = \Delta R_i \ , \qquad i=1,2,\ldots,W \tag{5.9}$$

$$R_i = \pm\frac{(k-1)}{2} \ , \ k=1,2,\ldots,2^{b-1} \tag{5.10}$$

where b is the number of bits in the DPCM-AQF code word. The locally decoded output sample, $\hat{x}_i$ is formed by adding y_i and $\hat{e}_i$.

However, it is R_i, and not $\hat{e}_i$ that is binary encoded and multiplexed with the binary representation of Δ. At the receiver, the binary signals are de-multiplexed to yield R'_i and Δ. The $\hat{e}_{ri}$ and $\hat{x}_{ri}$ samples are then produced as shown in Figure 5.2(b). Notice that $\hat{e}_{ri}$ and y_{ri}, in the absence of transmission errors, are $\hat{e}_i=\hat{e}_{ri}$ and $y_i=y_{ri}$. Also, Δ may be assumed to be received with negligible error if sufficient channel protection coding is employed.

5.2.2 Predictors

The Fixed First Order Prediction, FFOP, was used in our DPCM experiments as a performance bench-mark. The value of its fixed coefficient, a_1, is the long term first shift autocorrelation coefficient of the speech signal, typically 0.91-0.94 and 0.85 for 10 kHz and 8 kHz sampling respectively.

Now we are to briefly summarize the SAP and SGEP sequential predictors since, when used in DPCM encoders, they take a slightly different form, than

that presented in Chapter IV. Also for simplicity, the number of coefficients in adaptive prediction will be limited to two.

The second-order SAP predictor updates its coefficients, at the ith sampling instant, according to the following expressions:

$$a_{i,1} = a_{i-1,1} + P_{i-1}(x)\hat{e}_{i-1}\hat{x}_{i-2} \tag{5.11}$$

$$a_{i,2} = a_{i-1,2} + P_{i-1}(x)\hat{e}_{i-1}\hat{x}_{i-3} \tag{5.12}$$

or in a vector form

$$\hat{A}_i = \hat{A}_{i-1} + P_{i-1}(x)\hat{e}_{i-1}\hat{X}_{i-1} \tag{5.13}$$

where $\hat{X}_{i-1} = [\hat{x}_{i-2},\hat{x}_{i-3}]^T$

and

$$P_{i-1}(x) = \frac{A}{B + \frac{1}{M}\sum_{j=i-M-1}^{i-1}\hat{x}_j^2} \quad . \tag{5.14}$$

Notice that Equations (5.11)-(5.13) are of the same form with Equations (4.7-4.8) of Chapter IV, except that the x_{i-1} and e_{i-1} samples have been replaced by the decoded sample $\hat{x}_{i-1}$ and the quantized sample, $\hat{e}_{i-1}=Q(x_{i-1}-y_{i-1})$, respectively. A and B are system parameters and their values will be quoted in the results section.

The $P_{i-1}(x)$, given by Equation (5.14) with M=N, is used in DPCM-AQJ systems while for DPCM-AQF applications, the value of $P_{i-1}(x)=P$, is inversely proportional to the mean square value of the differences between adjacent speech samples, σ_d, calculated over the block of W samples, see Equation (5.7). That is,

$$P = \frac{A}{B + (\Delta/\alpha_q)^2} . \tag{5.15}$$

A and B are known at the receiver while the step size Δ is transmitted. As Δ changes for each block so does P. Consequently, P remains constant for W samples and because of its dependency on the channel protected Δ the codec's performance in the presence of channel errors is enhanced.

The second-order SGEP algorithm,[116] at ith sampling instant, forms four predictions $y_{i-1,1}, y_{i-1,2}, y_{i-1,3}$ and $y_{i-1,4}$ which are generated from $\hat{x}_{i-2}$ and $\hat{x}_{i-3}$. These intermediary predictions are

$$y_{i-1,1} = (a_{i-1,1}+s_1)\hat{x}_{i-2} + a_{i-1,2}\hat{x}_{i-3} \tag{5.16}$$

$$y_{i-1,2} = (a_{i-1,1}-s_1)\hat{x}_{i-2} + a_{i-1,2}\hat{x}_{i-3} \tag{5.17}$$

$$y_{i-1,3} = a_{i-1,1}\hat{x}_{i-2} + (a_{i-1,2}+s_2)\hat{x}_{i-3} \tag{5.18}$$

$$y_{i-1,4} = a_{i-1,1}\hat{x}_{i-2} + (a_{i-1,2}-s_2)\hat{x}_{i-3} \tag{5.19}$$

where s_1 and s_2 are system parameters, and $s_2<s_1$, see Equation (4.26). Notice that s_1 has been added and subtracted from $a_{i-1,1}$ to give $y_{i-1,1}$ and $y_{i-1,2}$ while a smaller change of $\pm s_2$ has been made to $a_{i-1,2}$ to give $y_{i-1,3}$ and $y_{i-1,4}$.

We now form the moduli of these prediction errors

$$FU_{i-1,j} = |\hat{x}_{i-1} - y_{i-1,j}| \ , \ j=1,2,\ldots,4 \tag{5.20}$$

and then compute

$$\Lambda_{i-1,1} = FU_{i-1,1} - FU_{i-1,2} \tag{5.21}$$

$$\Lambda_{i-1,2} = FU_{i-1,3} - FU_{i-1,4} \tag{5.22}$$

As $FU_{i-1,1}$ and $FU_{i-1,2}$ are the moduli of the prediction error when $a_{i-1,1}$ is increased and decreased by s_1 respectively, it follows that if $\Lambda_{i-1,1}>0$ then $a_{i-1,1}$ should be decreased and vice versa. Similar remarks apply for $\Lambda_{i-1,2}$ and $a_{i-1,2}$.

Consequently, the two coefficients specified by k=1,2, are updated according to

$$a_{i,k} = a_{i-1,k} - \Lambda_{i-1,k} \cdot \frac{P_{i-1}(x)}{k^{\alpha}} \tag{5.23}$$

where $P_{i-1}(x)$ is given by Equation (5.14) with M=N. The term k^{α}, where α is just less than unity, results in a smaller modification to a_2 than a_1 and this improves the prediction accuracy of the algorithm.

In a ADPCM-AQF coder, $P_{i-1}(x)$ is replaced by P as defined by Equation (5.15). Having determined $a_{i,1}$ and $a_{i,2}$, the decoded output x_i can be found by using

$$\hat{x}_i = a_{i,1}\hat{x}_{i-1} + a_{i,2}\hat{x}_{i-2} + \hat{e}_i \tag{5.24}$$

The schematic diagram representing both the second order SAP and SGEP adaptive predictors is shown in Figure 5.3.

Further to this brief explanation of the second-order SGEP scheme, we are to demonstrate the differences between SGEP and SAP that minimizes the absolute error, FU=<$|e_i|$>, (see Appendix E).

Using Equation (5.21),

$$\Lambda_{i-1,1} = |\hat{x}_{i-1} - a_{i-1,1}\hat{x}_{i-2} - a_{i-1,2}\hat{x}_{i-3} - s_1\hat{x}_{i-2}|$$
$$- |\hat{x}_{i-1} - a_{i-1,1}\hat{x}_{i-2} - a_{i-1,2}\hat{x}_{i-3} + s_1\hat{x}_{i-2}| \tag{5.25}$$

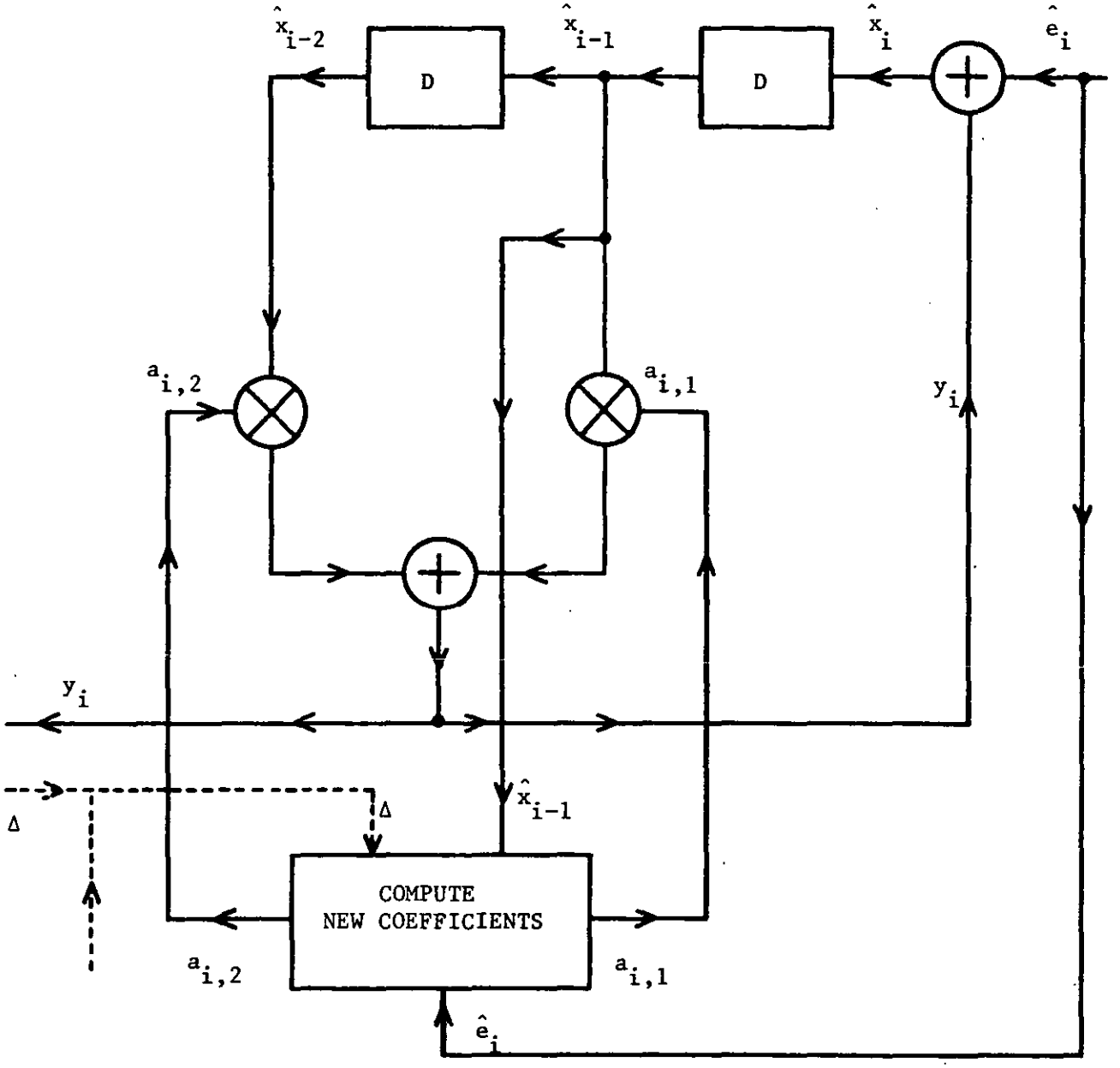


FIGURE 5.3: Two-coefficient adaptive predictor,
-------- for ADPCM-AQF codec

and since

$$\hat{e}_{i-1} = \hat{x}_{i-1} - a_{i-1,1}\hat{x}_{i-2} - a_{i-1,2}\hat{x}_{i-3} \tag{5.26}$$

Equation (5.25) is rewritten as

$$\Lambda_{i-1,1} = |\hat{e}_{i-1} - s_1\hat{x}_{i-2}| - |\hat{e}_{i-1} + s_1\hat{x}_{i-2}|. \tag{5.27}$$

Similarly,

$$\Lambda_{i-1,2} = |\hat{e}_{i-1} - s_2\hat{x}_{i-3}| - |\hat{e}_{i-1} + s_2\hat{x}_{i-3}|. \tag{5.28}$$

Letting "w" be $\hat{e}_{i-1}$ and "v" be $s_1\hat{x}_{i-2}$ or $s_2\hat{x}_{i-3}$, equation (5.27) is rewritten as

$$\Lambda_{i-1,1} = |w-v| - |w+v| \tag{5.29}$$

Equation (5.29) can be analysed with the aid of Figure 5.4.[125]

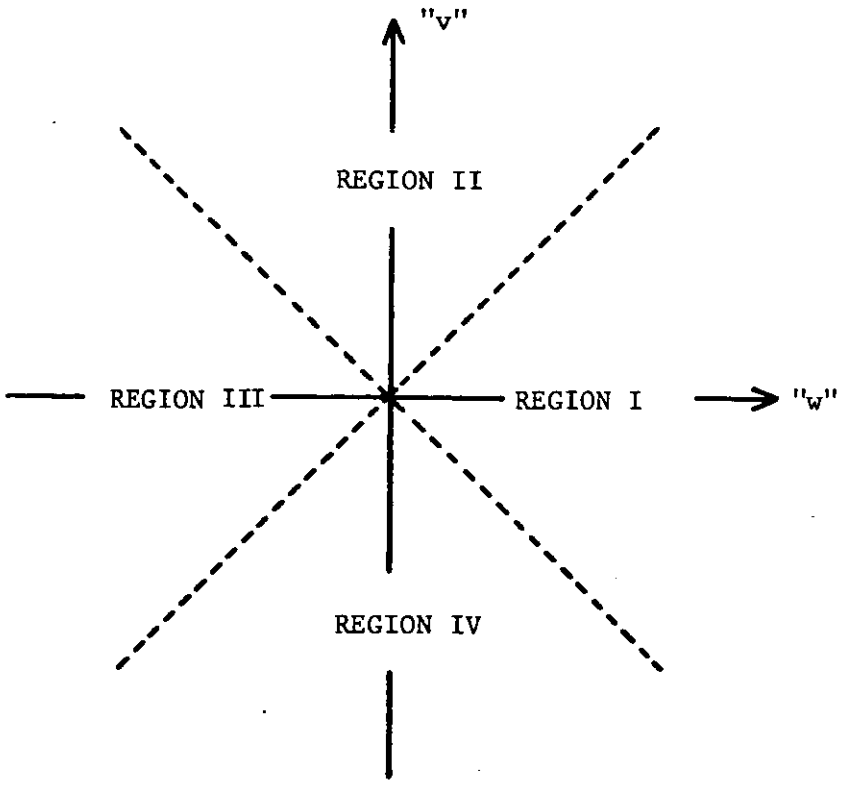


FIGURE 5.4: Analysis of $\Lambda_{i-1,k}$

That is,

In REGION I: $$|w-v|-|w+v| = -2v \tag{5.30}$$

In REGION II: $$|w-v|-|w+v| = -2w \tag{5.31}$$

In REGION III: $$|w-v|-|w+v| = 2v \tag{5.32}$$

In REGION IV: $$|w-v|-|w+v| = 2w \tag{5.33}$$

Hence, in REGIONS I and III, $|w|>|v|$ and

$$|w-v|-|w+v| = -2v\ \mathrm{sgn}(w) \tag{5.34}$$

Similarly, in REGIONS II and IV, $|v|>|w|$ and

$$|w-v|-|w+v| = -2w\ \mathrm{sgn}(v). \tag{5.35}$$

Thus Equation (5.27) yields

$$\Lambda_{i-1,1} = \begin{cases} -2s_1\hat{x}_{i-2}\mathrm{sgn}(\hat{e}_{i-1}), & \text{if } |\hat{e}_{i-1}|>|s_1\hat{x}_{i-2}| \\ -2\hat{e}_{i-1}\mathrm{sgn}(s_1\hat{x}_{i-2}), & \text{if } |s_1\hat{x}_{i-2}|>|\hat{e}_{i-1}| \end{cases} \tag{5.36}$$

while Equation (5.28) gives

$$\Lambda_{i-1,2} = \begin{cases} -2s_2\hat{x}_{i-3}\mathrm{sgn}(\hat{e}_{i-1}), & \text{if } |\hat{e}_{i-1}|>|s_2\hat{x}_{i-3}| \\ -2\hat{e}_{i-1}\mathrm{sgn}(s_2\hat{x}_{i-3}), & \text{if } |s_2\hat{x}_{i-3}|>|\hat{e}_{i-1}|. \end{cases} \tag{5.37}$$

Finally, substitution of Equations (5.36) and (5.37) in Equation (5.23), provides the adaptation equations of SGEP. When $|\hat{e}_{i-1}|>|s_1\hat{x}_{i-2}|$ and $|\hat{e}_{i-1}|>|s_2\hat{x}_{i-3}|$, SGEP is defined by

$$a_{i,1} = a_{i-1,1} + 2P_{i-1}(x).s_1.\hat{x}_{i-2}.\mathrm{sgn}(\hat{e}_{i-1}) \tag{5.38}$$

$$a_{i,2} = a_{i-1,2} + 2P_{i-1}(x).s_2.2^{-\alpha}.\hat{x}_{i-3}\mathrm{sgn}(\hat{e}_{i-1}). \tag{5.39}$$

Appendix E however, shows the SAP equations that minimize the

absolute error, FU=<$|e_i|$>≃$|e_i|$, to be

$$a_{i,1} = a_{i-1,1} + h.\hat{x}_{i-2}.\mathrm{sgn}(\hat{e}_{i-1}) \tag{5.40}$$

$$a_{i,2} = a_{i-1,2} + h.\hat{x}_{i-3}.\mathrm{sgn}(\hat{e}_{i-1}). \tag{5.41}$$

Thus the basic design of the adaptation process of SGEP has no significant departure from that of Cummiskey's[62] sequential algorithm with FU=$|e_i|$, if $|\hat{e}_{i-1}|>|s_1\hat{x}_{i-2}|$ and $|\hat{e}_{i-1}|>|s_2\hat{x}_{i-3}|$.

However, when the $|s_1\hat{x}_{i-2}|>|\hat{e}_{i-1}|$ or $|s_2\hat{x}_{i-3}|>|\hat{e}_{i-1}|$ inequalities are satisfied, SGEP assumes a very different form. That is,

$$a_{i,1} = a_{i-1,1} + 2P_{i-1}(x).\hat{e}_{i-1}\mathrm{sgn}(s_1\hat{x}_{i-2}) \tag{5.42}$$

$$a_{i,2} = a_{i-1,2} + 2P_{i-1}(x).2^{-\alpha}.\hat{e}_{i-1}.\mathrm{sgn}(s_2\hat{x}_{i-3}) \ . \tag{5.43}$$

Equations (5.42) and (5.43) indicate the degression from the gradient algorithm that minimizes FU=$|e_i|$.

The frequency of occurrance of Equations (5.38)-(5.39) or Equations (5.42-5.43) in ADPCM-AQF-SGEP coder for the same speech data used in Chapter IV is presented in Table 5.1.

COEFFICIENTS	PERCENTAGE OF OCCURRANCE OF INEQUALITIES $\|\hat{e}_{i-1}\|>\|s_1\hat{x}_{i-2}\|$ OR $>\|s_2\hat{x}_{i-3}\|$	PERCENTAGE OF OCCURRANCE OF INEQUALITIES $\|\hat{e}_{i-1}\|<\|s_1\hat{x}_{i-2}\|$ OR $<\|s_2\hat{x}_{i-3}\|$
$a_{i,1}$	33%	67%
$a_{i,2}$	38%	62%

TABLE 5.1: Frequency of occurrance of SGEP adaptation equations in ADPCM-AQF-SGEP encoder, b=4, W=256, input level=-5 dB

As is seen in this table, during the adaptation of $a_{i,1}$ and $a_{i,2}$, 67% and 62% of the time, diversions from the gradient algorithm with FU=$|e_i|$ are attained, respectively.

5.3 COMPUTER SIMULATION RESULTS OF DPCM-AQJ SPEECH CODECS EMPLOYING FFOP, SAP OR SGEP

We will now present the results of our DPCM-AQJ study in terms of long-term SNR, SNRSEG(dB) and waveform plots. The input speech signal used in the simulation experiments was the sentence *"An apple a day keeps the doctor away"*, bandlimited to 3.4 kHz and sampled at 10 kHz.

Figure 5.5 shows the SNR(dB) plot of a 4 bits per sample DPCM-AQJ-FFOP coder, for different power values of the input speech signal. The ratio of the maximum to the minimum quantization step size, see Equation (5.5), was fixed to 128 while the value of the quantization *"leakage"* constant, β_q, was set to unity, see Equation (5.4). The value of the fixed prediction coefficient, a_1, used in these experiments was 0.946.

Table 5.2 shows the SNR(dB) and SNRSEG(dB) values produced by this codec, at the centre of its dynamic range, when different β_q constants are employed to improve the performance of the system in the presence of transmission errors. Table 5.2 indicates that the best signal-to-noise ratio performance of the DPCM-AQJ-FFOP codec is obtained when β_q=31/32. Also, the SNRSEG(dB) for β_q=1 is less than that for β_q=31/32 by 1.4 dB, at 40 Kb/s. These results differ from what has been reported by others[67,123] whereby β_q=1 is producing the maximum SNR. The difference however, may be attributed to that the step size multipliers, M_i, employed in our experiments have been formulated from speech sampled at 8 kHz whereas our DPCM-AQJ-FFOP codecs are processing speech at the rate of 10 k samples

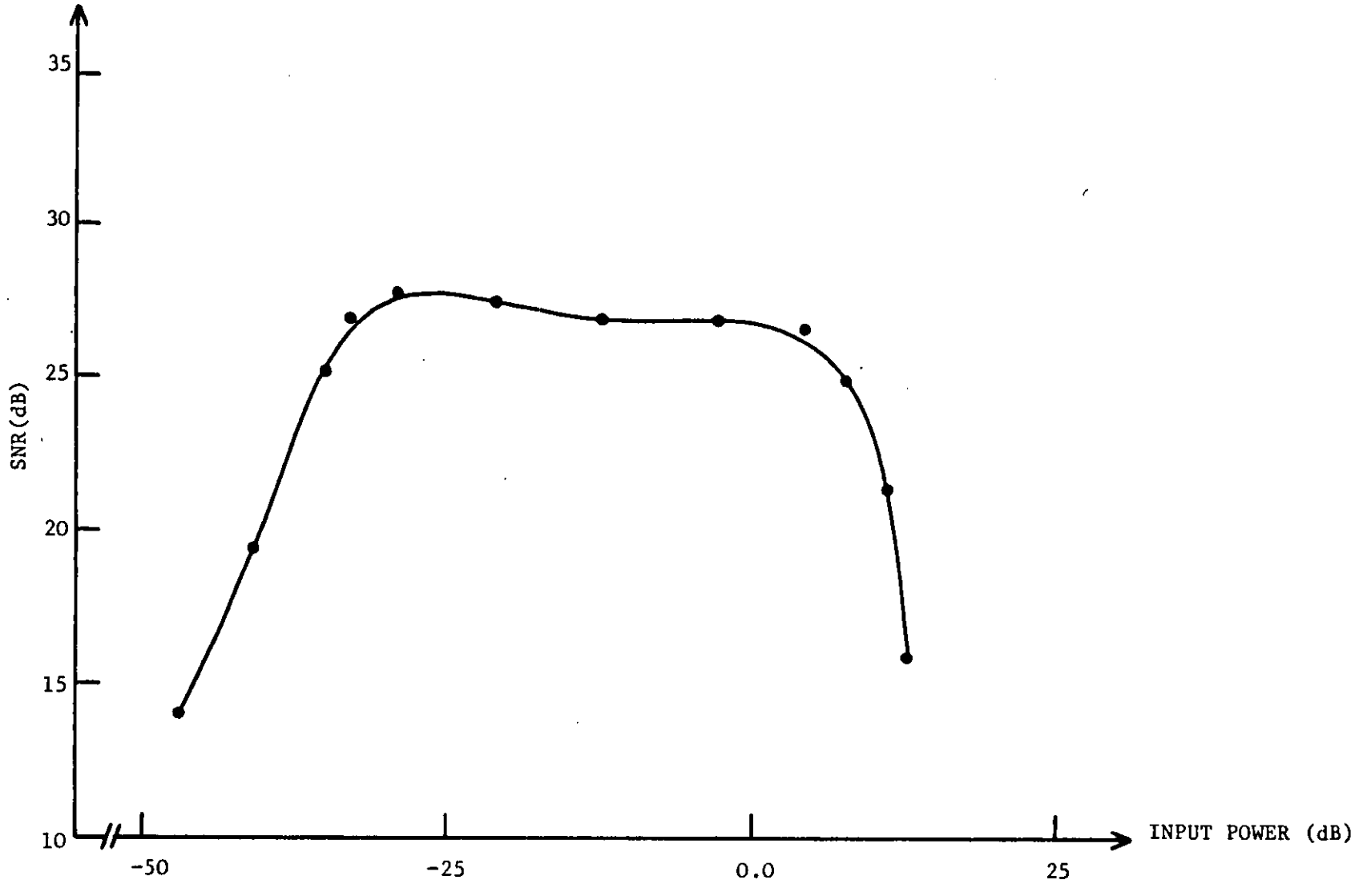


FIGURE 5.5: Performance of a 4 bits/sample DPCM-AQJ (non-robust)-FFOP

β_q	SNRSEG(dB) W=128	SNR(dB)
1/2	19.57	21.84
7/8	28.44	28.63
15/16	33.31	29.04
31/32	33.48	29.19
63/64	33.25	27.96
127/128	33.14	27.45
1.0	32.86	26.70

TABLE 5.2: Variation of SNRSEG(dB)-SNR(dB) with β_q at 40 Kb/s and at input power of -4.26 dB

per sec. The variation of SNR(dB) with respect to the input power level for β_q=31/32 is plotted in Figure 5.6.

Next, the SAP and SGEP prediction schemes were employed with DPCM-AQJ. The locally decoded signal applied to these predictors is of course, contaminated with quantization noise. Thus the predictor parameters differ from those used previously, see Section 4.7, and they are selected to peak the SNR at the centre of the coders dynamic range. The optimum prediction parameters for SGEP and SAP operating in the 40 Kb/s DPCM-AQJ codec were A=10.0, B=20.0, M=N, D=6.0, α/β=4.0, β=1/5 and A=0.05, B=40.0, respectively. No attempt was made to optimize again these parameters for different conditions, such as transmission data rate and the order of predictor. It was observed from the computer simulations that for low values of input power (<-30 dB), the SNR(dB) is increasing for all coders, at approximately 6 dB/octave as AQJ operated frequently with its minimum step-size, and rarely in its adaptive mode. As the input power increased, the prediction gains of SAP and SGEP over FFOP manifested as differences in coder SNR(dB). SNR(dB) gains of 6 and 3.5 dB's were achieved by the coder using SGEP when compared to DPCM-AQJ employing FFOP and SAP, respectively over a dynamic range, DR, of 30 dB. When the input level was sufficiently large for slope overload noise to frequently occur, the predictors operated on distorted speech signals, and the SNR rapidly deteriorated.

Figure 5.7 shows the SNR(dB) variation of 4 bits ADPCM as a function of prediction order, N. The gain in SNR(dB) by using SGEP instead of SAP

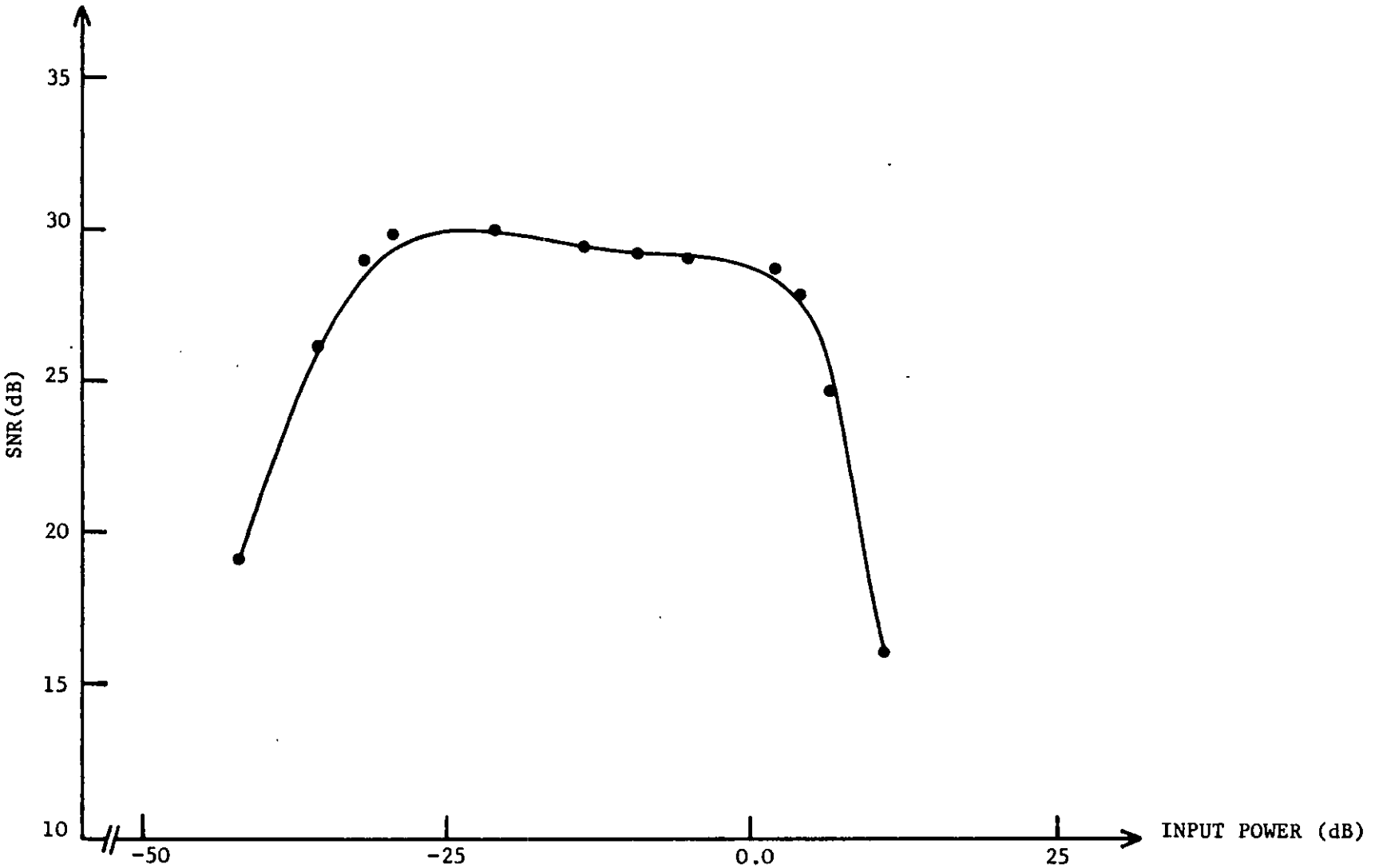


FIGURE 5.6: Performance of a 4 bits/sample DPCM-AQJ (Robust)-FFOP; β_q=31/32

is approximately 1 dB smaller than the prediction gain shown in Figure 5.8 where the predictors operated on quantization free speech samples. When the ADPCM bit-rate was reduced to about 16 Kb/s (3-level AQJ, $f_s.\log_2 3 \simeq 16$ Kb/s) the curves had similar shape with those in Figure 5.7, except that the SNR gain of SGEP over SAP was reduced to only 1 dB.

The improvement in SNR, achieved by increasing the bit-rate for the ADPCM-AQJ coder employing SGEP or SAP with N=2, at an input power level of -10 dB, is shown in Figure 5.9. The SNR gain due to employing SGEP instead of SAP in DPCM-AQJ, is of the order of 2 to 3.5 dB's, as the number of bits per code word, "b", is increased from 2 to 4. This improved performance can also be observed from the time waveforms of Figures 5.10-5.11. The comparison of curves (iii) and (iv) of Figure 5.10 with those of Figure 5.11 reveals that ADPCM employing SGEP in its feedback loop produces the smaller prediction error, and therefore the smaller quantization noise.

Further, the prediction order was increased from 2 to 6 and the SNR(dB) performance of the ADPCM-AQJ employing both adaptive prediction schemes was measured for different transmission bit-rates. The results are shown in Table 5.3. It is seen that as the predictor order increases ADPCM-AQJ-SGEP offers always a better SNR than ADPCM-AQJ-SAP. Moreover, when the order of SGEP predictor increases from 2 to 4, the SNR(dB) of the ADPCM-AQJ-SGEP encoder changes by 0.9 dB, at 40 Kb/s, and 1.4 dB's as N varies from 2 to 6. Consequently, although the adaptation process of SGEP tends to be complex, its SNR performance at high bit rates is not considerably degraded by using it in its simplest form, namely with two coefficients.

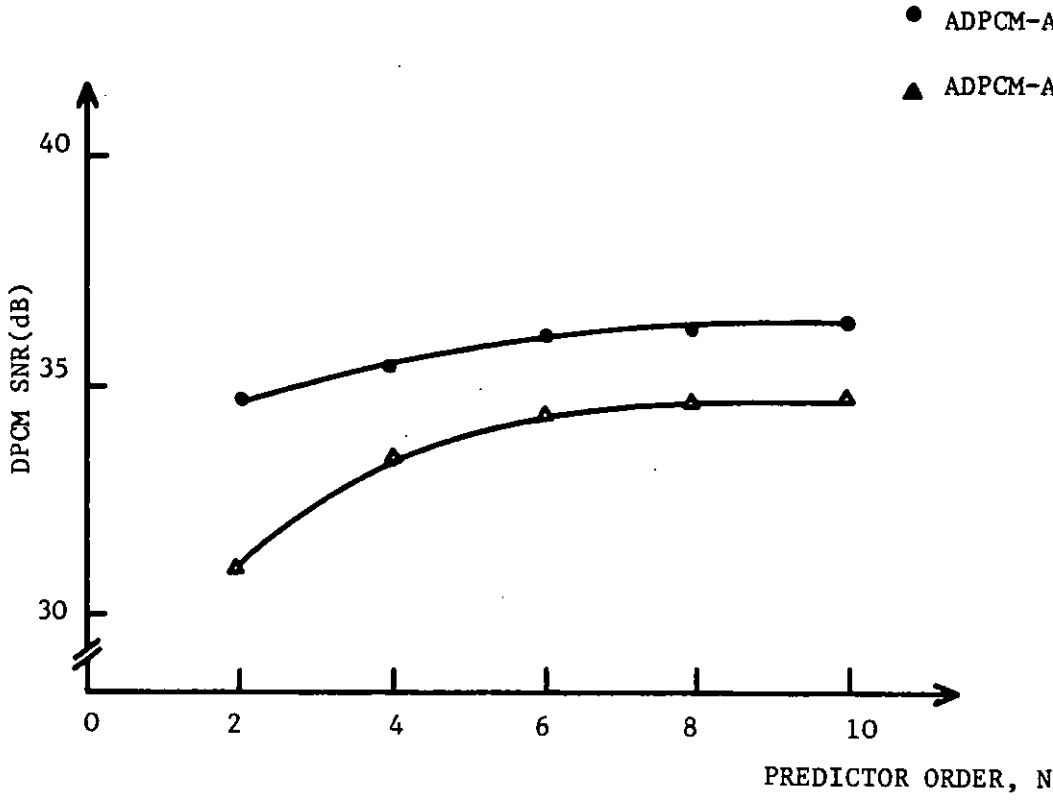


FIGURE 5.7: Variation of SNR(dB) versus predictor order, N for 4-bit ADPCM-AQJ having SAP and SGEP when input power is -10 dB

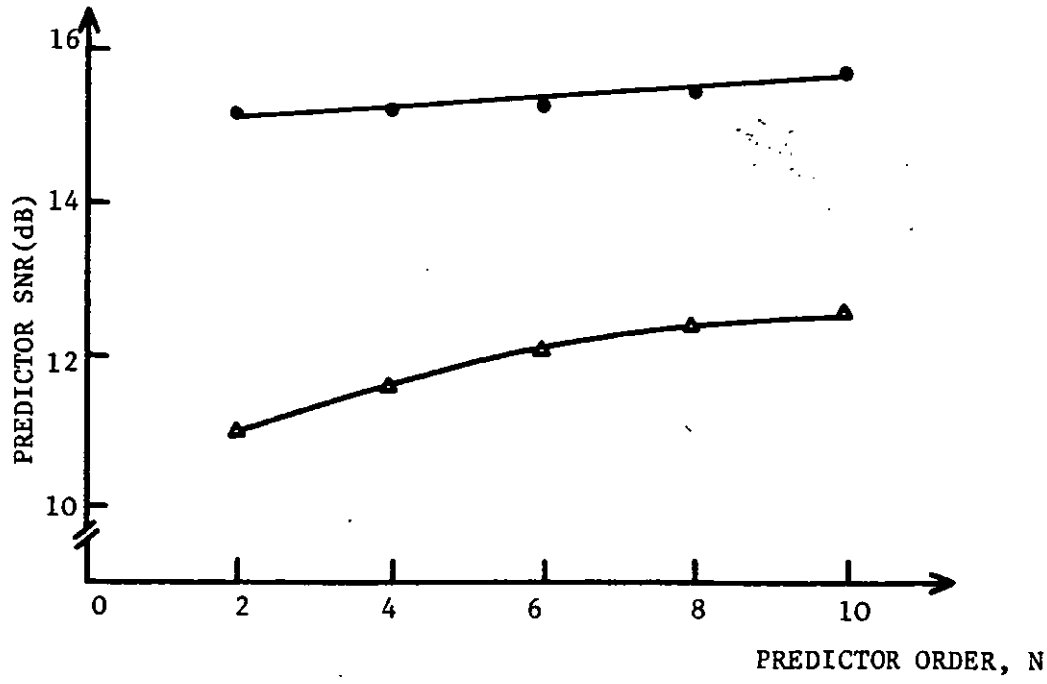


FIGURE 5.8: Variation of SNR as a function of predictor order, N for SAP and SGEP from Figure 4.11.

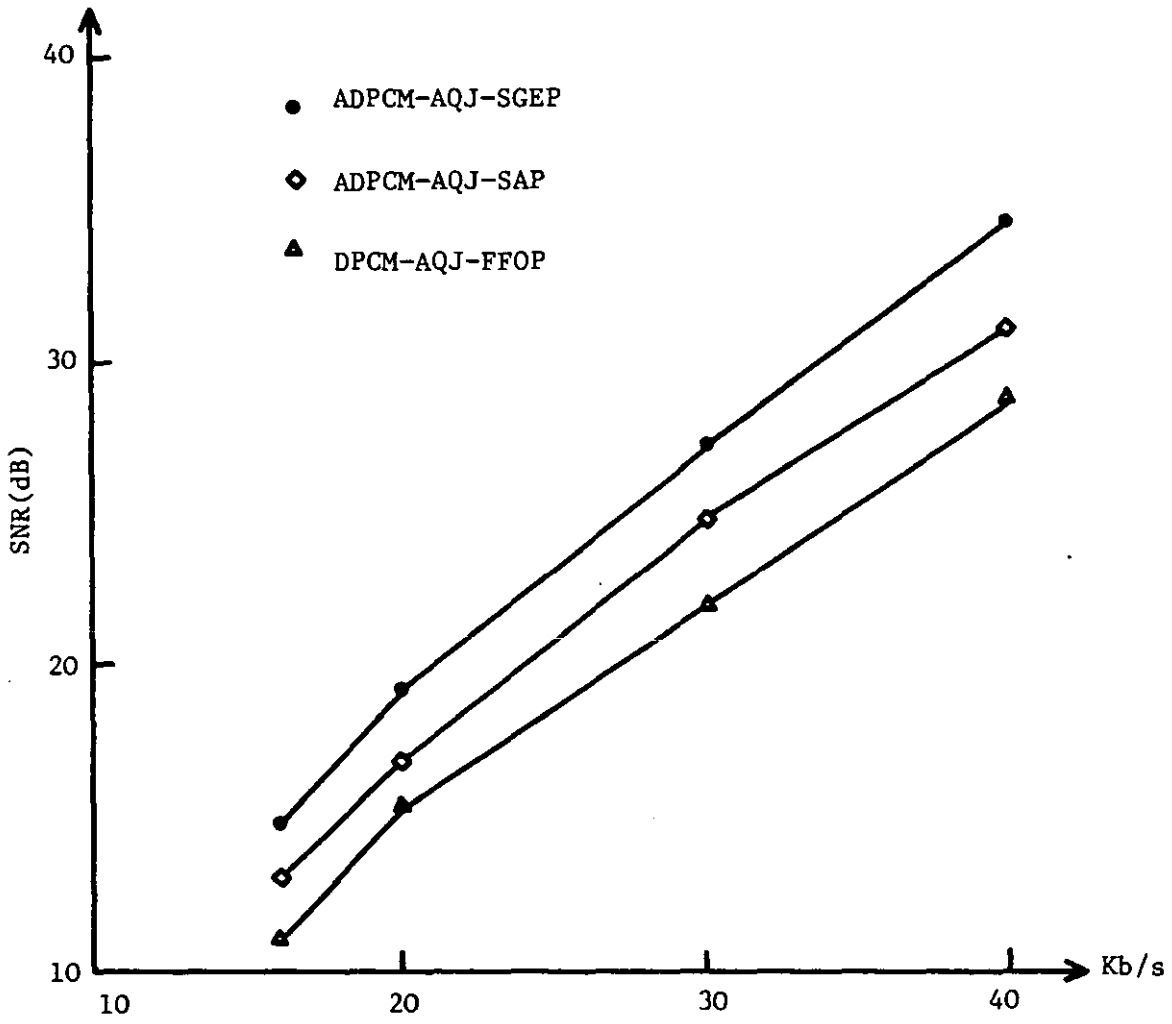
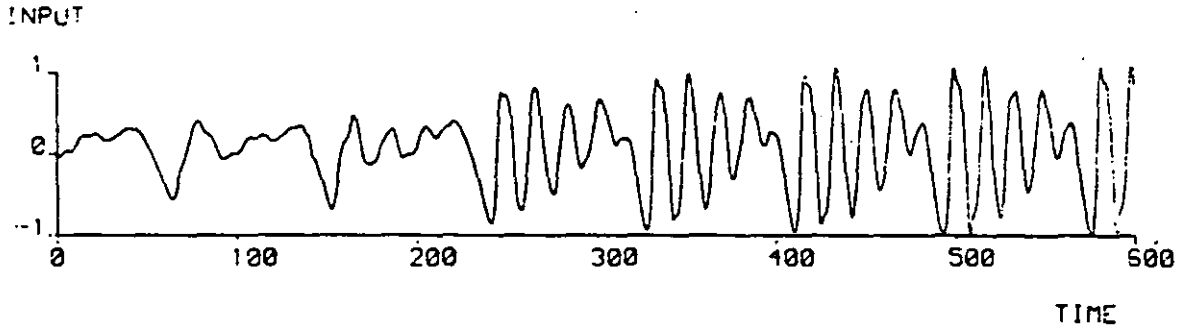
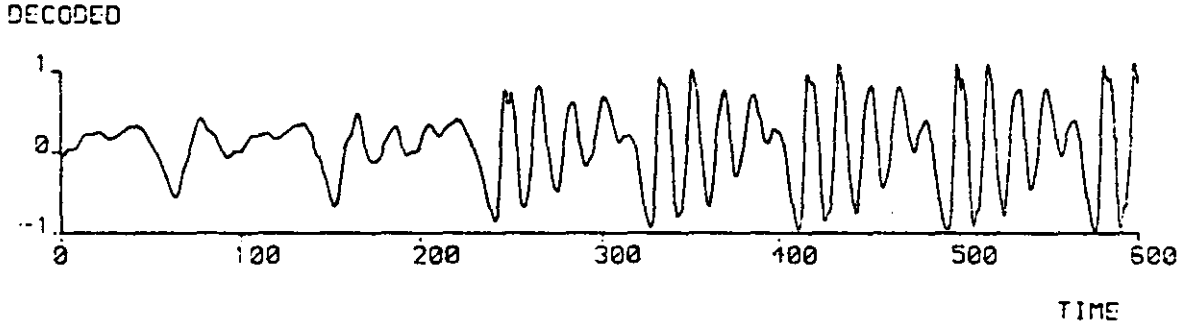


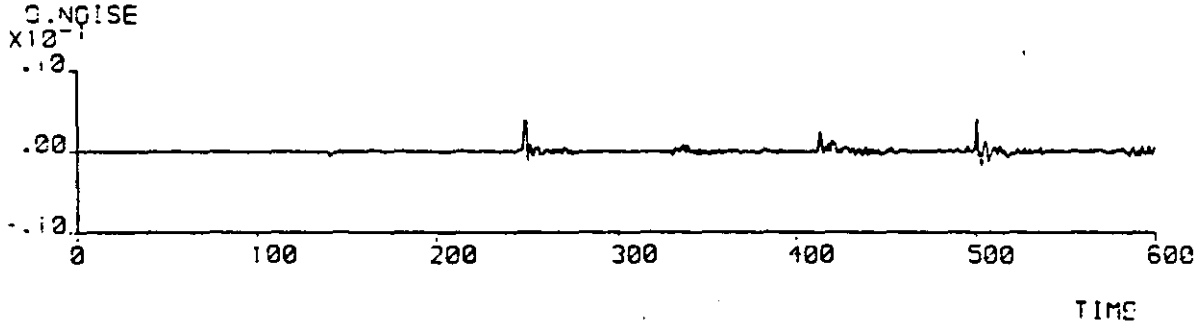
FIGURE 5.9: Variation of SNR with transmission bit rate, N=2, input power level = -10 dB



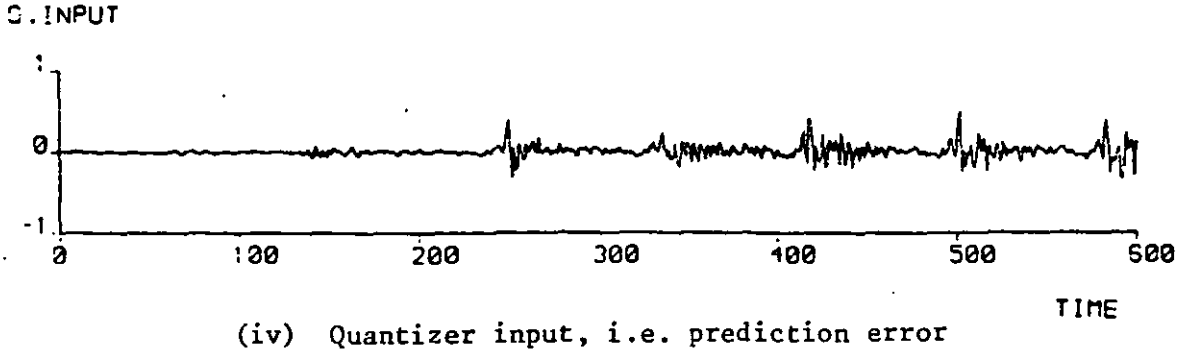
(i) Segment of original speech



(ii) ADPCM-AQJ-SGEP encoded speech

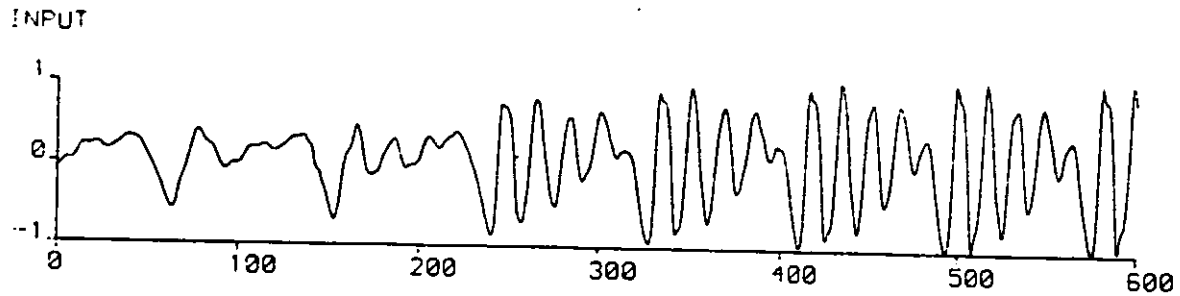


(iii) Quantization noise

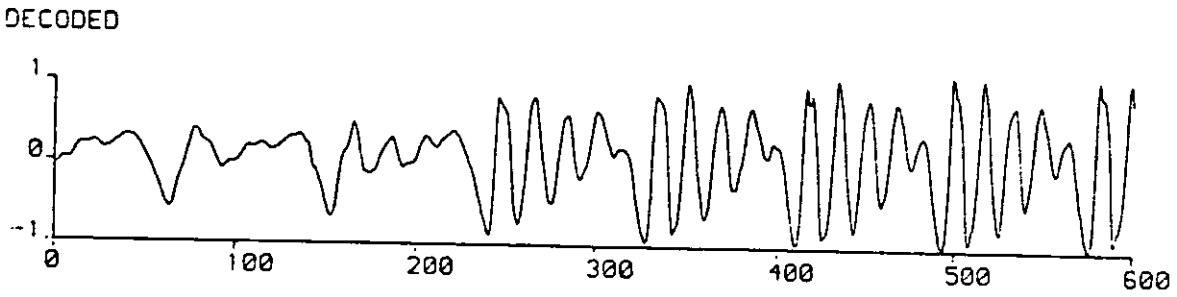


(iv) Quantizer input, i.e. prediction error

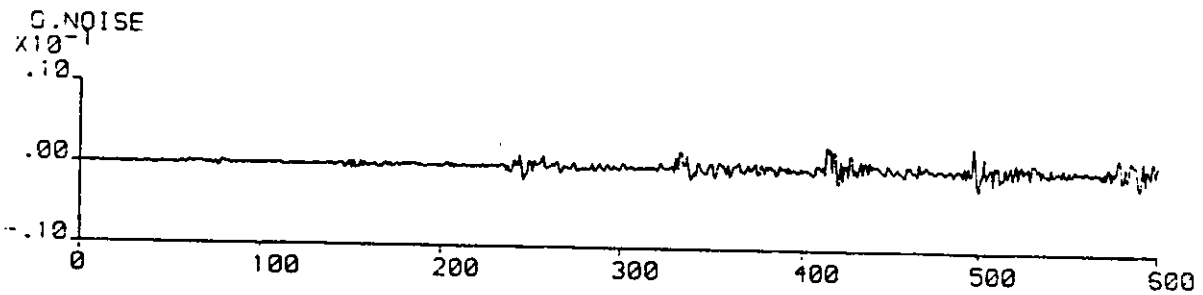
FIGURE 5.10: Time waveforms for ADPCM-AQJ-SGEP, b=4, N=2



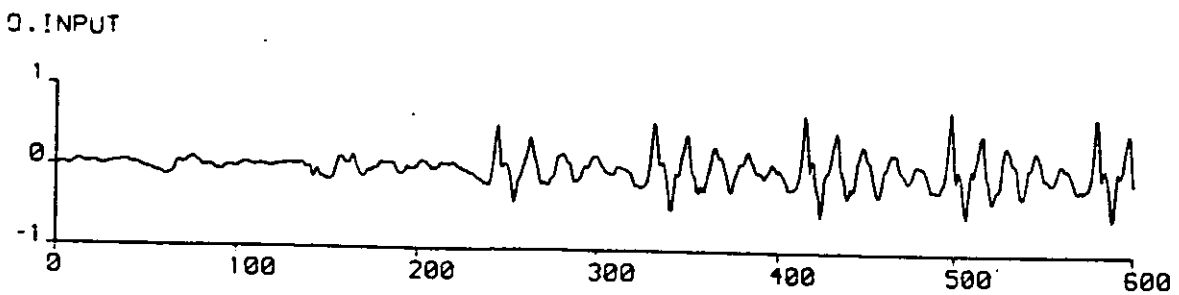
(i) Segment of original speech



(ii) ADPCM-AQJ-SAP encoded speech



(iii) Quantization noise



(iv) Quantizer input, i.e. prediction error

FIGURE 5.11: Time waveforms for ADPCM-AQJ-SAP, b=4, N=2

ORDER OF PREDICTOR	TYPE OF PREDICTOR	SNR(dB)			
		16 Kb/s	20 Kb/s	30 Kb/s	40 Kb/s
N=1	FFOP	11.24	15.24	22.00	28.75
N=2	SAP	13.04	16.48	24.64	31.11
	SGEP	14.85	19.02	27.69	34.72
N=4	SAP	14.74	17.81	27.00	33.70
	SGEP	15.94	20.61	28.25	35.63
N=6	SAP	15.40	18.75	27.12	34.34
	SGEP	16.65	21.60	29.36	36.10

TABLE 5.3: Variation of prediction order, N with bit-rates for ADPCM-AQJ-SAP/SGEP

Finally, the stability of the sequentially adaptive prediction algorithms used in our simulation experiments was checked by observing the roots of their characteristic equation, I(z). I(z) for the second-order predictor at the ith sampling instant is defined as[126]

$$I_i(z) = z^2 - a_{i,1}z - a_{i,2} \tag{5.44}$$

If the roots of $I_i(z)$ are located inside the unit circle in the z-plane, the system is *"stable"*. In sequential algorithms, because a new set of coefficients is computed at every sampling instant, $I_i(z)$ is also computed in a sequential manner. It should be emphasized that at some sampling instants, a system can be unstable. However, as long as the duration of unstability is not long enough - if the codec employing one of the sequential prediction schemes recovers itself fast from unstable region - there is no major problem. For lower-order predictors (N=2), Schur-Cohn[127] stability criterion can be used to determine whether the system is stable or not. In Schur-Cohn criterion, the necessary and sufficient conditions for the roots of $I_i(z)=0$ lying inside the unit circle are related to the following inequalities:

$$|I_i(0)| = |a_{i,2}| < 1 \tag{5.45}$$

$$I_i(1) = 1 - a_{i,1} - a_{i,2} > 0 \tag{5.46}$$

$$I_i(-1) = 1 + a_{i,1} - a_{i,2} > 0 \tag{5.47}$$

The Equations (5.45)-(5.47) define the stable region for $a_{i,1}$ and $a_{i,2}$. Figure 5.12, PATH 1 presents a simple flow-chart in order to check stability and correct it. The simulation run conducted for the ADPCM-AQJ-SGEP encoder having second-order predictor at 40 Kb/s shows that unstable

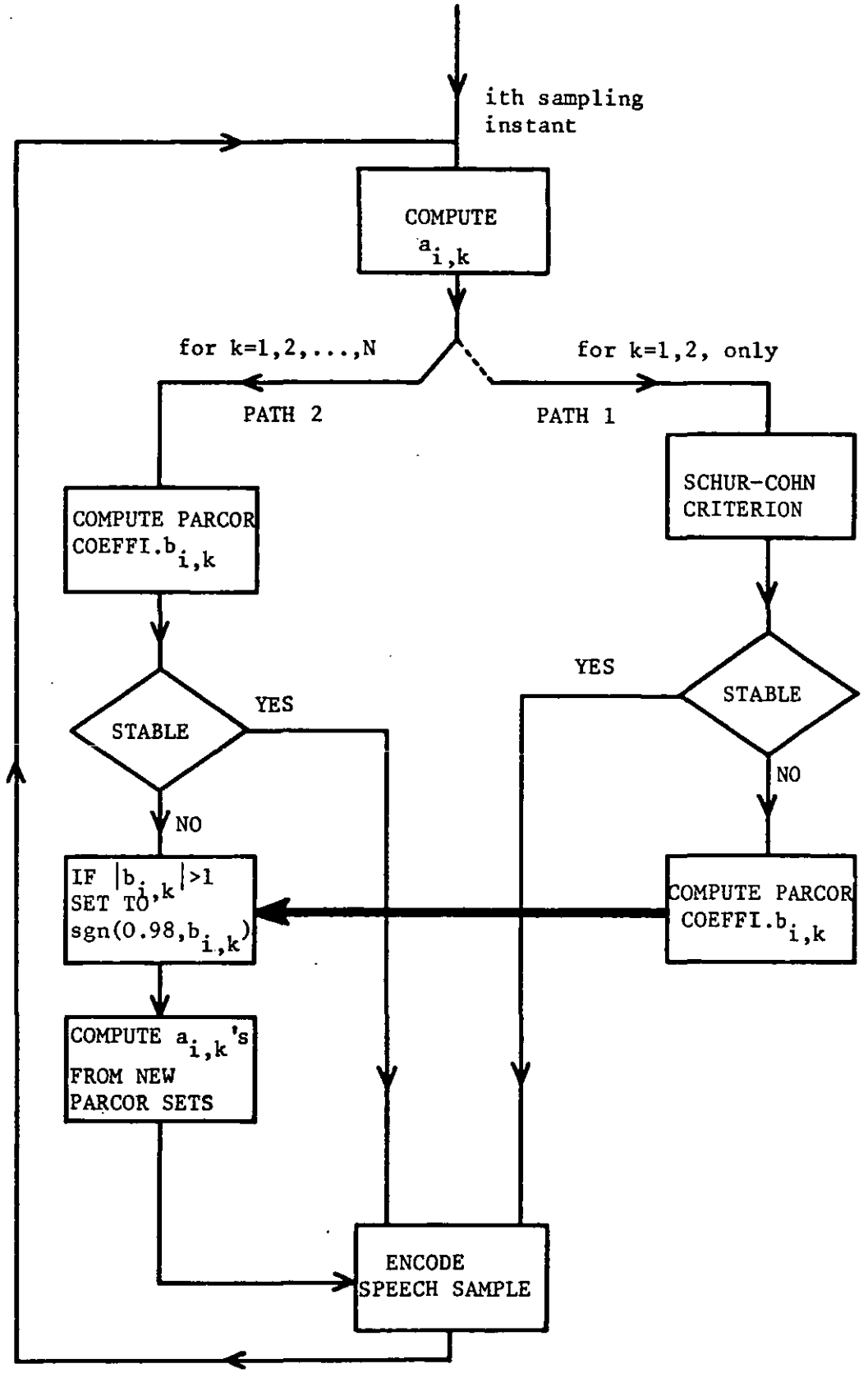


FIGURE 5.12: A schematic flow-chart for stability criteria

coefficients were generated by the SGEP algorithms about 9% of the time. For higher-order predictor, the above criterion may only be applied when the predictor is decomposed into cascaded stages each having the second-order predictors. A different method for checking the stability is to compute the PARCOR coefficients, $b_{i,k}$'s, and can be applied to any order predictors. As described in Appendix F, the set of prediction coefficients, $a_{i,k}$'s, are initially transformed into PARCOR domain.[26] The observation of $b_{i,k}$'s indicates the system stability, i.e. if $|b_{i,k}|>1$ the system is unstable. Then by suitable manipulation, $b_{i,k}$'s are adjusted to be ≤ 1 and transformed back to $a_{i,k}$'s.

5.4 COMPUTER SIMULATION RESULTS OF DPCM-AQF SPEECH CODECS EMPLOYING FFOP, SAP OR SGEP

In Section 5.3, we have shown that by exchanging the FFOP with adaptive predictors, an improved DPCM-AQJ performance is obtained, and that SGEP out-performs SAP for transmitted bit-rates ranging from 16 to 40 Kb/s.

This section examines the case of DPCM-AQF systems with second-order sequentially adaptive predictors. Once again, the encoding performance of DPCM-AQF was evaluated by computer simulations. The input speech signal was the same with that used in the DPCM-AQJ experiments, i.e., the sentence, *"An apple a day keeps the doctor away"*, bandlimited to 3.4 kHz and sampled at 10 kHz/sec.

In generating the step-size, Δ, α_q was set to 0.33 in accordance with the findings of reference[110] and the quantizer, QS, used for step-size transmission, had 256 levels while the ratio of its maximum to its minimum step-size was 128. The decision levels of the quantizer QE were $\pm1,\pm2,\ldots$, and output levels are defined by Equation (5.1), and they are scaled by Δ.

The predictors used in the DPCM-AQF experiments are those discussed in Section 5.2.2. For the SAP predictor, the A and B parameters of Equation (5.15) were 0.05 and 2.0 while for the second-order SGEP predictor, the values of A and B were 6.0 and 100.0, respectively. These values of A and B yield the maximum peak SNRSEG values measured over the

entire utterance. The constants s_1 and s_2 in the second-order SGEP algorithm, computed from Equation (4.26) with $\beta=1/5$, were 0.167 and 0.145 respectively. The value of α in Equation (5.23) was 0.8. Also, the initial value of a_1 was the first shift normalized correlation coefficient, of the speech signal, $c_1=0.94$, while the initial value of a_2 was set to zero.

The SNRSEG(dB) was used as an objective criterion to assess the performance of the various codecs, see Equation (3.4). The variation of SNRSEG(dB) as a function of input power for the channel error-free DPCM-AQF system using either FFOP or SAP or SGEP is shown in Figure 5.13 for 40 Kb/s transmission bit-rate (b=4), and for W=256, $\alpha_q=0.33$, and N=2. The peak SNRSEG(dB) of ADPCM-AQF-SGEP is seen to be 3 dB's greater than the peak SNRSEG(dB) values of ADPCM-AQF-SAP. We found that total SNR(dB) as defined in Equation (3.3), produced lower values than those of segmented SNR measurements, but the SNR(dB) gain of ADPCM-AQF-SGEP over the other systems was maintained, see Table 5.4.

DPCM-AQF CODEC	W=256		W=128		SNR(dB) OF DPCM-AQJ CODEC
	SNRSEG(dB)	SNR(dB)	SNRSEG(dB)	SNR(dB)	
FFOP	28.75	25.44	31.09	25.91	26.70
SAP	34.51	29.41	35.05	31.75	33.40
SGEP	37.83	34.2	38.54	34.94	36.10

TABLE 5.4: SNRSEG(dB)-SNR(dB) values of various codecs when input power level = -5 dB, b=4, N=2

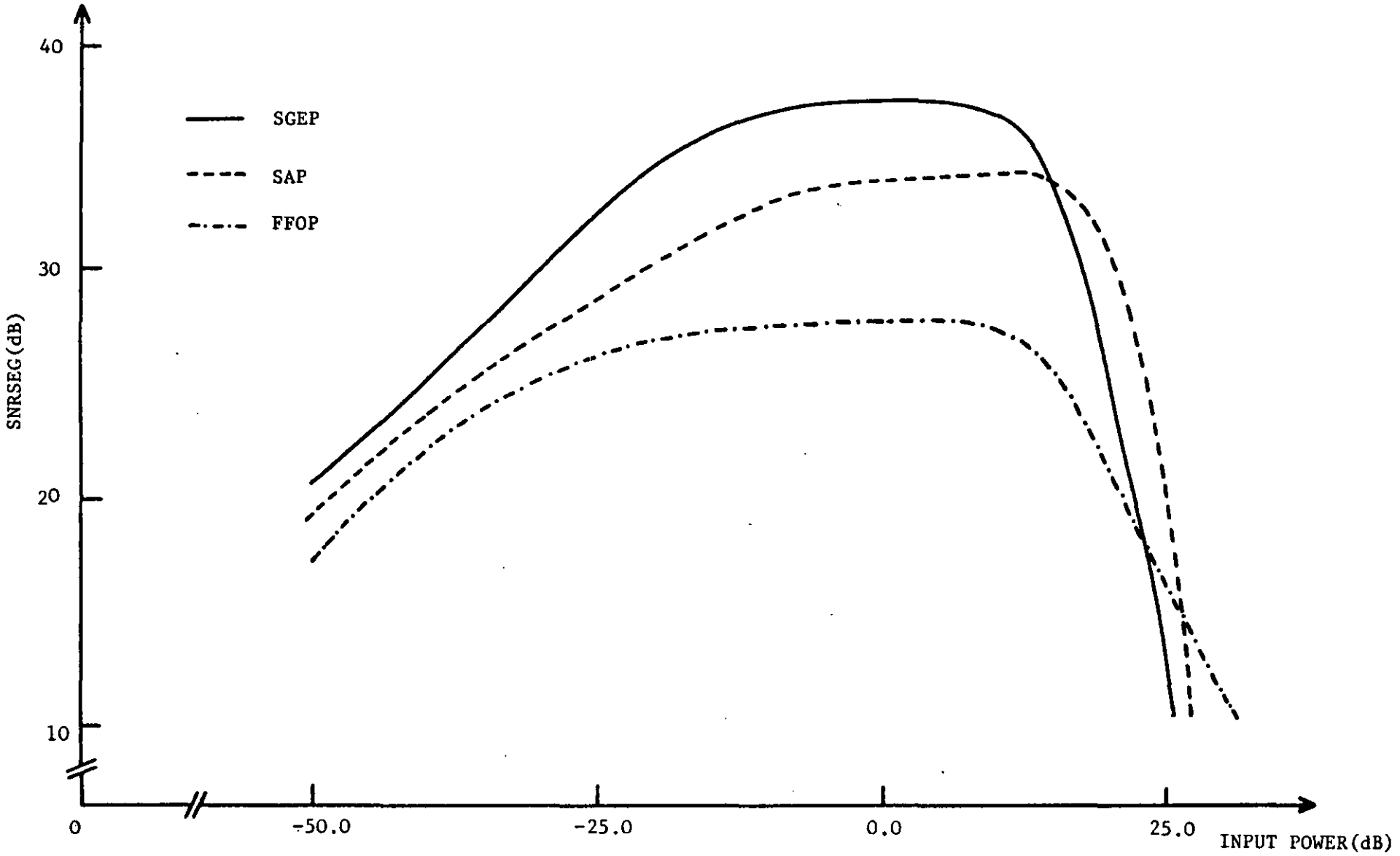


FIGURE 5.13: Variation of DPCM-AQF SNRSEG(dB) with input power:W=256, b=4, N=2, BER=0.0

The effect in varying the quantization block size, W, is shown in Figure 5.14 for an input level of -5 dB, i.e., SNRSEG(dB) values were selected at the centre of the codec's dynamic range. The SGEP predictor enables DPCM-AQF to maintain a nearly constant SNRSEG as W is varied from 32 to 512. It is observed that the loss in SNRSEG for larger values of W, is greater in DPCM-AQF-FFOP. This demonstrates the efficiency of the adaptive predictors, within the DPCM-AQF, encoder, in coping with the reduced quantization accuracy of QE. The variation of the SNR(dB) for every W=128 samples is shown in Figure 5.15, and the superiority of the ADPCM-AQF-SGEP over the ADPCM-AQF-SAP and DPCM-AQF-FFOP is once again observed.

The effects of transmission errors in DPCM-AQF codecs were also examined. Here the 8-bit codewords (256 levels for QS) used for the transmission of the step size Δ, were assumed to be protected from transmission errors while the bit stream at the output of the QE quantizer, see Figure 5.2, was subjected to random errors. These errors were induced according to the method described in Section 3.6.5, i.e., the number of samples deemed to be in error depends on the bit error rate (BER), total number of samples, and number of bits, see Equation (3.56). The performance of the DPCM-AQF system (W=256) using a fixed first order predictor (FFOP) is almost independent of transmission errors for BER<0.01% as can be seen in Figure 5.16, and is only limited by quantization noise. For BER>0.4%, the noise resulting from transmission errors swamps the quantization noise and the slope of the curve is 45^{o},

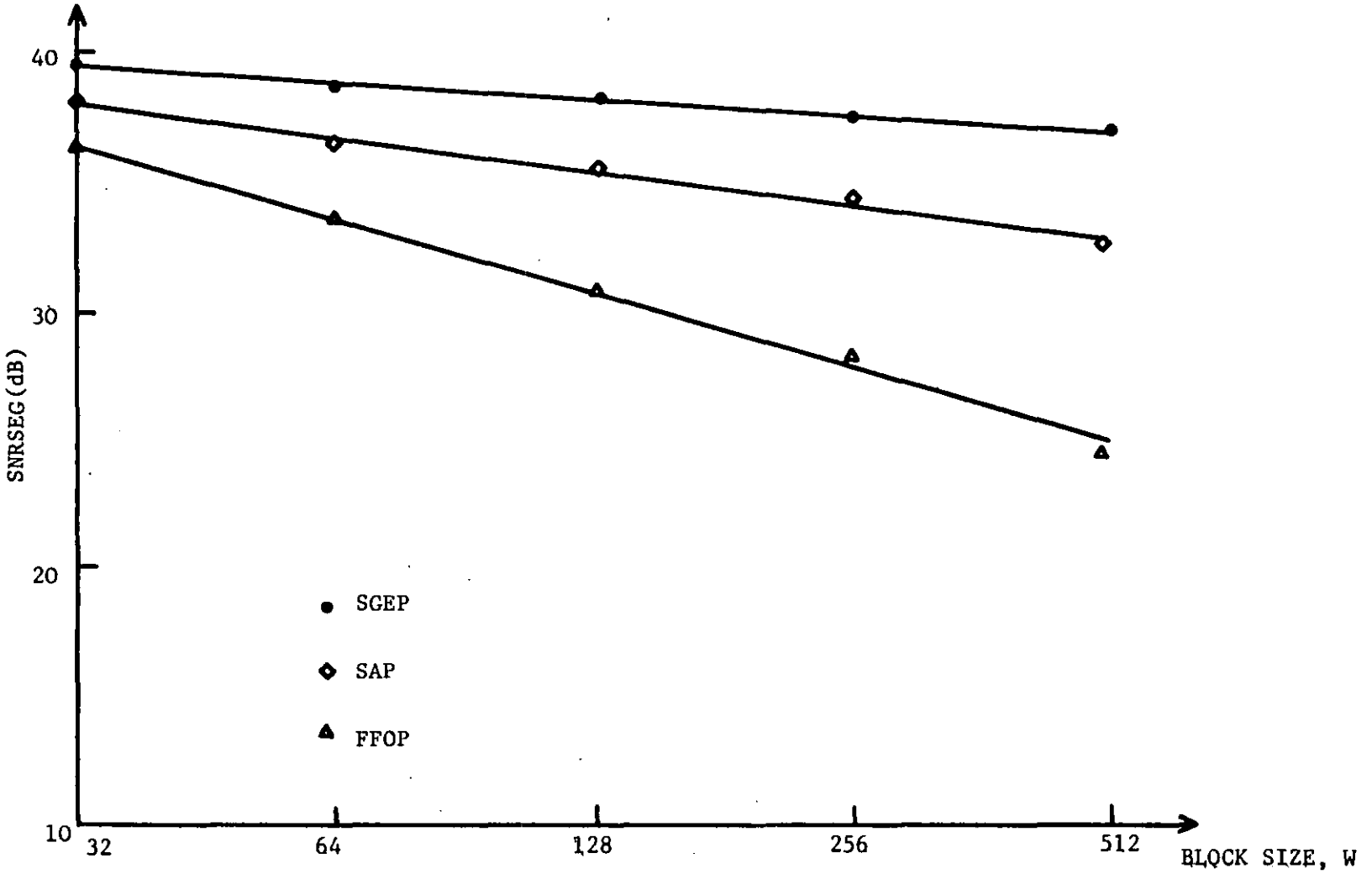


FIGURE 5.14: Variation of DPCM-AQF SNRSEG(dB) with block size, W:input power level = -5 dB, b=4, N=2, BER=0.0

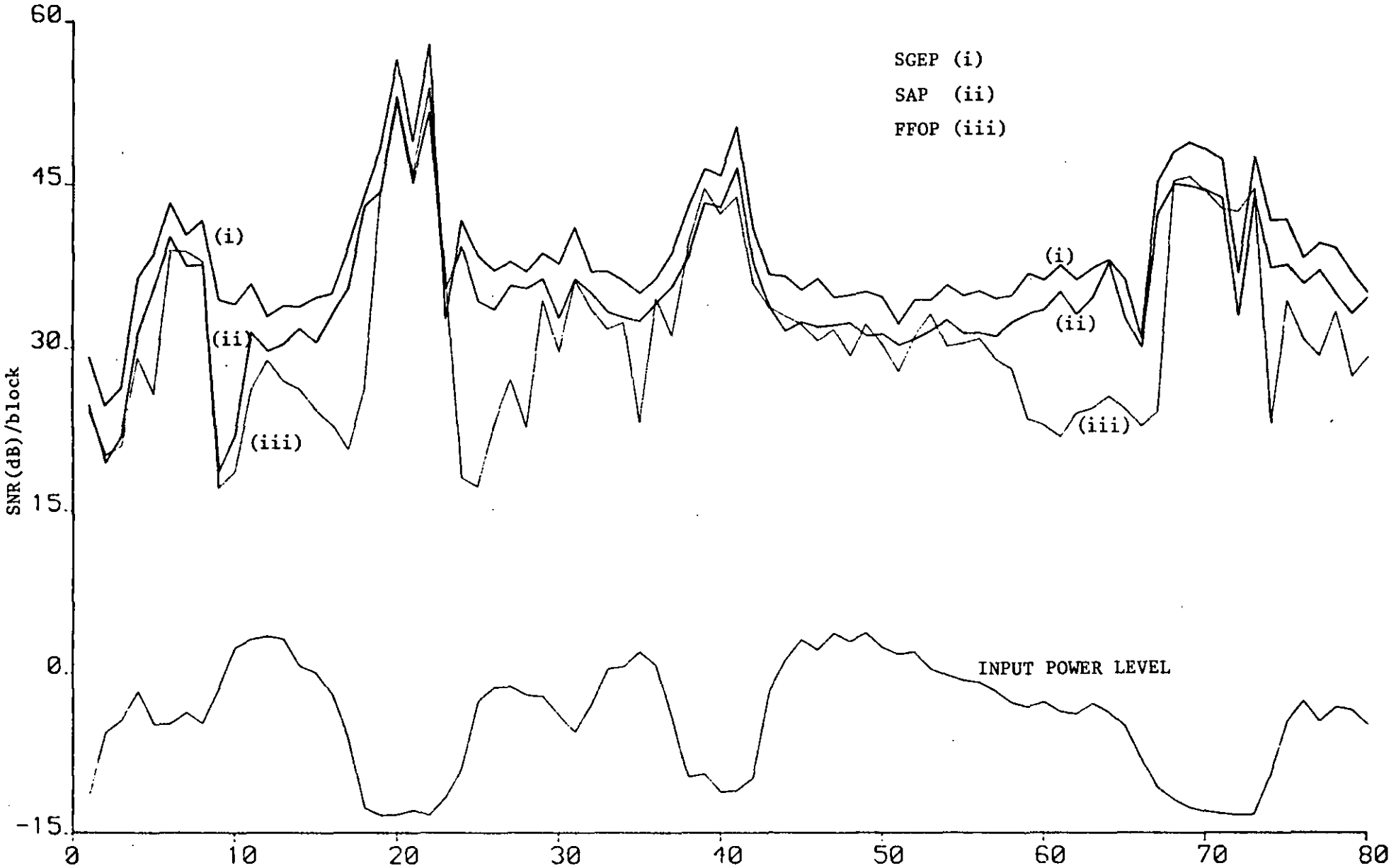


FIGURE 5.15: Variation of SNR(dB)/block, W=128, b=4, N=2, BER=0.0

i.e., the noise in the recovered speech is proportional to BER. With adaptive predictors, the average SNR starts to decrease for lower BER than encountered with the FFOP. This is because there is an error propagation effect in the second-order adaptive predictors, and although the performance of SGEP is better than SAP, the slope of the former is the greater for BER>0.4%. The use of only two predictor coefficients in the adaptive predictors and the channel protected Δ prevents the SNRSEG plumetting faster. The SNRSEG with SGEP and SAP is greater than that with FFOP for BER<0.08% and 0.03% respectively.

As Δ is represented by an 8-bit codeword and there are Wb (1024 bits for W=256, b=4), bits of DPCM data in each block, reliable protection of Δ can be provided for only a small expansion in the bit rate. For example, if Δ is transmitted three times with every block of W-DPCM words, then the step size $\tilde{\Delta}$ used for decoding at the receiver can be formed from the three received values of the step size. Specifically, the receiver would examine the most significant bit, MSB, of each of the three received Δ's, and then would select the bit that occurs on two or more occasions. It does this for the next MSB's and so on until the least significant bit, LSB's whence $\tilde{\Delta}$ is available. The probability of getting at least two bits correct in each bit position of $\tilde{\Delta}$ is[128]

$$h_e = [3(1-H_b)^2 H_b + (1-H_b)^3]^8 \qquad (5.44)$$

where H_b is the probability of any bit being received in error (statistical independence between bits being in error is assumed).

For a BER of 1%, H_b=0.01 and h_e=0.9976, see Equation (5.44). Thus

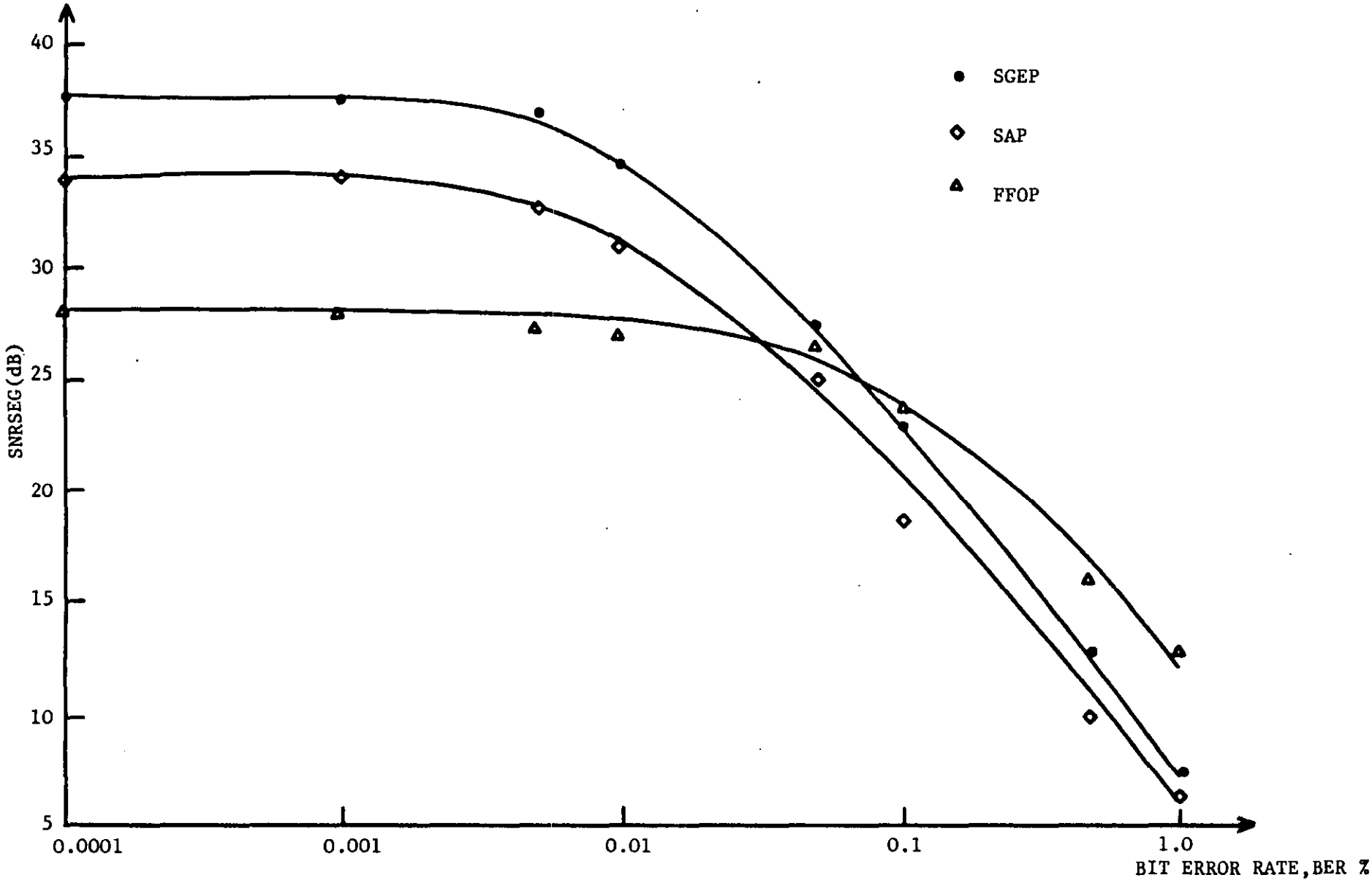


FIGURE 5.16: Variation of SNRSEG(dB) with BER %, input power = -5 dB, b=4, N=2, W=256

the probability of $\tilde{\Delta} \neq \Delta$ is $\tilde{h}_e = (1-h_e)$, and equal to 2.38×10^{-3}. If there are 256 4-bit DPCM words for every Δ transmitted, on the average one erroneous value of $\tilde{\Delta}$ is expected every $256 \times 4 / 2.38 \times 10^{-3} = 430$ Kb of transmitted DPCM data. Note that such a protection of Δ results in a 1.6% expansion of the bit rate. This simple example is included to demonstrate that protection of Δ is not a difficult problem, and of course more elaborate channel coding techniques are available to decrease the probability of Δ being received in error for only a modest increase in transmission bit rate.[129] Thus the curves in Figure 5.16 should be reviewed as representative rather than an upper bound for the particular speech signal encoded here.

Finally, informal listening tests of DPCM-AQF (W=256) with different predictors were carried out. For this purpose, the power level, at the input of the encoder was set to -5 dB, i.e., at the centre of the codec's dynamic range. At 40 Kb/s, DPCM-AQF employing SGEP, SAP or FFOP predictors yield an SNRSEG(dB) of 38.54, 35.05 and 31.09 dB's respectively. We noticed in our experiments that respective SNR gains between the various codecs could not be perceived. At the reduced transmission bit-rate of 30 Kb/s however, the subjective gain of the codec employing SGEP became significant over ADPCM-AQF-SAP. Also the quality of ADPCM-AQF-FFOP decoded speech was noticeably degraded.

5.5 DISCUSSION AND CONCLUSIONS

In this chapter DPCM-AQJ and DPCM-AQF speech codecs employing adaptive predictors have been treated. The bulk of the chapter was devoted to sequentially adaptive SGEP and SAP predictors, and the performance of DPCM codec employing these predictors, for transmission bit-rates of 16-40 Kb/s.

In Section 5.3, adaptive SGEP prediction algorithm in DPCM has been investigated in detail. It was observed that the faster convergence rate of SGEP over SAP, also manifests as a greater prediction gain. Consequently, this gain is reflected in a 2 to 3.5 dB's improvement in SNR as the number of bits per codeword is increased from 2 to 4.

We also demonstrated that the reduction in SNR of DPCM when SGEP is replaced by FFOP is 3.7 to 5.7 dB's as the transmission bit-rates increases from 16 to 40 Kb/s respectively, i.e., ignoring channel impairments, conventional DPCM-AQJ-FFOP codec is a poor performer compared with its adaptive predictor counterparts, although it does have the virtue of simplicity.

In Section 5.4, we have considered DPCM-AQF codec with adaptive predictors. This is because, DPCM-AQF codec has the advantage over DPCM-AQJ of being easier to implement, albeit a delay of W/f_s seconds, in calculating the step-size, Δ. Also, the complexity of the codec was further reduced by restricting the order of the adaptive predictors to two.

The experiments showed that by using the adaptive predictors, the

SNRSEG(dB) performance of the DPCM-AQF can be significantly improved over FFOP. For transmission bit-rate of 40 Kb/s, and a block size of W=256 speech samples, the ADPCM-AQF-SGEP codec has SNRSEG gains of 3 and 9 dB's compared to the ADPCM-AQF-SAP and DPCM-AQF-FFOP codecs respectively. However, it must be pointed out that, by arranging for the fixed predictor to have two coefficients, the SNRSEG(dB) increases by some 5.5 dB's, confirming Gibson's observation[64] that at bit rates >16 Kb/s the SAP predictor offers no advantage over fixed predictors having the same number of coefficients (FSOP).

Further, for an error-free channel, SNRSEG's of ADPCM-AQF-SGEP, ADPCM-AQF-SAP and DPCM-AQF-FFOP codecs decrease with increasing block length, W, due to reduced adaptation rate of the step size. The losses in SNRSEG(dB) for SGEP and SAP are however, 2.8 and 4.6 dB's respectively, as W is increased from 32 to 512. In the case of DPCM-AQF-FFOP, SNRSEG(dB) drops almost by 10 dB as W varies from 32 to 512. The DPCM-AQF codec employing SGEP therefore, has a weaker dependence on the block size, W, than those of employing SAP or FFOP predictors. When transmission errors are introduced, the codec employing SGEP has higher SNRSEG than that achieved with FFOP for BER<0.08%.

Finally, informal listening tests showed that, at 30 Kb/s, (step-size optimizing coefficient, α_q=0.50) SNRSEG(dB) gains of ADPCM-AQF-SGEP codec over ADPCM-AQF-SAP and DPCM-AQF-FFOP codecs could be perceived.

5.6 NOTE ON PUBLICATIONS

A paper entitled, *"DPCM-AQF Using Second Order Adaptive Predictors for Speech Signals"* [130] in co-authorship with Dr. C.S. Xydeas and Dr. R. Steele has been published in IEEE Transactions on Acoustic, Speech and Signal Proc., Vol. ASSP-29, No.3, pp.337-341, June, 1981. This paper is an abridged version of Sections 5.2, 5.4 and 5.5.

A paper entitled, *"Sequential Adaptive Predictors for ADPCM Speech Encoders"*, in co-authorship with the same authors has been accepted for publication in IEEE Transactions on Communications. This paper is a version of Sections 4.6, 4.8, 5.2 and 5.3.

CHAPTER VI

DPCM-AQF SPEECH CODECS WITH CORRELATION SWITCHED PREDICTORS

6.1 INTRODUCTION

During recent years, DPCM speech digitizers with Feedforward Adaptive Quantization (AQF) have received considerable attention.[60,128,130] Fixed or adaptive linear predictors have been used with DPCM-AQF and reported results clearly indicate the SNR advantage which can be obtained by using adaptive instead of fixed prediction. This SNR gain is of the order of 2 to 4 dB and may result in a considerable improvement in the quality and intelligibility of the decoded speech signal, particularly at low transmission bit rates.

Adaptive predictors can be Forward adaptive or Backward Sequential adaptive. Forward Block adaptive (FBA) predictors as discussed in Chapter III calculate the values of their prediction coefficients from the input speech samples every 10 to 20 msec. by minimizing the mean squared prediction error within this interval. The prediction coefficients are encoded and forward transmitted to the receiver in order for the decoder to operate with the same set of coefficients. Also, to ensure a stable decoding process, the prediction coefficients are usually transformed into reflection coefficients prior to transmission.

Backward Sequential Adaptive predictors update their coefficients at every sampling instant from previously decoded speech samples. In this case the prediction coefficients are determined at the receiver of DPCM-AQF codec using only the received samples from the output of the AQF quantizer. Since no other information is transmitted, the saving in bit rate and the possible synchronization difficulties which might be

encountered by combining the FBA prediction coefficients at regular intervals with the samples at the output of the quantizer encouraged the use of Backward Sequential Adaptive predictors. Their performance, when employed in the feedback loop of DPCM-AQF codec have been examined in Chapter V.

The price however to be paid is threefold. First, the look ahead procedure of the Forward adaptive prediction is not allowed in the sequential computation of the coefficients and therefore the estimation accuracy of the sequential predictors is reduced. Second, in the presence of transmission errors the performance of the predictor at the receiver can be seriously affected. Thus,for stability reasons, the sequential adaptation strategy is slightly modified and this can result in some degradation in the performance of the predictor. Third, the complexity of the receiver is considerably higher in the case of backward prediction since the decoder forms the prediction coefficients following the same procedure as the encoder. Gibson,[131] in a theoretical study of ADPCM systems indicated that no clear cut preference can be made between systems with Forward or Backward Sequential Adaptive prediction.

In this chapter, we study a new simplified (FBA) algorithm called the *"Correlation Switched Prediction"* (CSP). CSP divides the range of the first shift normalized correlation coefficient, c_1, of the speech signal into zones and as the value of c_1 changes when computed over successive blocks of W speech samples, the predictor coefficients undergo a substantial modification. In general, when the range of c_1 $(-1.0<c_1<1.0)$

is divided into (Z+1) zones, the predictor is referred to as (Z+1)-point CSP or Z-order CSP. The CSP technique has shown a considerable performance advantage when compared to fixed or sequential adaptive prediction, particularly when speech is transgressing from unvoiced to voiced sounds.

The first part of this chapter presents the CSP algorithm and a comparative study of DPCM-AQF systems employing one of the following prediction techniques:

a. Fixed First-Order Prediction, DPCM-AQF-FFOP,

b. Fixed Second-Order Prediction, DPCM-AQF-FSOP,

c. Sequentially Adaptive SGEP scheme, ADPCM-AQF-SGEP,

d. 4-point/3rd-order CSP associated with FSOP, ADPCM-AQF-CSP(4)-FSOP,

e. 4-point/3rd-order CSP associated with SGEP, ADPCM-AQF-CSP(4)-SGEP,

f. 8-point/7th-order CSP associated with FSOP, ADPCM-AQF-CSP(8)-FSOP,

g. 8-point/7th-order CSP associated with SGEP, ADPCM-AQF-CSP(8)-SGEP,

h. Forward Block Adaptive Prediction, FBADPCM-AQF.

In all these codecs, both SNRSEG(dB) and SNR(dB) are used as performance criteria, while the prediction order N is equal to two. Our studies were confined only to DPCM-AQF systems since (i) AQF quantization is more robust to transmission errors compared to the AQJ algorithm and also former scheme is considerably easier to implement, (ii) the discrepancies between the SNR values of DPCM-AQF and DPCM-AQJ systems using sequential adaptive predictors are quite small. In addition, only the SGEP prediction algorithm was employed in our DPCM-AQF system due to its superior performance over SAP.

Computer simulation results for the range of 16 to 32 Kb/s, with input speech signals, sampled at 8 kHz and bandlimited to 3.4 kHz indicated that:

i) SNRSEG(dB) gains of 1.7 and 3.2 were obtained for DPCM-AQF using a hybrid scheme formed by 4-point CSP and SGEP predictor, when compared to DPCM-AQF with a fixed second order predictor.

ii) SNRSEG(dB) performance of the codec when using FBA instead of Backward Sequentially Adaptive Prediction was improved. The introduction of the relatively simple CSP produced SNRSEG(dB) values comparable to those obtained from FBADPCM-AQF. A more complex prediction scheme which combines CSP and SGEP was shown to provide the best overall SNRSEG(dB) performance.

The computational complexity of the adaptive prediction algorithms was also considered in terms of multiplications and additions required to form the prediction coefficients within a fixed time interval.

The second part of the chapter examines the proposed DPCM-AQF codecs when used to digitize signals obtained from the Voiced/Unvoiced Band switching VUBS, system.[12] The VUBS preprocessor operates on 0.3 to 6.0 kHz wideband speech, and compresses the input signal into a 3.4 kHz bandwidth. The compressed signal can be sampled at 8 kHz and then digitized for transmission. At the receiver, following the decoding of the binary signal, the VUBS postprocessor is able to reproduce speech whose bandwidth is within the 0.3 to 6.0 kHz range. By employing DPCM-AQF together with the CSP prediction schemes, to digitize the output of the VUBS preprocessor, it was found that the reproduced speech signal is preferable to 0.3-3.4 kHz telephonic speech, digitized by the same codecs and for transmission bit rates of 16 to 32 Kb/sec.

6.2 CORRELATION SWITCHED PREDICTION SCHEME

Correlation switched predictors alter significantly the values of the prediction coefficients according to the first shift normalized correlation coefficient c_1 of the input speech samples where

$$c_1 = \frac{\sum_{i=1}^{W-1} x_i \, x_{i+1}}{\sum_{i=1}^{W} x_i^2} \quad . \tag{6.1}$$

The $\{x_i\}$ sequence is therefore divided into blocks of W samples and for each block the value of c_1 is computed. c_1 is then compared with a set of thresholds, TR_j, $j=1,2,\ldots,Z$ which divide the (-1.0,1,0) range of c_1 into (Z+1) zones. A specific zone is thus selected according to the value of c_1 and unique set of prediction coefficients $[a_1'', a_2'', \ldots, a_N'']$ assigned to that zone is then used in the linear predictor, thereby ensuring a high prediction gain for the W input samples being encoded.

In contrast to Forward Block Adaptive prediction, there is no need to transmit the prediction coefficients as these are stored in a look-up table at the receiver. All that is required is to transmit the value of the threshold j and this can be accomplished with a word having $\log_2(Z+1)$ bits. For typical values of Z this word consists of only 2 or 3 bits and therefore the increase in the bit rate is minimal. In this way the receiver accepts $\log_2(Z+1)$ bits every W sampling instants and selects the proper set of prediction coefficients to be used in the decoding procedure.

Obviously the question arises of how to define the sets of the prediction coefficients which form the entries of the look-up table. This procedure can be described as follows:

Initially the (-1.0,1.0) range is divided into $(2/\delta c)$ sub-zones using $[(2/\delta c)-1]$ thresholds TR_ℓ where δc is the sub-zone size and $(2/\delta c)>>Z+1$. c_1 is then measured for every block of W input speech samples and the $[a_1,a_2,\ldots,a_N]$ prediction coefficients are also determined using the autocorrelation method. After processing a large number of blocks, new coefficients $[a'_1,a'_2,\ldots,a'_N]$ are formed and they are assigned to each of the $(2/\delta c)$ sub-zones. In particular, the prediction coefficients $[a'_{1\ell},a'_{2\ell},\ldots,a'_{N\ell}]$ of the ℓth sub-zone are the average values of the $[a_1,a_2,\ldots,a_N]$ coefficients obtained from all the blocks of speech samples which satisfy the $TR_{\ell-1}<c_1<TR_\ell$ inequality, $TR_\ell=TR_{\ell-1}+\delta c$. Thus if the number of blocks satisfying the above inequality is M1, then,

$$a'_{k,\ell} = \frac{1}{M1}\sum_{r=1}^{M1} a_{k,r} \quad , \quad k=1,2,\ldots,N. \tag{6.2}$$

In this way a Master table consisting of $(2/\delta c)$ sets of prediction coefficients is formed and can be subsequently used to obtain the look-up table of an (Z+1)-point CSP algorithm.

Let us assume for the moment that the values of the Z thresholds TR_ℓ are known. Then the prediction coefficients $[a''_1,a''_2,\ldots,a''_N]$ assigned to each of the (Z+1) zones of the required (Z+1) point look-up table are obtained by averaging all the sets of coefficients of the Master table corresponding to those sub-zones contained within each of the much larger zones of the (Z+1)-point table. The prediction

coefficients corresponding to the pth zone of the (Z+1)-point table are therefore defined as

$$a''_{k,p} = \frac{1}{k_p} \sum_{r=1}^{k_p} a'_{k,(m+r)} \qquad (6.3)$$

where $k=1,2,\ldots,N$,

$m=k_1+k_2+\ldots+k_n+\ldots+k_{p-1}$,

$k_n=\delta n/\delta c$,

δ_n = the size of the nth zone of the Master table.

Equation (6.3) clearly shows the dependence of the $\{a''_{k,p}\}$ coefficients on the value of the thresholds. Furthermore for switched predictor to be efficient, the sets of the $\{a''_{k,r}\}$, $r=1,2,\ldots,Z+1$ and $k=1,2,\ldots,N$ coefficients should approximate those sets of the Master table formed with large values of M1, see Equation (6.2). The thresholds TR_j are therefore selected to minimize the effect of the averaging process of Equation (6.3) when applied to those coefficients $\{a'_{r,k}\}$, $r=1,2,\ldots,(2/\delta c)$ and $k=1,2,\ldots,N$ of the Master table which are formed with large values of M1. That is, in selecting TR_ℓ, an attempt is made to reduce the distortion in the values of the prediction coefficients corresponding to blocks of speech samples whose average statistics show a high probability of occurrance.

In our experiments, Table 6.1 is the Master table and gives the average prediction coefficients a'_1 and a'_2 determined using 157 blocks of W=256 input speech samples obtained at the rate of 8000 samples per second. The value of δc is 0.1 and the [-1.0+1.0] range is divided into $2/\delta c$=20 zones. The same table also provides the percentage of the blocks

whose c_1 value satisfied each of the 20 inequalities.

Having defined Table 6.1, according to the long term speech statistics, we can use it to obtain the required (Z+1)-zone look-up table of an (Z+1)-point or Z-order CSP scheme. As an example for a 4-point CSP algorithm, it was decided to have TR_1=0.7, TR_2=0.4 and TR_3=0.0, while the a_1'',a_2'' coefficients were determined by averaging the coefficients of the zones between 1.0 to 0.7, 0.7 to 0.4, 0.4 to 0.0 and 0.0 to -1.0, see Table 6.1. As previously stated, this allocation of prediction coefficients to each of the resulting 4 zones is clearly geared to minimize the distortion in the values of the coefficients for speech blocks whose average statistics show a high probability of occurrance. Table 6.2 and 6.3 present the thresholds and the prediction coefficients [a_1'',a_2''], of an 4-point and 8-point second-order CSP scheme, respectively.

In its present form the proposed CSP-FSOP algorithm behaves, within a block of W speech samples as a second order fixed predictor. CSP-FSOP is able however to modify every W samples, its prediction coefficients by selecting one out of (Z+1) sets of (a_1'',a_2'') prediction coefficients.

Switched prediction can be also combined with backward sequential prediction, for example the SGEP algorithm. In CSP-SGEP, the average prediction coefficients of each of the (Z+1) zones are used as the initial values of the SGEP prediction coefficients, for a block of W samples, thereby facilitating a faster coefficient convergence rate. However, if the value of c_1 does not change zones between the rth and

ZONES OF c_1	$-1.0<c_1\leq 1.0$	AVERAGE 2ND ORDER PREDICTOR COEF.		PROBABILITY OF OCCURRANCE OF c_1 IN 157 TRIALS	4-POINT CSP	8-POINT CSP
		a_1'	a_2'			
1	$0.9<c_1\leq 1.0$	1.6505	-0.759	26%	1	1
2	$0.8<c_1\leq 0.9$	1.424	-0.677	28%		2
3	$0.7<c_1\leq 0.8$	1.060	-0.410	12%	2	3
4	$0.6<c_1\leq 0.7$	0.840	-0.280	5%		4
5	$0.5<c_1\leq 0.6$	0.653	-0.141	5.7%		
6	$0.4<c_1\leq 0.5$	0.589	-0.337	6.3%	3	5
7	$0.3<c_1\leq 0.4$	0.420	-0.194	5.7%		6
8	$0.2<c_1\leq 0.3$	0.310	-0.170	2.5%		
9	$0.1<c_1\leq 0.2$	0	0	0		
10	$0.0<c_1\leq 0.1$	0	0	0		
11	$-0.1<c_1\leq 0.0$	-0.041	+0.256	1.2%	4	7
12	$-0.2<c_1\leq -0.1$	-0.170	-0.336	1.2%		
13	$-0.3<c_1\leq -0.2$	-0.221	0.115	2.5%		8
14	$-0.4<c_1\leq -0.3$	0	0	0		
15	$-0.5<c_1\leq -0.4$	-0.694	-0.547	1.9%		
16	$-0.6<c_1\leq -0.5$	-1.084	-0.873	1.2%		
17	$-0.7<c_1\leq -0.6$	-0.976	-0.617	0.6%		
18	$-0.8<c_1\leq -0.7$	-1.232	-0.541	0.6%		
19	$-0.9<c_1\leq -0.8$	0	0	0		
20	$-1.0<c_1\leq -0.9$	0	0	0		

TABLE 6.1: The Master Table and the Organisation of the 4-point and 8-point look-up tables

j	THRESHOLD TR_j	CORRELATION ZONE	COEFFICIENT a''_1	COEFFICIENT a''_2
1	0.7	0.7 to 1.0	1.524	-0.718
2	0.4	0.4 to 0.7	0.950	-0.345
3	0.0	0.0 to 0.4	0.493	-0.210
		-1.0 to 0.0	-0.631	-0.363

TABLE 6.2: Look-Up Table for 4-Point CSP or 3rd Order CSP

j	THRESHOLD TR_j	CORRELATION ZONE	COEFFICIENT a'_1	COEFFICIENT a''_2
1	0.9	0.9 to 1.0	1.605	-0.760
2	0.8	0.8 to 0.9	1.424	-0.680
3	0.7	0.7 to 0.8	1.060	-0.410
4	0.4	0.4 to 0.7	0.750	-0.210
5	0.3	0.3 to 0.4	0.590	-0.334
6	0.0	0.0 to 0.4	0.365	-0.182
7	-0.4	-0.4 to 0.0	-0.144	0.012
		-1.0 to -0.4	-0.996	-0.645

TABLE 6.3: Look-Up Table for 7th-Order or 8-Point CSP

(r+1)th blocks of input samples, the initial values of the coefficients $[a_1'', a_2'', \ldots, a_N'']$ for the (r+1)th block are the last values of $[a_1'', a_2'', \ldots, a_N'']$ defined by the SGEP algorithm after the processing of the rth block of samples. That is, the prediction coefficients stored in the (Z+1)-point look-up table (see Table 6.2 or 6.3), are only introduced as the initial values of the SGEP algorithm when a zone change occurs. An ADPCM-AQF employing a (Z+1)-zones CSP-SGEP will be referred to as ADPCM-AQF-CSP(Z+1)-SGEP.

It should be emphasised that in the SGEP algorithm we slightly modify the predictor convergence term, P_i, previously defined by Equation (5.14) to

$$P = \frac{A}{(\Delta/\alpha_q)^2+B} \tag{6.4}$$

where the step size Δ and the scaling factor α_q are specified in Section 5.2 of the previous chapter. Consequently, P remains constant for W samples and because of its dependency on the channel protected Δ the codec's performance in the presence of channel errors is enhanced. It is important to note that Equation (6.4) also reduces the number of multiplications and additions (see Table 4.2), by N, i.e. now the computation of N prediction coefficients using SGEP algorithm requires (4N+4) multiplications and 5N additions per sampling instant.

A general block diagram of an encoder employing Z-order or (Z+1)-point CSP schemes is presented in Figure 6.1.

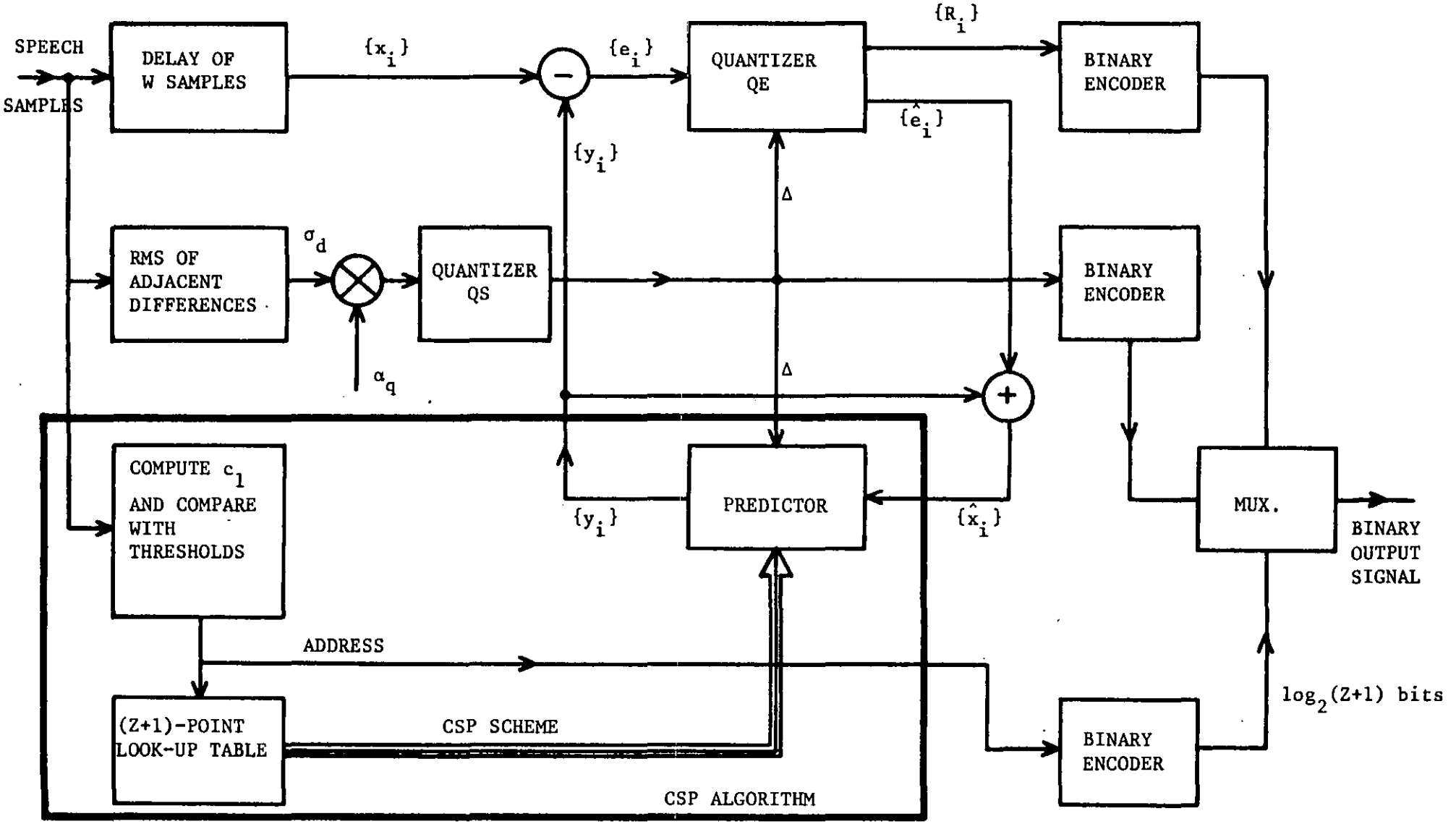


FIGURE 6.1: General Block Diagram of DPCM-AQF Encoder Employing Z-point CSP Scheme

6.3 THE VOICED/UNVOICED BANDSWITCHING SYSTEM, VUBS

The VUBS system developed by Patrick et al,[12] has been used in conjunction with DPCM-AQF-CSP(4)-FSOP and ADPCM-AQF-CSP(4)-SGEP to produce a wideband (0.3-6.0 kHz) speech digitizer operating at bit rates equal to or less than 32 Kb/s. Consequently, prior to the computer simulation results of the VUBS plus the DPCM-AQF system, it will be appropriate to briefly discuss the VBUS pre/post processor and the motives leading to its design.

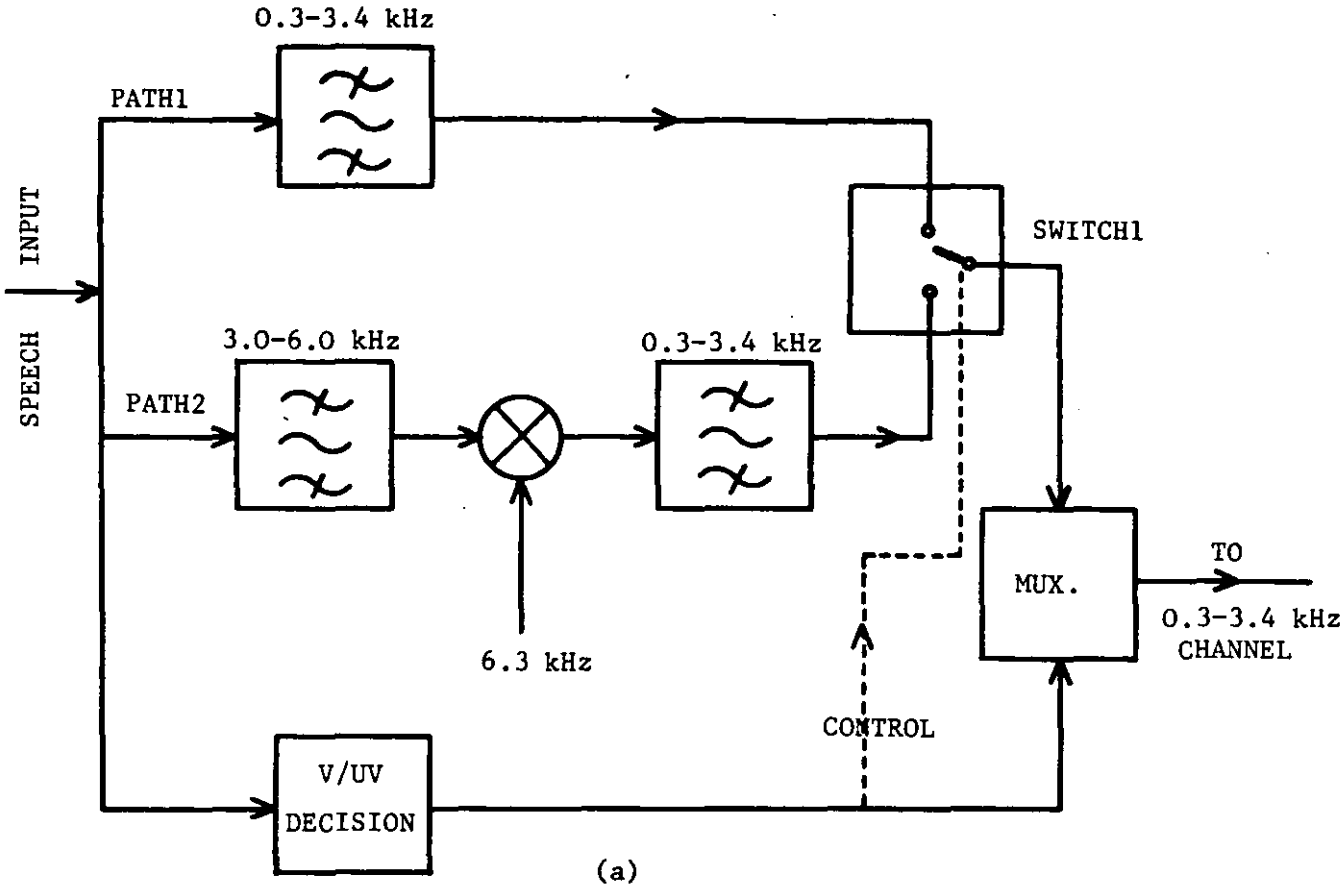
The VUBS system offers a conceptually simple method for the transmission of relatively wideband speech, 0.3-6.0 kHz, over the telephonic bandwidth 0.3 to 3.4 kHz and appears to show an improvement in intelligibility and quality of the reconstructed speech. In telephone bandwidth channels, certain unvoiced sounds such as /s/ or /f/ are usually perceived incorrectly because a large amount of their energy is concentrated above the upper cut-off frequency of the normal telephone channel. Therefore, by transmitting the frequency components of unvoiced speech which are perceptually most significant and still occupying a 3 kHz bandwidth, speech close in quality to the original 6 kHz speech can be perceived.

Figure 6.2(a) presents the block diagram of VUBS preprocessor where the 6.0 kHz input speech signal follows two paths: The first path, PATH1, limits the bandwidth of speech to within the 0.3 to 3.4 kHz frequency range while in the second path, PATH2, only the 3.0 kHz to 6.0 kHz frequency range is selected and subsequently shifted down to

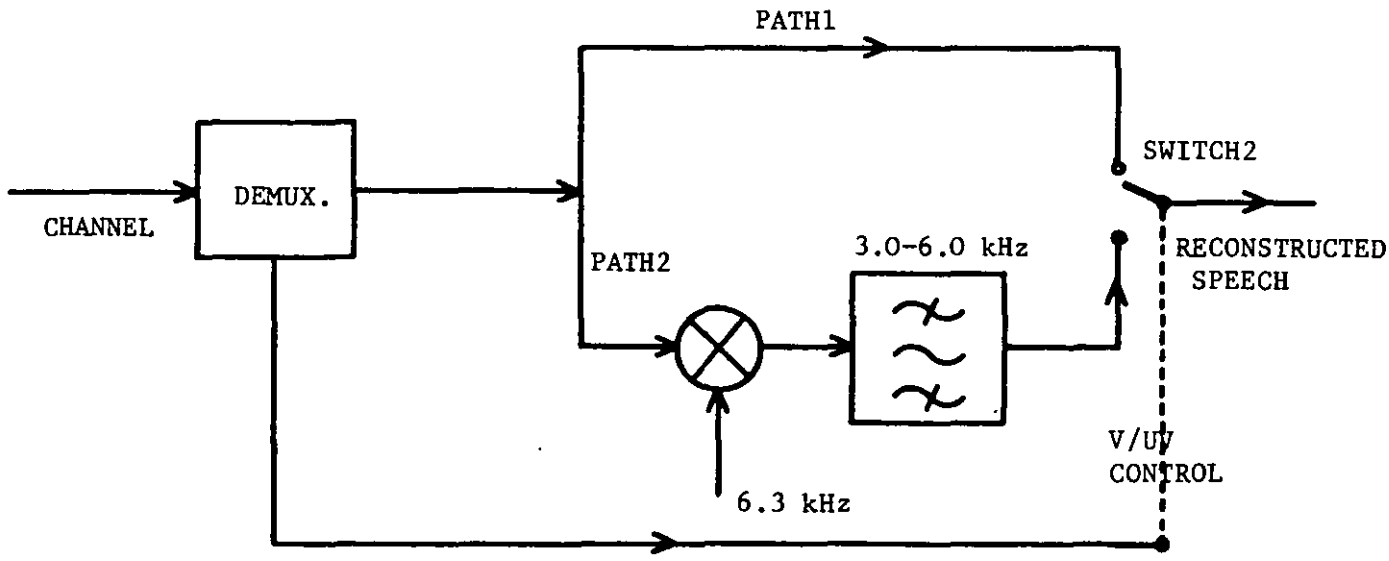
the 0.3 to 3.4 kHz band. The decision concerning whether PATH1 or PATH2 is to be transmitted is made by a voiced/unvoiced switch.[132] This leads to the transmission of the signal in PATH1 of voiced speech is present while the signal formulated in PATH2 is transmitted when the input speech is unvoiced. The V/UV decision is also transmitted to the receiver as a side information.

Figure 6.2(b) shows the block diagram of VUBS postprocessor. When the V/VU switch indicates voiced speech to the receiver, the received signal is directed to the output via SWITCH2, if however, unvoiced speech is deemed to be present by the V/UV switch, then the received signal is shifted up in frequency from 0.3-3.4 kHz bandwidth to 3.0-6.0 kHz range before being sent to the output.

It should be stressed that VUBS system recreates a signal that occupies the 6 kHz band, although never all of it at any instant. Informal listening experiences on a small sample of input speech material seemed to confirm that the VUBS speech is preferable to telephone band limited speech.



(a)



(b)

FIGURE 6.2: The VUBS System
a. Transmitter, b. Receiver

6.4 COMPUTER SIMULATION RESULTS AND DISCUSSION OF DPCM-AQF SPEECH CODECS USING SWITCHED PREDICTORS

The study of the proposed correlation switched prediction when applied to DPCM-AQF and its comparison with the DPCM-AQF-FSOP and SGEP systems was performed by computer simulations. The evaluation of the DPCM-AQF codec performance was ascertained by a) waveform plots and b) SNRSEG(dB) and SNR(dB) measurements selected at the centre of the codec's dynamic range.

Speech data, band limited to 0.3-3.4 kHz and sampled at 8 kHz was used as the input signal to the codecs. This signal referred to as the Band Limited Signal, BLS, was then encoded at 2,3 and 4 bits/sample. The following test words formed the input speech signal: *"sister, father S.K. Harvey, shift, thick, fist, talk, spent and vote"*. These utterances have numerous unvoiced/voiced transitions and this, coupled with the lower sampling rate, 8 kHz rather than 10 kHz as in previous chapters, exacerbates the difficulties of the predictors. We deliberately introduced these difficulties to determine the effectiveness of switched prediction during the voiced/unvoiced transitions of the input speech, BLS.

The DPCM-AQF parameters used in our simulation were α_q=0.33, 0.50 and 1.0, for b=4,3,2 bits, respectively, while the block size, W was set to 256. The SGEP parameters were A=5.0, B=100.0, D=10.0, β=1/5, α/β=4 and initial SGEP coefficients were a_1=0.86, a_2=0.00. For the FSOP the coefficients were fixed to a_1=1.580 and a_2=-0.778.

The waveform plot for the word *"sister"* is displayed in Figure 6.3, together with the first correlation coefficients, c_1, for each block of

256 samples. Curves (a) and (b) in Figures 6.4, 6.5 and 6.6 show the variation of block SNR for a DPCM-AQF-FSOP and a ADPCM-AQF-SGEP codec as a function of block number for the word *"sister"* and b=2,3,4 bits, respectively.

The FSOP was designed using the long-term statistics of speech signals, and because voiced speech is much more prevalent than unvoiced speech, the DPCM-AQF-FSOP has a relatively poor performance for unvoiced speech. This is evident from curve (a) in Figures 6.4 to 6.6. For voiced sounds, the DPCM-AQF-FSOP is inferior to ADPCM-AQF-SGEP because the latter has a greater prediction accuracy, being able to cope more successfully with the local variations in the speech statistics. This can be seen by comparing curves (a) and (b) in Figures 6.4, 6.5 and 6.6. However, a careful inspection reveals that the DPCM-AQF-FSOP out-performs ADPCM-AQF-SGEP in blocks number 3,10 and 11. This is due to the speech experiencing a transition from unvoiced to voiced speech (on-set of /t/) demanding that SGEP converges rapidly to new coefficient values. The convergence rate required is so fast that it would produce instability when encoding other segments of speech. The FSOP, on the other hand handles the situation better as the signal is changing to a form it is more capable of handling, namely voiced speech.

To improve the performance of the DPCM-AQF-FSOP system,we devised a switched-fixed predictor, SFP, with different sets of fixed prediction coefficients used for voiced (a_1=1.580, a_2=-0.773) and unvoiced (a_1=0.494, a_2=-0.3080) sounds. This scheme has resulted in a prediction gain during the encoding of unvoiced speech. It should be noted however, that a

voiced/unvoiced decision is required to determine which set of prediction coefficients is to be employed for the encoding of a particular speech segment. Also voiced/unvoiced information has to be multiplexed with the digitized speech and transmitted so that the decoder operates with the correct prediction design.

In order to further improve this switched-fixed prediction, SFP scheme, specially during voiced speech, a SGEP-S-FSOP predictor was developed whereby the SGEP algorithm was used to process voiced speech and FSOP was employed for the prediction of unvoiced sounds. In this way, the short-term statistical variation of voiced speech was treated by the sequentially adaptive algorithm ensuring a larger prediction gain compared to the SFP scheme. The values of the FSOP coefficients were determined from the long-term autocorrelation function of unvoiced speech (a_1=0.4944, a_2=-0.3080). The initial values of the SGEP prediction coefficient used to encode a voiced speech segment were the last values, assumed by SGEP when processing the previous voiced speech segment.

Computer simulation results of the DPCM-AQF-FSP and ADPCM-AQF-SGEP-S-FSOP systems showed an advantage over DPCM-AQF using SGEP or FSOP predictors, but the new schemes required a voiced/unvoiced switch whose output is multiplexed at irregular intervals, with the digitized speech.

It was therefore decided to develop an alternative switched prediction scheme, namely the Correlation Switched Predictor, CSP of Section 6.2, which could operate on a regular block basis and could also offer more than two sets of fixed coefficients to be used for the encoding

of input speech segments. The variations of block SNR(dB) as a function of block number for a ADPCM-AQF-CSP(4)-FSOP codec is shown in Figures 6.4 to 6.6, curve (c), for 2,3 and 4 bits respectively. Generally the ADPCM-AQF-CSP(4)-FSOP system has a superior performance compared to the ADPCM-AQF-SGEP and particularly at 16 Kb/s, see curves (b)-(c) in Figure 6.4.

When extending the idea of switched prediction to SGEP, the resulting CSP(4)-SGEP algorithm again used the entries of Table 6.2 but now as the initial values of the prediction coefficients. It is also recalled that if "c_1" do not change correlation zone between the r^{th} and $(r+1)^{th}$ blocks, the initial values of the coefficients a_1 and a_2 for the $(r+1)^{th}$ block are the last values of a_1 and a_2 formed in the r^{th} block. ADPCM-AQF-CSP(4)-SGEP gave the best overall performance when compared to previous systems, as shown by Figures 6.4 to 6.6 and b=2,3, and 4 bits, respectively.

The 4-point CSP schemes were then replaced by 8-point CSP prediction algorithm. In this case, Z=7, and the number of bits to be transmitted to the receiver is increased to $\log_2(Z+1)=3$ bits. The corresponding set of prediction coefficients for each correlation zone is given in Table 6.3. It was found however that contrary to the increase in the amount of information provided to the predictor, the gains in SNRSEG(dB) or SNR(dB) values for ADPCM-AQF-CSP(8)-FSOP and ADPCM-AQF-CSP(8)-SGEP compared to DPCM-AQF with CSP(4)-FSOP or CSP(4)-SGEP predictors were negligible. This can be attributed to the mis-judgement in grouping the coefficients of the 8-point CSP scheme.

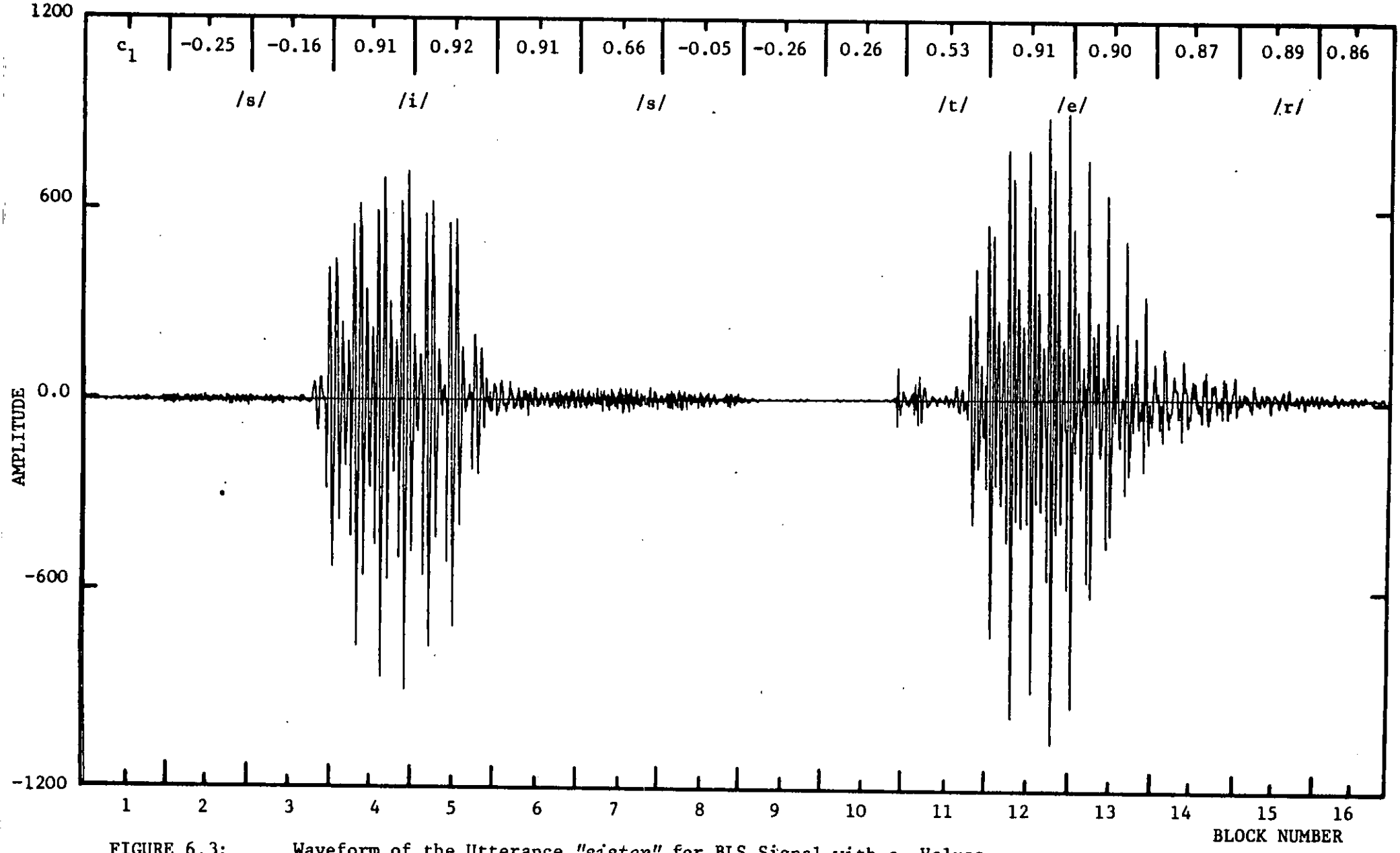


FIGURE 6.3: Waveform of the Utterance *"sister"* for BLS Signal with c_1 Values

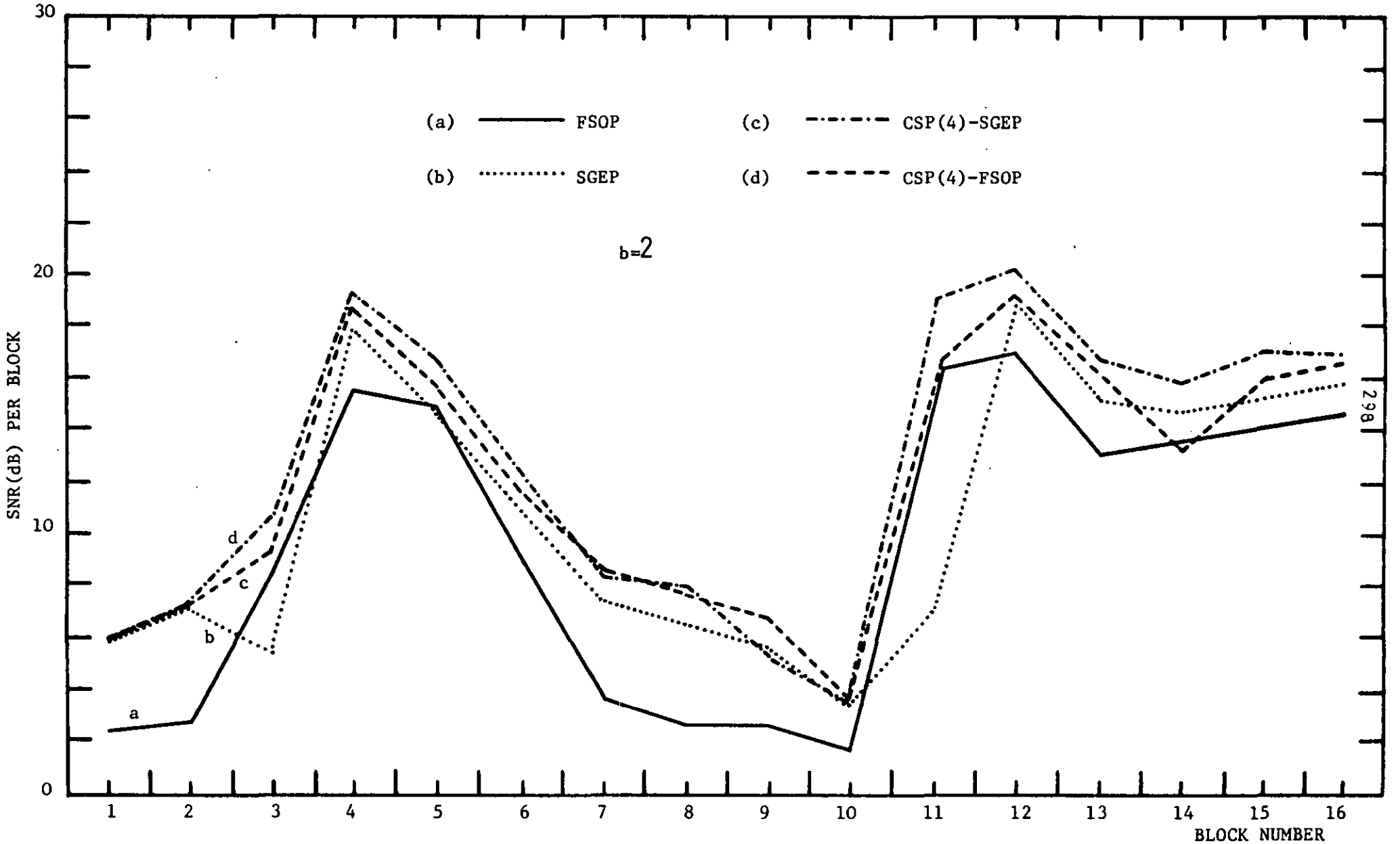


FIGURE 6.4: Variation of SNR(dB) per Block for Utterance sister when Encoded by 2 bits/sample DPCM-AQF Encoder

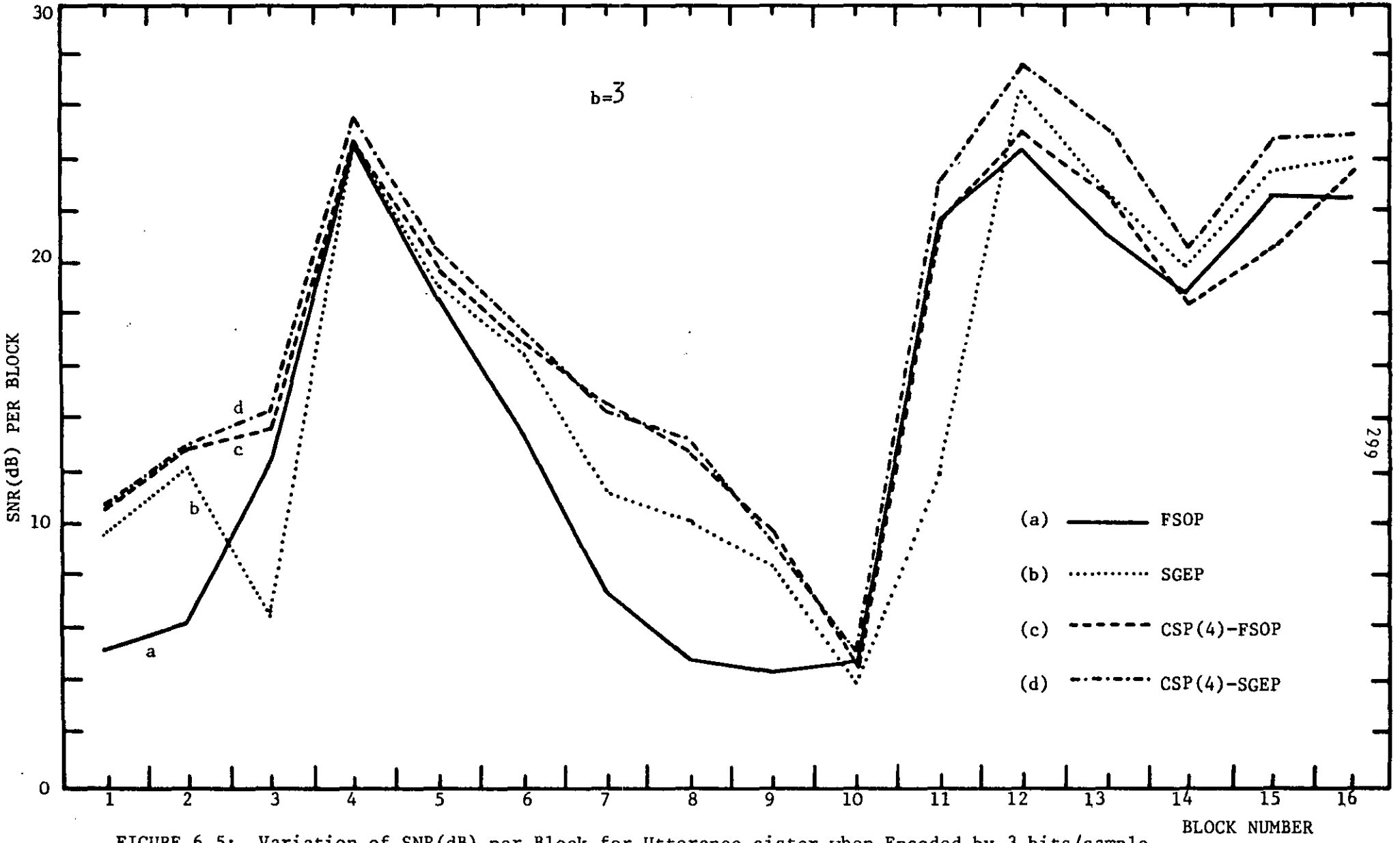


FIGURE 6.5: Variation of SNR(dB) per Block for Utterance sister when Encoded by 3 bits/sample DPCM-AQF Encoder

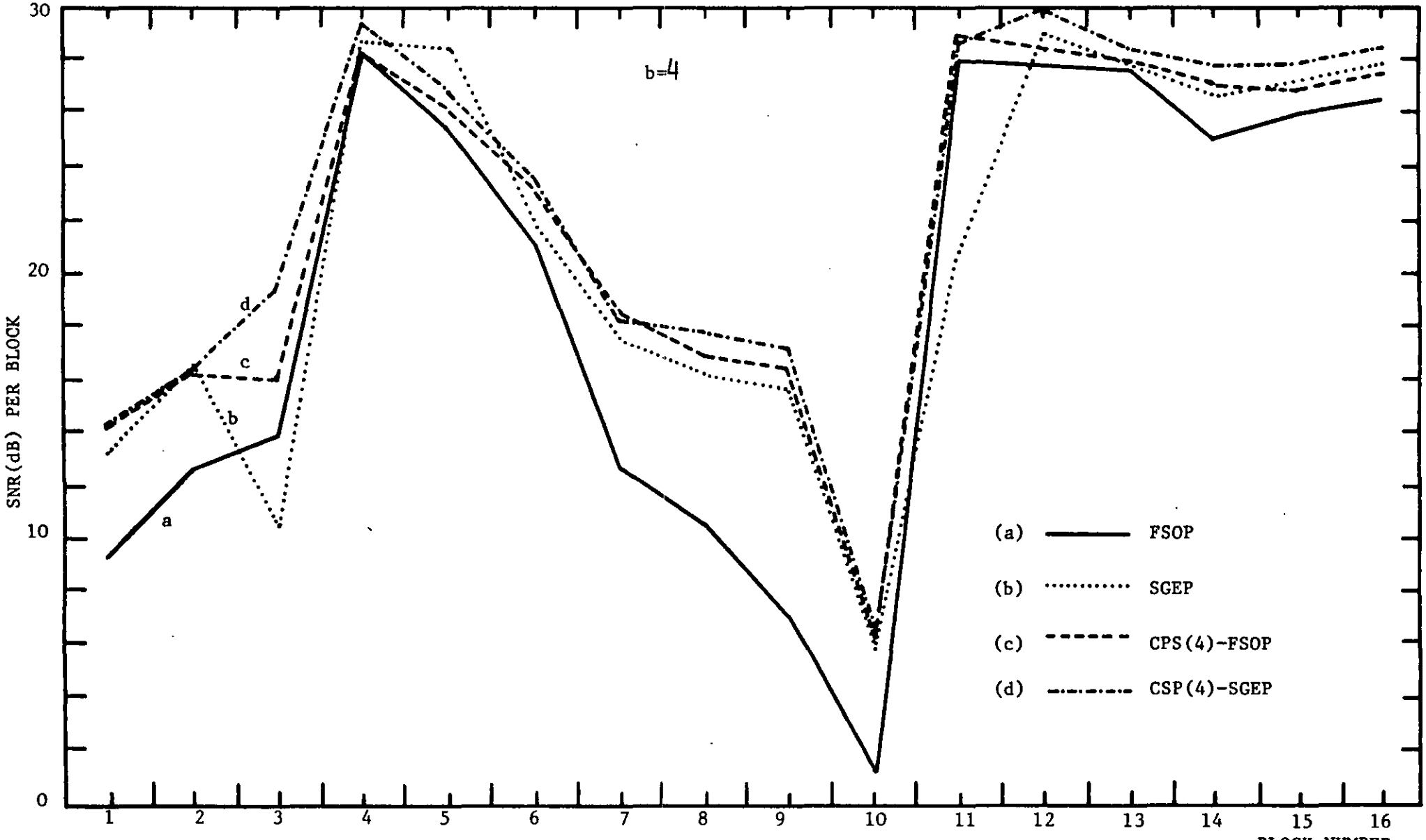


FIGURE 6.6: Variation of SNR(dB) per Block for Utterance sister when Encoded by 4 bits/sample DPCM-AQF Encoder

Furthermore, for comparison purposes we have also conducted DPCM-AQF experiments with Forward Block Adaptive prediction schemes and their prediction coefficients, for every block of 256 speech samples, were defined using Durbin's sequential algorithm, see Appendix G.

The SNRSEG(dB) and total SNR(dB) values for all the codecs discussed in this chapter are displayed in Table 6.4. These results are the average values obtained from two input utterances, namely: (sister, father, S.K. Harvey) and (shift, thick and fist). Figure 6.7 (a), (b) and (c) presents the SNRSEG(dB) values of eight DPCM-AQF coders when $\{e_i\}$ is quantized to an accuracy of 2,3 and 4 bits per sample. In general, DPCM-AQF-FSOP provides an average gain of 2 dB over the coder using fixed first-order prediction (FFOP). When adaptive prediction is used in the form of SGEP, this SNRSEG gain increases to 3 dB. A further gain of about 1 dB is obtained by using 4-point correlation switched prediction, CSP(4),while the hybrid CPS(4)-SGEP predictor provides an additional 1 dB improvement. Thus DPCM-AQF with CSP(4)-SGEP offers SNRSEG(dB) values of about 2 dB more than those of DPCM-AQF-SGEP. Finally DPCM-AQF with forward block adaptive prediction (FBA) provided only a marginal SNRSEG(dB) improvement over the DPCM-AQF-CSP(4)-SGEP system.

In Figure 6.7, although the number of bits per sample used to quantize the error sequence $\{e_i\}$ is kept constant for all codecs, the actual transmission bit rates vary slightly from one system to another. This is because different encoders require different amounts of extra

THE TYPE OF PREDICTOR IN DPCM-AQF CODEC	BITS PER SAMPLE					
	2		3		4	
	SNRSEG(dB)	SNR(dB)	SNRSEG(dB)	SNR(dB)	SNRSEG(dB)	SNR(dB)
FFOP	9.35	10.00	12.56	13.43	18.60	20.93
FSOP	11.30	15.89	15.29	20.41	19.97	25.14
SGEP	12.18	15.96	16.70	21.08	21.36	25.78
CSP(4)-FSOP	12.79	15.60	17.28	20.80	22.10	26.12
CSP(4)-SGEP	12.98	16.52	17.96	21.41	23.16	26.75
CSP(8)-FSOP	12.80	16.33	17.46	21.24	22.19	25.94
CSP(8)-SGEP	12.96	16.54	17.71	21.30	22.85	26.80
FBA	13.03	16.61	17.90	21.39	23.38	26.37

TABLE 6.4: SNRSEG(dB) and SNR(dB) for DPCM-AQF with 8 different prediction schemes

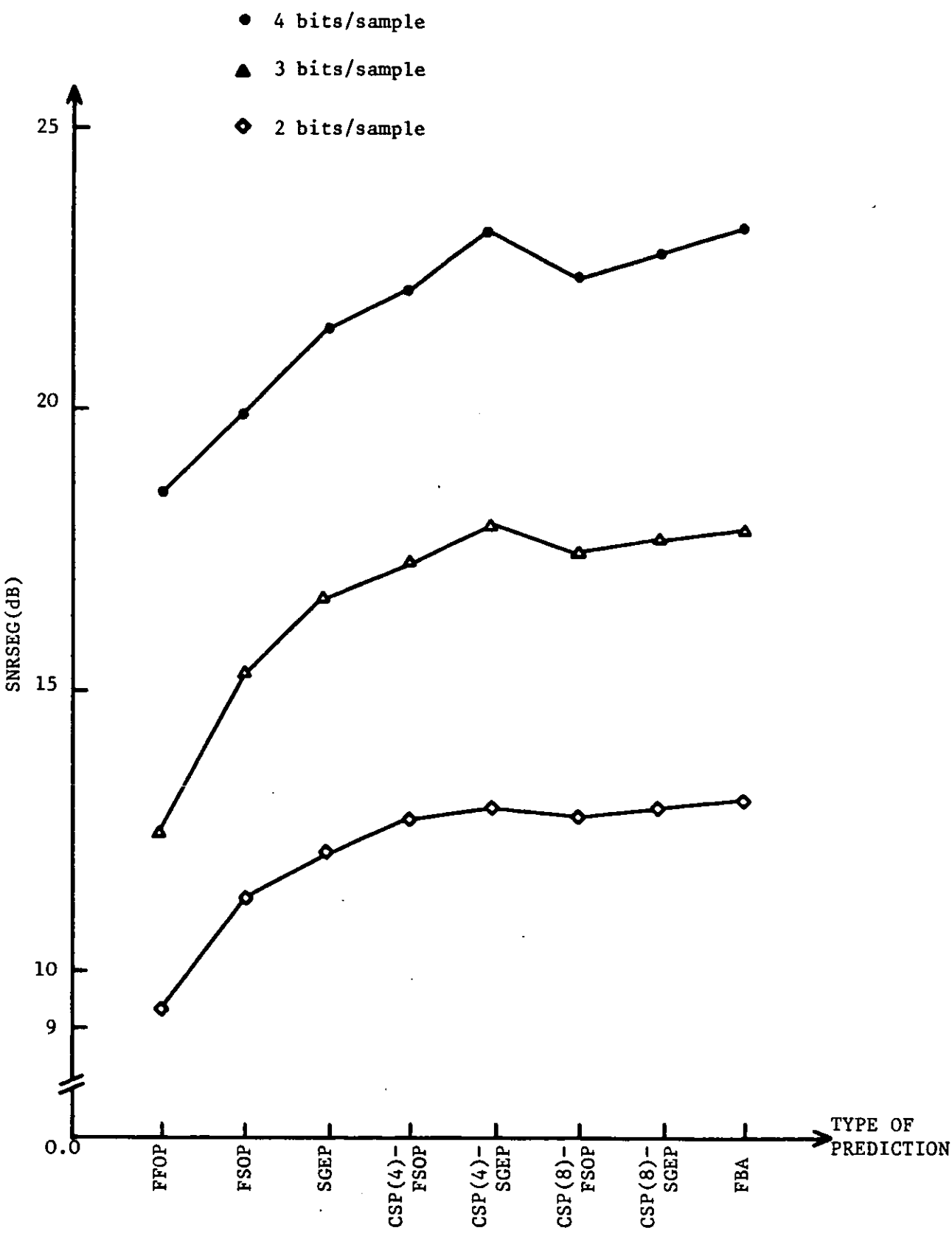


FIGURE 6.7: Variation of SNRSEG(dB) for Various DPCM-AQF Speech Codecs with Quantization Accuracy of 2,3 and 4 bits/sample

information to be multiplexed with the information at the output of the DPCM quantizer. In particular, DPCM-AQF coders with CSP or FBA predictors require an extra 63 or 500 bits respectively, when compared to coders with fixed or backward adaptive prediction. An 8-bits quantization for the FBA coefficients is assumed. This bit rate increase however, can be translated into an equivalent signal-to-noise ratio loss which can then be subtracted from the values shown in Figure 6.7. For example, using Noll's expression[48]

$$\text{LOSS IN SNR} = 6.02 \ \frac{b_c}{W} \ \text{dB} \tag{6.5}$$

where b_c bits is the number of bits to be allocated per coefficient, the LOSS IN SNR for the system with the larger increase in the bit rate i.e., DPCM-AQF-FBA, is found to be 0.376 dB. Thus with such minimal SNR losses, the relative SNRSEG(dB) performance of the DPCM-AQF encoders will hardly change from that indicated in Figure 6.7. Perhaps the only visible SNR reduction occurs in the case of DPCM-AQF-FBA and this is obviously to the advantage of DPCM-AQF-CSP(4)-SGEP which now shows a marginal gain over the coder with FBA prediction.

Next, we consider the computational requirements of the adaptive predictors used in our experiments. Table 6.5 shows the number of multiplications required by the FBA, CSP(4), CSP(4)-SGEP and SGEP algorithms in order to determine the N prediction coefficients for a block of W samples. Notice that CSP(4) is the algorithm with the smaller number of multiplications and that SGEP demands within W samples, a larger number of operations than the FBA predictor. On the other hand, if we are to consider the storage required by these adaptive

algorithms, then SGEP shows a considerable advantage over the block adaptive FBA, CSP(4) or the hybrid CSP(4)-SGEP scheme. In our particular case however, where the predictor is used in DPCM-AQF, a storage of W samples is already provided by the AQF quantizer. Thus there is little difference in storage requirements between the coders in Figure 6.7.

TYPE OF PREDICTION	NUMBER OF MULTIPLICATIONS PER BLOCK, W	NUMBER OF ADDITIONS PER BLOCK, W
FBA	$[N^2+(7+2W)N+2W]/2$	$[N^2+(2W-3)N+2W-2]/2$
SGEP	$(4N+4)W$	$5NW$
CSP(4)	$2W+1$	$2W-3$
CSP(4)-SGEP	$(4N+6)W+1$	$(5N+2)W-3$

TABLE 6.5: Computational requirement per block of W samples for Nth order adaptive predictor

Finally, as will be noticed in the SNRSEG(dB) or total SNR(dB) values in Table 6.4 are lower than those presented in Chapter V. This is due to the profound differences between the speech signals employed in Chapters V and VI, and to a lesser extent because of the differences in the sampling rate. It will be reminded that the sentence used with ADPCM-AQJ/AQF was almost entirely voiced, whereas the utterances used in evaluating the DPCM-AQF systems of this chapter were composed of

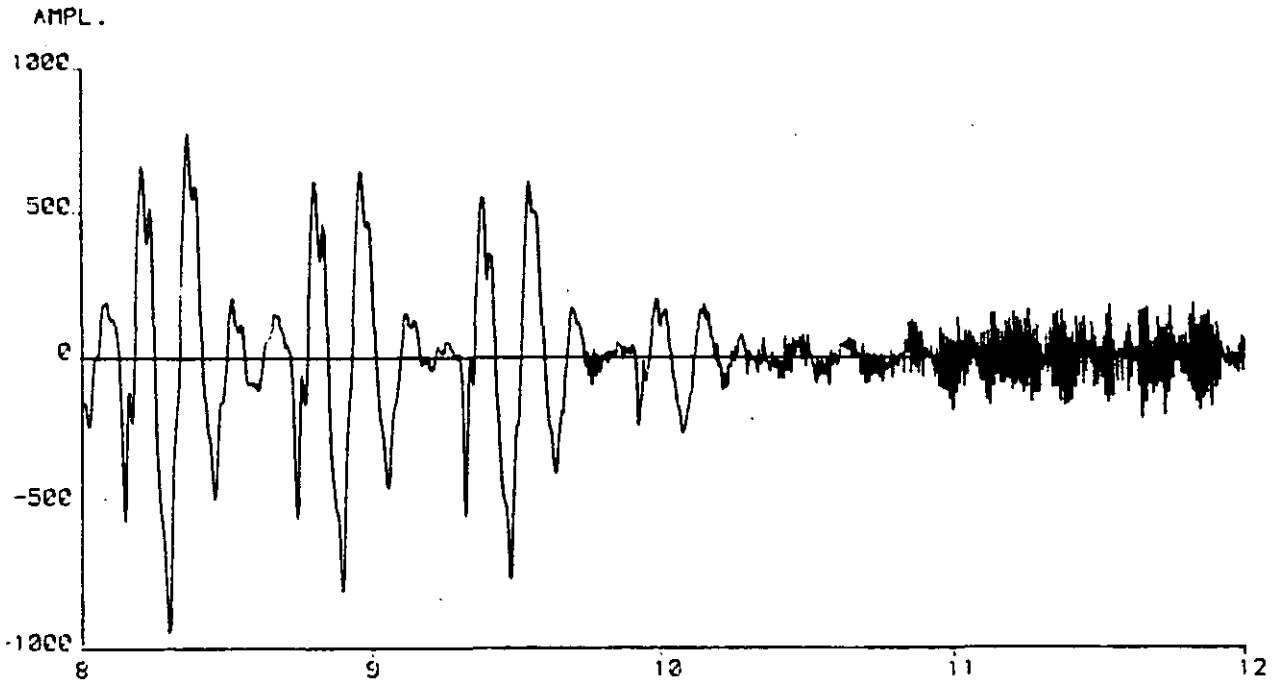
relatively long periods of unvoiced speech. From Figure 6.6(d), b=4 and with a bit-rate of 32 Kb/s, we observe that when voiced speech occurs the SNR is almost 30 dB. However, the average SNR, i.e., SNRSEG(dB) is low due to the unvoiced sections and the unvoiced/voiced transitions. ADPCM-AQJ-SGEP with N=2 and b=4, i.e., at 40 Kb/s, has a SNR of 35-37 dB, see Figure 5.7. When the same codec was used to digitize the 8 kHz speech signal, the SNRSEG(dB) was found to be 22.10 dB while ADPCM-AQF-SGEP resulted in the SNRSEG(dB) of 21.36 dB, see Table 6.4. A small difference of 0.74 dB is attributed to the use of backward rather than forward quantization in the latter case. As a summary, the results of Figure 5.7 and Table 6.4 are consistent, although they are very different, highlighting the dangers of judging the performance of a system in terms of absolute values of SNRSEG(dB) or SNR(dB). What matters is their relative values and this is how the results of Figure 5.7 and Table 6.4 should be appraised.

6.5 COMPUTER SIMULATION RESULTS AND DISCUSSION OF WIDEBAND QUALITY DPCM-AQF SPEECH CODECS

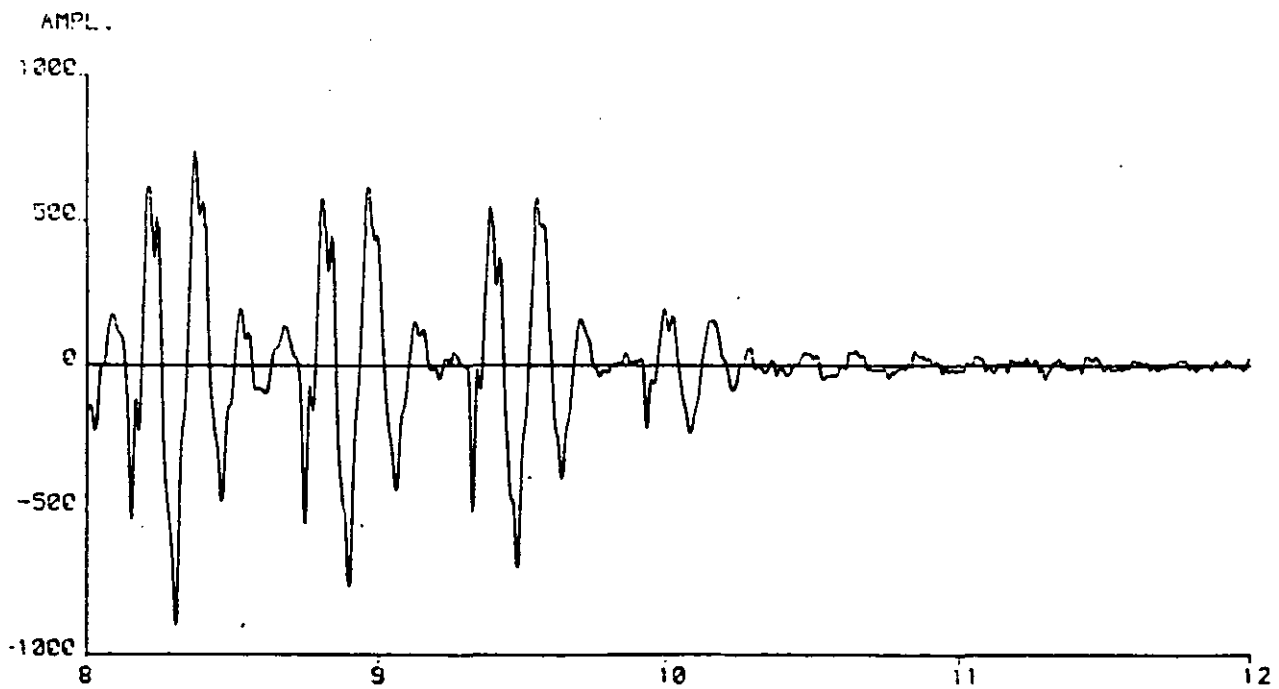
Investigations in the performance of the proposed correlation switched prediction in DPCM-AQF codecs, when connected in tandem with the VUBS system, were carried out by computer simulation. As in the previous section, the comparison of the performance was ascertained by the waveforms and SNRSEG(dB) values selected at the centre of the codec's dynamic range.

The original Wide Band Speech Signal, WBS, whose frequency band is limited to 6.0 kHz was sampled at 16 kHz while the 0.3-3.4 kHz Narrowband Processed Speech Signal, NPSS at the output of VUBS preprocessor was sampled at the rate of 8 kHz. NPSS was then encoded with 2,3 and 4 bits/sample while, for comparison purposes, WBS was encoded with 1 and 2 bits/sample. The SNRSEG(dB) results and the waveform plots presented in this section were obtained from the encoding of the utterance *"sister, father"*, through differential codecs or the VUBS system followed by differential encoding.

Figure 6.8(a) shows the /IS/ segment of the 6.0 kHz WBS waveform taken from the word *"sister"*. When the bandwidth is limited to that of a telephone channel (0.3-3.4 kHz), the /IS/ waveform is shown in Figure 6.8(b). These two figures suggest that the unvoiced sound /s/ is significantly distorted by the bandwidth limitation whereas the voiced sound /I/ is almost unaffected. Figure 6.8(c) shows the same speech segment at the output of the VUBS preprocessor, having also a


AMPL.
1000
500
0
-500
-1000
8
9
10
11
12
BLOCK NO.

(a)


AMPL.
1000
500
0
-500
-1000
8
9
10
11
12
BLOCK NO.

(b)

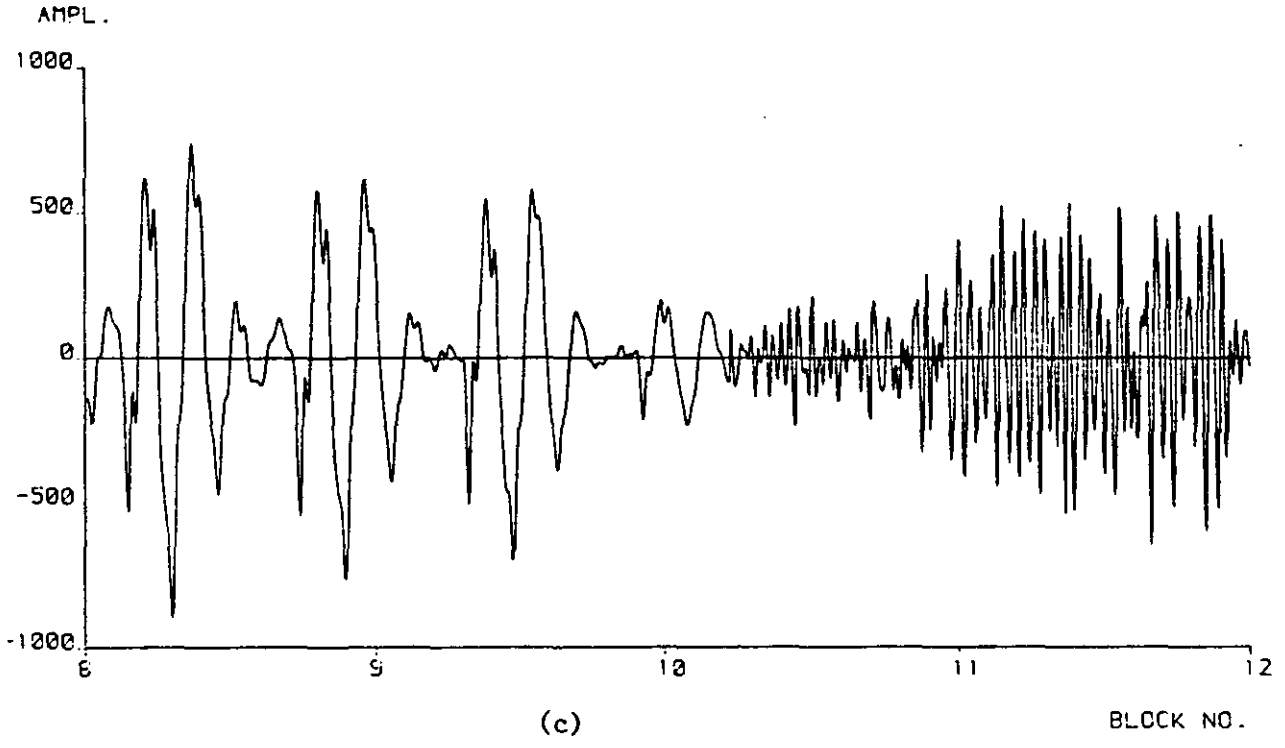


FIGURE 6.8: Time Waveforms for /IS/ in sister

a. Original Wideband Speech Signal (WBS), 0.3 to 6.0 kHz,

b. Bandlimited Speech Signal (BLS), 0.3 to 3.4 kHz,

c. VUBS Compressed Signal (NPSS), 0.3 to 3.4 kHz, (unvoiced segment is amplified)

bandwidth of 3.4 kHz. Figure 6.9 presents the waveform plot of the word *"sister"*, sampled at 16 kHz and bandlimited to 0.3-6.0 kHz, WBS.

Our consideration was given to the differential encoding of the compressed signal for bit rates between 16 and 32 Kb/s. The DPCM-AQF parameters were α_q=0.33, 0.50 and 1.0 for b=4,3,2 bits, respectively and W=256. The SGEP parameters were A=5.0, B=100.0, D=10.0, β=1/5, α/β=4 and the initial SGEP coefficients were a_1=0.86 and a_2=0.0. Notice that, these parameters are the same as those used in Section 6.4. The prediction coefficients of the fixed second order predictor were a_1=1.520 and a_2=-0.763 and they were defined from the long term autocorrelation function of the compressed signal.

The waveform plot of the word, *"sister"*, resulted at the output of the preprocessor and subsampled at 8 kHz, is displayed in Figure 6.10, together with the first correlation coefficient c_1 of each block of 256 samples. Curves (a) and (b) in Figures 6.11, 6.12 and 6.13 show SNR variation of DPCM-AQF-FSOP and ADPCM-AQF-SGEP codecs as a function of the block number for the preprocessed segment *"sister"*, for b=2,3, and 4 bits. Note that the statistical characteristics of the *"unvoiced"* sounds at the output of the VUBS preprocessor are different from those of either the original wideband speech or the 0.3-3.4 kHz input speech signal, see Figures 6.10, 6.9, 6.3 respectively.

Figures 6.11, 6.12, 6.13 indicate the poor performance of DPCM-AQF-FSOP during unvoiced sounds. This is attributed to the fact that voiced speech constitutes up to 80% of the speech signal, and the *"average"*

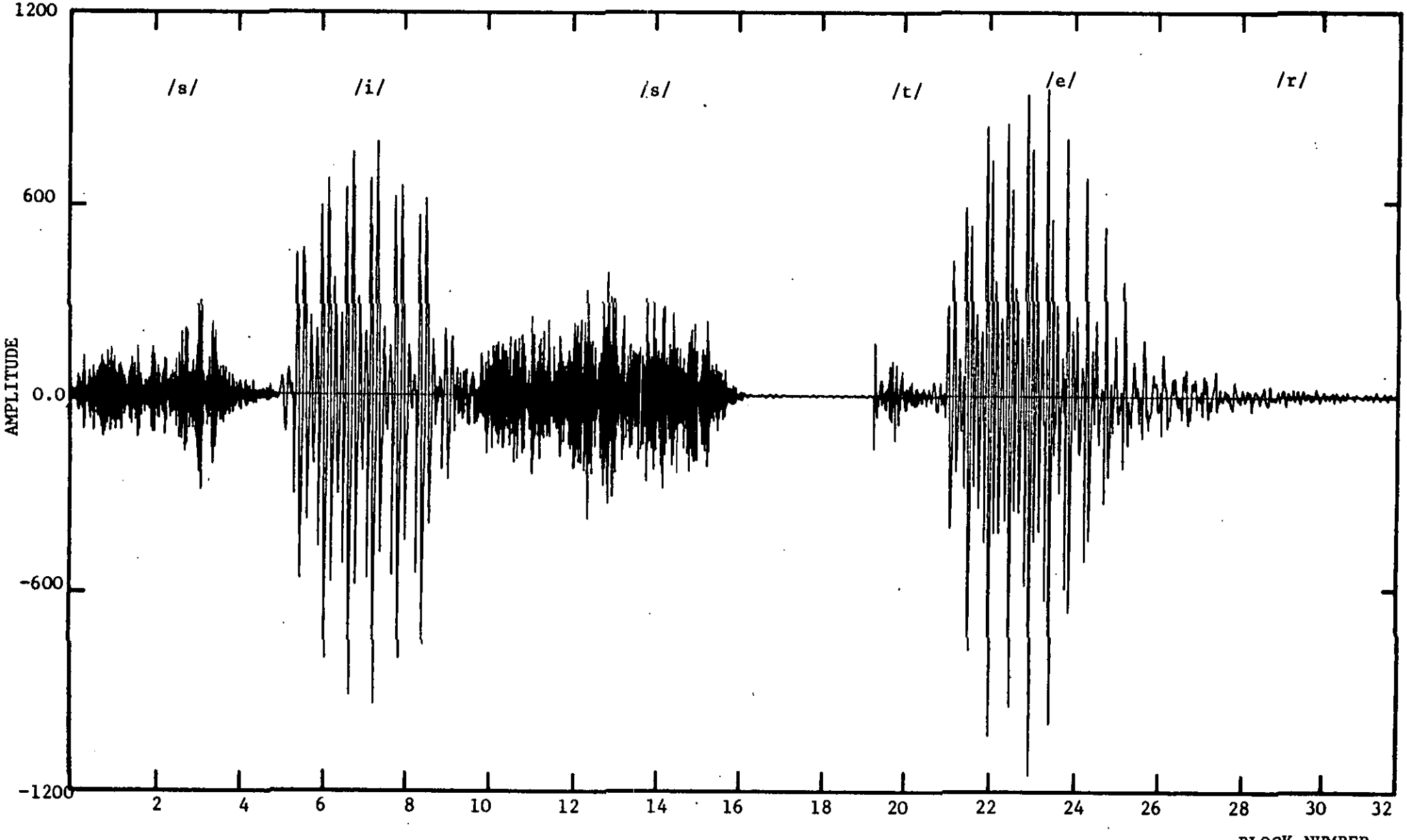


FIGURE 6.9: The Waveform of Utterance *"sister"* for Wideband Speech Data (WBS)

coefficients of the second-order fixed predictor are matched to voiced rather than to unvoiced speech. Notice that DPCM-AQF-FSOP provides lower *"unvoiced"* SNR values when operating on the preprocessed signal (Figures 6.11 to 6.13) than when digitizing the band limited input speech signal (Figures 6.4 to 6.6). When FSOP is replaced by SGEP, the overall SNR performance of the DPCM-AQF codec is improved for all the bit rates. DPCM-AQF-FSOP is producing however, slightly better peak SNR values for voiced speech, see Figures 6.11 to 6.12. This is because the same SGEP parameters were used as in Section 6.4 and thus the adaptation rate of SGEP was not optimized with respect to the signal at the output of the VUBS preprocessor. Also, as observed in Section 6.4, SGEP looses its advantage over FSOP during the intervals of unvoiced to voiced transitions, see blocks 4 and 12 in Figure 6.10. (There is a delay of one block between Figures 6.3 and 6.10. This is due to VUBS preprocessor of Section 6.3).

In order to improve the performance of the DPCM-AQF codecs, we employed the *"Correlation Switched Prediction"* method with FSOP and SGEP. As the statistical properties of VUBS preprocessed signal are different than those of the 0.3 to 3.4 kHz band limited input speech signal, new CSP look-up tables were determined using the same procedure described in Section 6.2. Table 6.6 illustrates the thresholds and the values of the a_1'', a_2'' prediction coefficients to be used in a 4-point CSP, designed to operate on the preprocessed VUBS signal. The variation of SNR(dB) as a function of block number, resulting from DPCM-AQF employing a CSP(4)-FSOP predictor, is shown in Figures 6.11

to 6.13 curve (c) for b=2,3 and 4 bits, respectively. The ADPCM-AQF-CSP(4)-FSOP codec has, in general, an improved performance during unvoiced to voiced transitions and unvoiced intervals, when compared to both the ADPCM-AQF-SGEP and DPCM-AQF-FSOP codecs. The CSP(4)-SGEP algorithm was then used to predict the incoming VUBS processed samples and Table 6.6 was also employed to determine the initial values of the SGEP prediction coefficients at the beginning of a block of 256 samples. If, however, the correlation coefficient, c_1 does not change zone in two adjacent blocks of samples, the initial values of the prediction coefficients for the 256 samples to be encoded are the last coefficient values, obtained during the encoding of the previous block of samples. This provides a faster convergence rate than that of the SGEP algorithm. A careful inspection of curves (d) in Figures 6.11 to 6.13 shows that DPCM-AQF-CSP(4)-SGEP codec produces the best overall performance when compared to all the other codecs.

Table 6.7 presents the SNRSEG(dB) values of DPCM-AQF codecs, employing different prediction schemes, when encoding band limited, BLS or preprocessed, NPSS, signals. For comparison purposes, wideband speech signal, WBS was also encoded with 1 or 2 bits/sample, i.e., only fewer bits per sample are available compared to digitization of the BLS or NPSS signals sampled at 8 kHz.

When FSOP scheme was employed in DPCM-AQF codec for encoding of WBS signal at 32 Kb/s, the SNRSEG(dB) was found to be 14.21 dB. The predictor coefficients were calculated from the long-term statistics of the WBS signal.

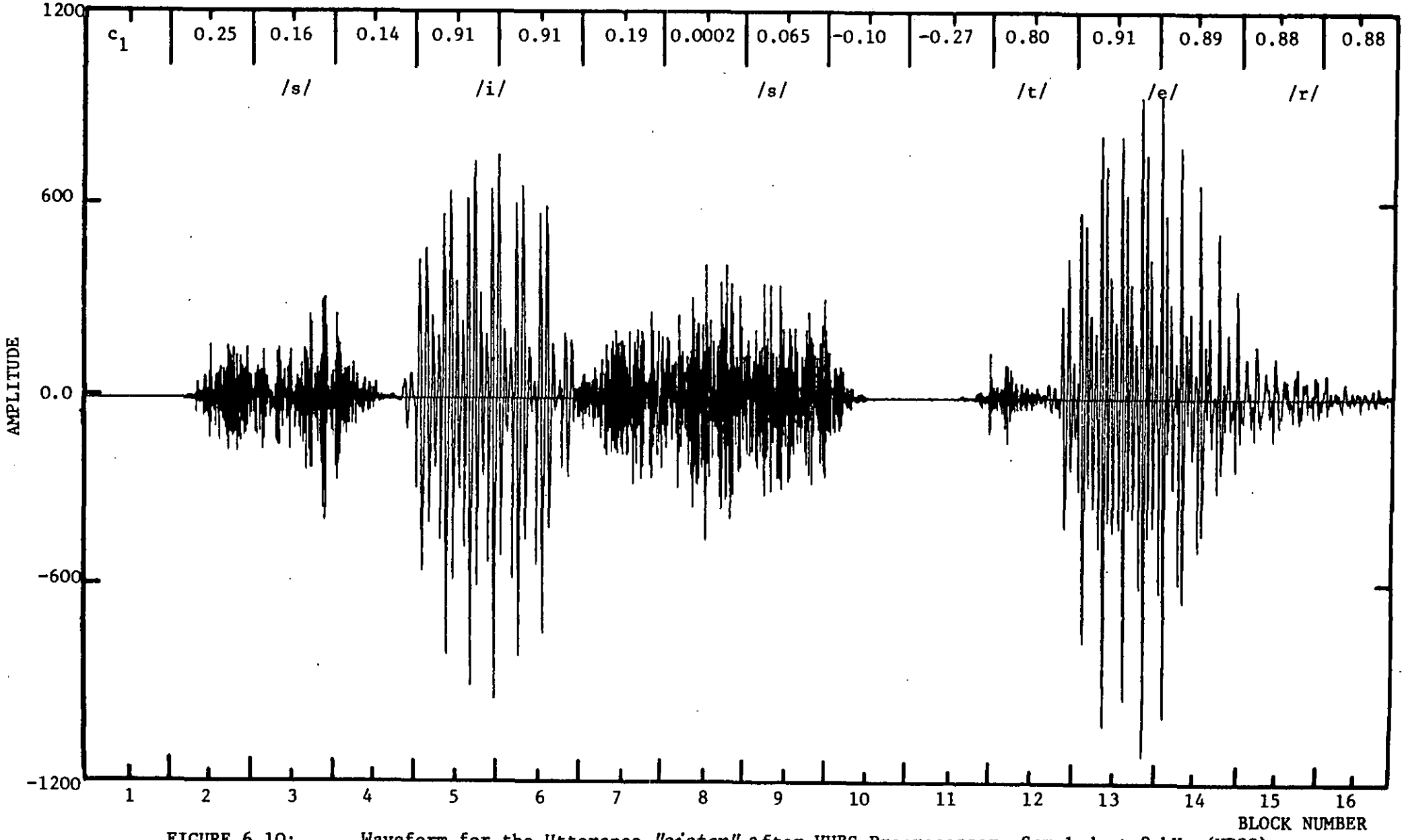


FIGURE 6.10: Waveform for the Utterance *"sister"* after VUBS Preprocessor, Sampled at 8 kHz (NPSS)

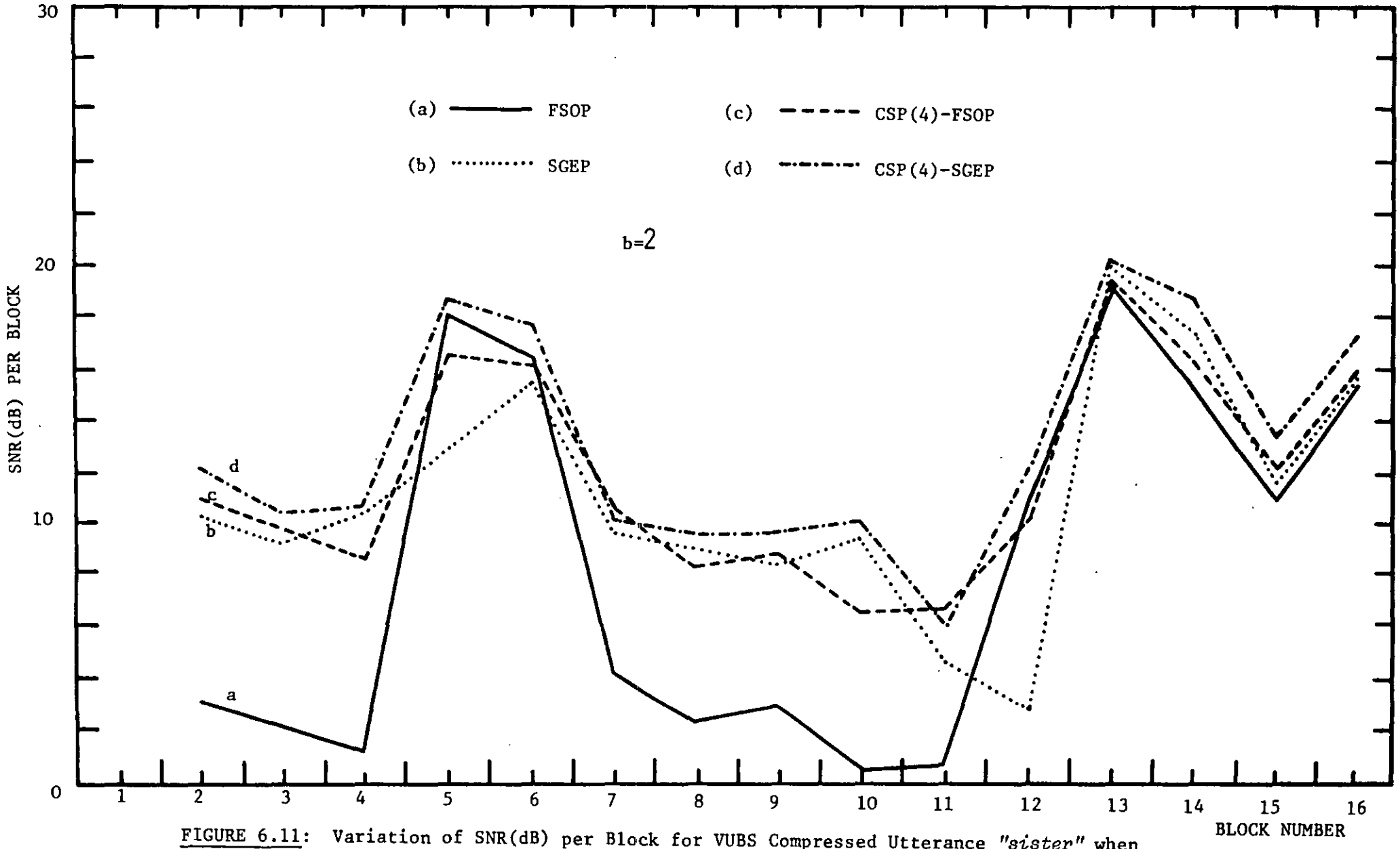


FIGURE 6.11: Variation of SNR(dB) per Block for VUBS Compressed Utterance *"sister"* when Encoded by 2 bits/sample DPCM-AQF Encoder

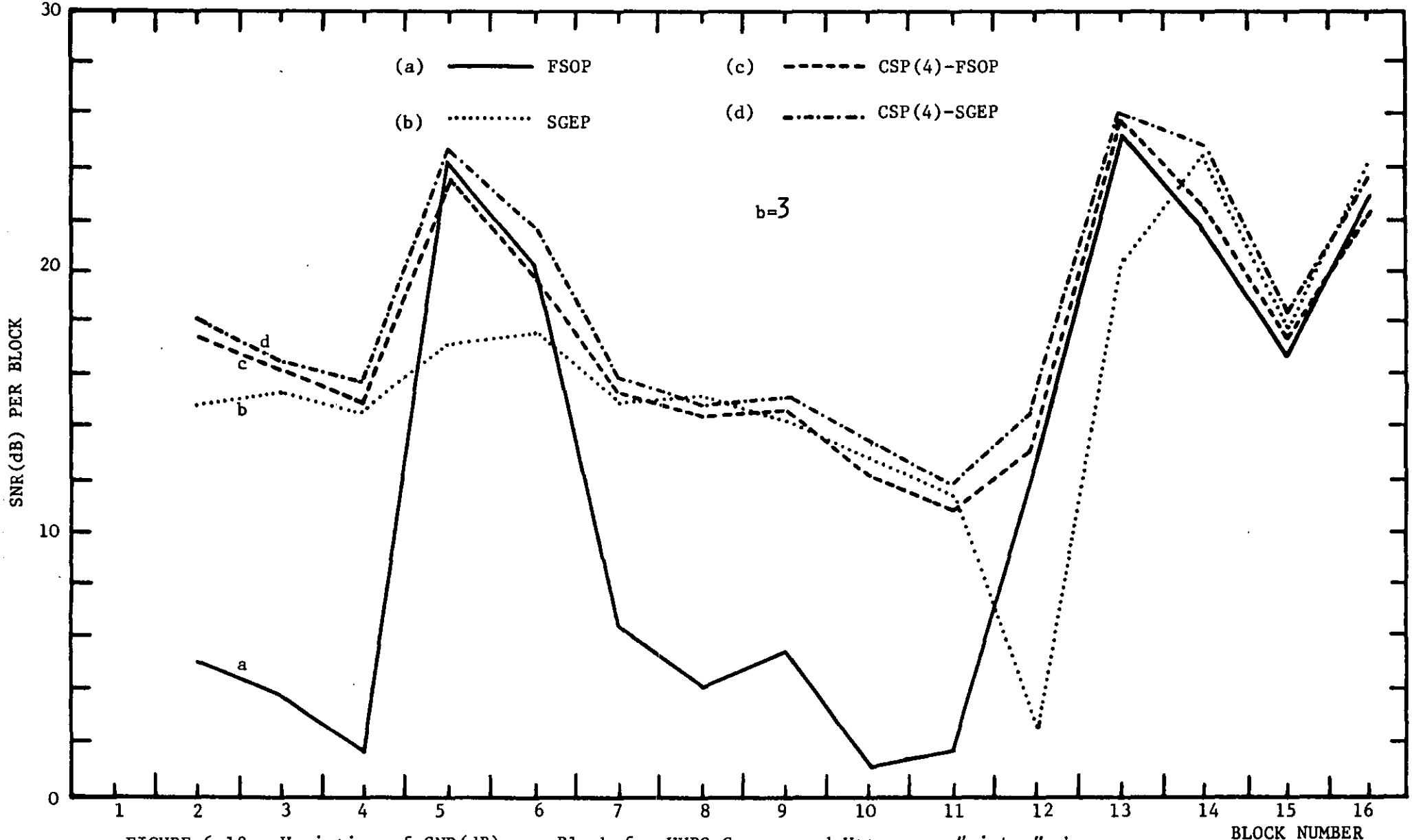


FIGURE 6.12: Variation of SNR(dB) per Block for VUBS Compressed Utterance *"sister"* when Encoded by 3 bits/sample DPCM-AQF Encoder

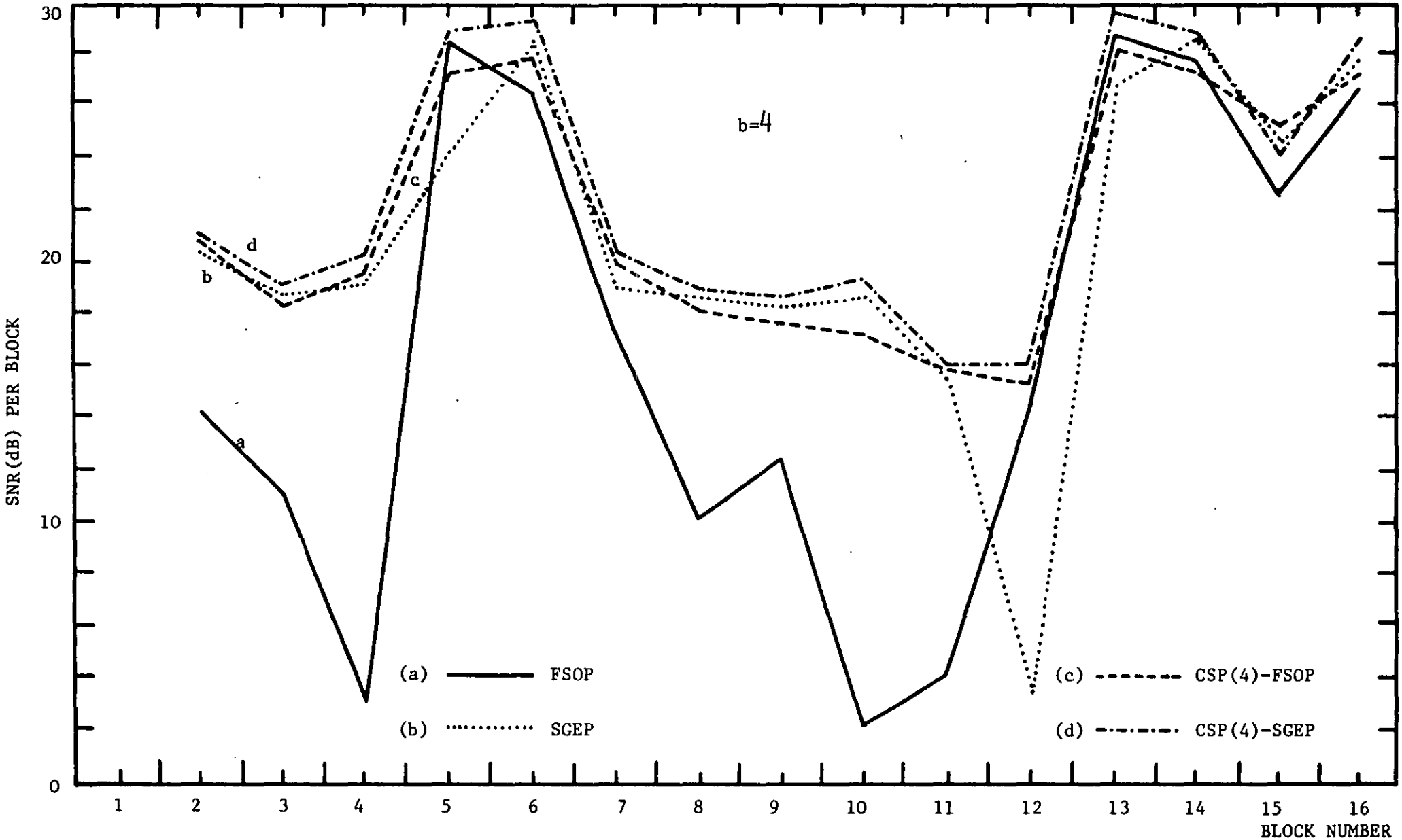


FIGURE 6.13: Variation of SNR(dB) per Block for VUBS Compressed Utterance *"sister"* when Encoded by 4 bits/sample DPCM-AQF Encoder

j	THRESHOLD TR_j	CORRELATION ZONE	COEFFICIENT a''_1	COEFFICIENT a''_2
1	0.7	0.7 to 1.0	1.520	-0.763
2	0.2	0.2 to 0.7	0.510	-0.380
3	0.0	0.0 to 0.2	0.060	-0.750
		-1.0 to 0.0	-0.322	-0.490

TABLE 6.6: Look-Up Table for 3rd-Order or 4-Point CSP Based on VUBS Preprocessed Output

TYPE OF PREDICTION IN DPCM-AQF	BITS PER SAMPLE			TYPE OF INPUT SIGNAL
	2	3	4	
FSOP	11.41	15.26	19.92	BANDLIMITED
	10.47	14.94	19.58	VUBS COMPRESSED
SGEP	11.93	16.96	21.84	BANDLIMITED
	11.90	16.62	21.17	VUBS COMPRESSED
CSP(4)-FSOP	13.22	17.67	22.24	BANDLIMITED
	12.21	17.01	21.83	VUBS COMPRESSED
CSP(4)-SGEP	13.45	18.20	23.67	BANDLIMITED
	12.86	18.04	22.81	VUBS COMPRESSED

TABLE 6.7: SNRSEG(dB) of DPCM-AQF with Second Order Predictors for the Utterances *"sister, father"*

Notice that the SNRSEG(dB) values of all DPCM-AQF codecs, operating on the preprocessed signal, are marginally lower than the SNRSEG(dB) obtained when the input signal is speech bandlimited to 3.4 kHz. However, according to informal listening experiences, the VUBS system combined with any of the examined digitizers, seems to produce recovered speech of better quality and improved intelligibility when compared to the corresponding DPCM-AQF encoding of bandlimited speech signals. The reason is that although the codec SNRSEG(dB) performance for the BLS signal looks superior to the SNRSEG(dB) obtained for the VUBS pre-processed signals, the perceptually significant high frequencies of unvoiced sounds are missing from the recovered bandlimited speech. Additionally, degradation of 5 dB's in SNRSEG values, resulting from DPCM-AQF-FSOP when encoding process of WBS is attributed to a higher frequency components that exist in WBS signal and these components cause slope overload to occur more frequently[133] in differential encoders compared to encoding of 0.3-3.4 kHz.

6.6 CONCLUSIONS

In this chapter the performances of DPCM-AQF codecs have been examined. The emphasis in the presentation was devoted to a novel prediction technique called the *"Correlation Switched Prediction"*, CSP. The input speech signal contained numerous unvoiced/voiced transitions, unlike the input sentence used in the previous chapters. This coupled with the lower sampling rate of 8 kHz, exacerbates the difficulties of the predictors. These difficulties were intentionally introduced in order to determine the effectiveness of switched predictors in enhancing the performance of a codec.

"Correlation Switched Predictors" significantly modify the values of the prediction coefficient according to a simple statistical property of the speech signal, namely the first correlation coefficient, c_1. Employing a 4-point CSP scheme associated with FSOP, and then SGEP, the performance of DPCM-AQF codec was examined for each block of samples. It was observed that CSP(4)-SGEP, in a DPCM-AQF codec, handles unvoiced to voiced transitions efficiently, and it also improves the SNR values for voiced speech.

Computer simulation results for the range of 16 to 32 Kb/s, also indicated an improved SNRSEG(dB) performance when using FBA instead of Backward Sequentially adaptive prediction. The introduction, in DPCM-AQF, of the relatively simple CSP scheme produced SNRSEG(dB) values comparable to those obtained from FBADPCM-AQF. A more complex prediction scheme which combines CSP and Backward Sequentially Adaptive Prediction, CSP(4)-SGEP, was shown to provide the best overall SNRSEG(dB) performance.

Gibson[121] has demonstrated that SNR gain of approximately 2 dB can be obtained by using a pitch compensating quantizer instead of Jayant's quantizer in DPCM codec. It has been shown here that a similar SNR improvement can be achieved when the codec employs efficient switched prediction technique together with FSOP or SGEP.

The second part of this chapter presents computer simulation results of DPCM-AQF codecs, in tandem with the VUBS system. The VUBS system[12] bandpass filters the voiced speech to 0.3 or 3.4 kHz, and also compresses the unvoiced speech, occupying the frequency band of 0.3 to 6.0 kHz, into the 0.3 to 3.4 kHz band. Consideration was given to the digital encoding of the VUBS compressed signals for bit-rates between 16 and 32 Kb/s. The DPCM-AQF codecs used in our computer simulation studies employed the FSOP, SGEP, CSP(4)-FSOP or CSP(4)-SGEP prediction algorithms. The performance evaluation of the various codecs was based on waveforms, SNRSEG(dB) values, and informal listening tests. It was found that the SNRSEG(dB) of the ADPCM-AQF-CSP(4)-SGEP encoder operating on the VUBS compressed signals, is slightly reduced, compared to the SNRSEG(dB) of the same encoder digitizing 0.3 to 3.4 kHz band-limited speech signals. However, an improved quality speech is obtained from the VUBS/DPCM-AQF system due to the better reproduction of unvoiced sounds.

6.7 NOTE ON PUBLICATIONS

A paper entitled *"Sequential Adaptive Predictors for ADPCM Speech Encoders"*,[134] in co-authorship with Dr. C.S. Xydeas and Dr. R. Steele has been published in National Telecommunications Conference Proceedings, NTC81, New Orleans, U.S.A., pp. E8.1.1-5, November 1981.
This paper is a version of the 4-point CSP schemes described in Sections 6.2 and 6.4.

A paper entitled *"A Comparative Study of DPCM-AQF Speech Codecs for Bit-Rates of 16 to 32 Kb/s"*,[135] in co-authorship with Dr. C.S. Xydeas has appeared in the IEEE Int. Conf. Proceedings on ASSP, ICASSP82, Paris, France, Session S14, May 1982.
This paper is an abridged version of Sections 6.1, 6.2 and 6.4.

CHAPTER VII

RECAPITULATION

7.1 INTRODUCTION

In this thesis a number of novel digitization systems, for speech signals, have been proposed and investigated. These digitizers were designed to achieve:

a) the best possible quality of recovered speech at a given transmission bit rate,

b) a modest implementation complexity and therefore low cost,

c) a characteristic of robustness to the transmission errors.

Our investigations were focussed on waveform encoding techniques operating with a bit rate of 16 to 40 kb/s.

Differential Pulse Code Modulation (DPCM) is the central theme of the thesis. The performance of DPCM coders depends on: a) the estimation efficiency of the predictor, and b) the performance of the quantizer used in the system. Adaptive quantization, as used in differential digitizers, has been studied in depth and several algorithms have been proposed. The prediction problem however, has received less attention and therefore our research efforts were directed towards the development of improved speech prediction algorithms.

Initially the existing fixed (time-invariant) and Forward Block Adaptive, FBA prediction methods were considered and used in DPCM. Taking SNR(dB) as the measure of encoding performance, the relative merits of these two prediction schemes have been pointed out.

Sequential prediction algorithms were also considered, in the form of the Stochastic Approximation Predictor, SAP. Because of limitations

in the performance of SAP, a novel algorithm called the Sequential Gradient Estimation Predictor, SGEP was developed and compared with SAP.

Next, the performance of DPCM codecs employing both adaptive quantizer and sequential predictors was evaluated for transmission bit rates of 16-40 Kb/s. Finally, searching for prediction methods that would improve the encoding performance of the previous DPCM system during the Voiced/Unvoiced transitions in the speech waveform, the concept of the *"Correlation Switched Prediction"* was developed. These predictors switch their coefficients according to a simple statistical measure of the speech, i.e., first shift correlation coefficient.

All the adaptive prediction schemes and the performance results of the speech codecs examined are summarized in the subsequent sections. Suggestions for further research are also made in a number of topics which may be of value in the area of speech coding.

7.2 DPCM EMPLOYING FIXED OR BLOCK ADAPTIVE PREDICTION

Chapter III deals with fixed and FBA prediction when applied in DPCM codecs. The design of fixed predictors is accomplished using the autocorrelation method. An upper bound improvement factor, SNRI is determined for the speech signals used in our experiments. It is shown that the high correlation which exists between the speech samples manifested itself in high SNRI values. The best overall performance in DPCM-AQJ operating at bit rates of 16-40 Kb/s with fixed prediction was achieved when the order of fixed predictor was N=2. At 16 Kb/s, the codecs employing higher-order predictors experienced instability. This is attributed to the mismatching effect of the higher-order prediction coefficients to the decoded samples.

The performance of DPCM encoders was further improved by using Forward Block Adaptive prediction, FBA,where the prediction coefficients are updated periodically. Operating at high transmission bit rates (>30 Kb/s), FBA scheme showed a peak SNR advantage of 5 dB's over that of fixed prediction algorithm (N=8), but the transmission of the prediction coefficients was required as side information.

Looking back to the section dealing with DPCM employing fixed or FBA prediction schemes, we can itemize the limitations and disadvantages of these systems as follows:

a) The fixed (time-invariant) predictor attempts to model the characteristics of a time varying vocal tract function.

b) Since the voiced speech is more prevalent than unvoiced speech, a fixed predictor designed from the average statistics of the

speech signal gives relatively poor SNR performance during the prediction of unvoiced speech.

c) At low transmission bit rates ($\lesssim$16 Kb/s), higher-order predictors required to operate on severely distorted decoded speech samples, cause system divergence.

d) FBA predictors when used in a DPCM codec, require extra information, namely the values of the prediction coefficients, to be multiplexed with the output of the DPCM quantizer. For example, an 8-bits quantization of the FBA prediction coefficients (N=2) will increase the transmission bit rate by 500 bits.

7.3 SEQUENTIAL PREDICTION ALGORITHMS AND THEIR APPLICATIONS IN DPCM

In Chapter IV, our efforts were directed towards the development of efficient speech prediction algorithms which overcome the limitations inherent in both fixed and block adaptive schemes. The performance of the sequentially adaptive Stochastic Approximation linear predictor, SAP was first examined and its adaptation rate was shown to be inadequate to follow fast variations in the statistics of speech. In an attempt to obtain faster convergence towards the optimum prediction coefficients, a novel technique called the Sequential Gradient Estimation Predictor, SGEP has been devised and studied in depth. The superiority of the SGEP over SAP was ascertained by the error waveforms which indicate that the prediction error signals for SGEP are typically 3 dB lower than those obtained using the SAP algorithm. A mathematical analysis of SGEP and SAP schemes, supported by waveform plots, also indicate that the rate of convergence of SGEP is considerably faster than that of SAP. In addition, it was shown that the ability of SGEP to work efficiently with fewer coefficients, typically N=2, off-sets the increased complexity of the algorithm.

DPCM systems were then simulated on the computer and the codec with a leaky integrator in its feedback loop was used as a performance bench-mark due to its virtue of simplicity compared with its adaptive predictor counterparts. It was found that as quantization accuracy increases from 2 to 4 bits per sample, DPCM codecs using Jayant's adaptive quantizer and the SGEP prediction algorithm, ADPCM-AQJ-SGEP, show an overall SNR advantage of approximately 2 to 3.5 dB's over the

SNR of ADPCM-AQJ-SAP. Thus, the better prediction accuracy of SGEP over SAP is enhancing the performance of the codec. When the DPCM-AQJ codec used a fixed first-order predictor, DPCM-AQJ-FFOP, its SNR compared to that of ADPCM-AQJ-SGEP was considerably reduced.

Next, another quantization strategy namely adaptive quantization with forward transmission of step size, AQF, was adopted and applied to differential encoding systems, DPCM-AQF. The reason our investigations were focussed on DPCM-AQF encoded speech was that such an encoder is inherently more robust to the transmission errors. Second-order predictors have been used in our DPCM-AQF experiments. It was found that, at 40 Kb/s, ADPCM-AQF-SGEP shows an overall SNRSEG(dB) advantage of approximately 3 and 9 dB's over the SNRSEG(dB) values of the ADPCM-AQF-SAP and DPCM-AQF-FFOP systems, respectively. When transmission errors were introduced ADPCM-AQF-SGEP has a higher SNRSEG(dB) than that achieved with DPCM-AQF-FFOP for BER<0.08%.

7.4 CORRELATION SWITCHED PREDICTORS EMPLOYED IN DPCM

Among the codecs employing FFOP, FSOP or SGEP prediction algorithms, a deterioration in the performance was observed when the speech signal transgressed from unvoiced to voiced segments. In order to overcome this, in Chapter VI we searched for alternative prediction algorithms. As a consequence a new correlation switched prediction, CSP scheme was proposed. A switched prediction algorithm divides the range of the first correlation coefficient of the speech signal, c_1, into zones and as the value of c_1 varies, when computed over successive blocks of W speech samples, the predictor coefficients are altered significantly (switched). When the range of c_1 is divided into (Z+1) zones, the predictor was referred to as (Z+1)-point or Z-order CSP, CSP(Z+1). 3rd- and 7th order (Z=3 or 7) switched predictors, associated with second-order FSOP and SGEP algorithms, were used in DPCM-AQF. When the computational requirements of CSP schemes was considered, it was observed that CSP is the algorithm with the smaller number of multiplications compared with either SGEP or FBA schemes.

Computer simulation studies for the range of 16 to 32 Kb/s indicated the performance superiority of codecs with feedforward block adaptive prediction, FBADPCM-AQF, when compared to DPCM-AQF with fixed or SGEP adaptive prediction. When correlation switched prediction was used, however, the SNRSEG(dB) performance of the differential coders (DPCM-AQF) was found to be comparable to that of FBADPCM-AQF. Finally, the more complex hybrid prediction scheme, CSP-SGEP when used in the same codec, resulted in best overall SNRSEG(dB) performance.

7.5 SUGGESTIONS FOR FURTHER RESEARCH

The topics for further research can include the following:

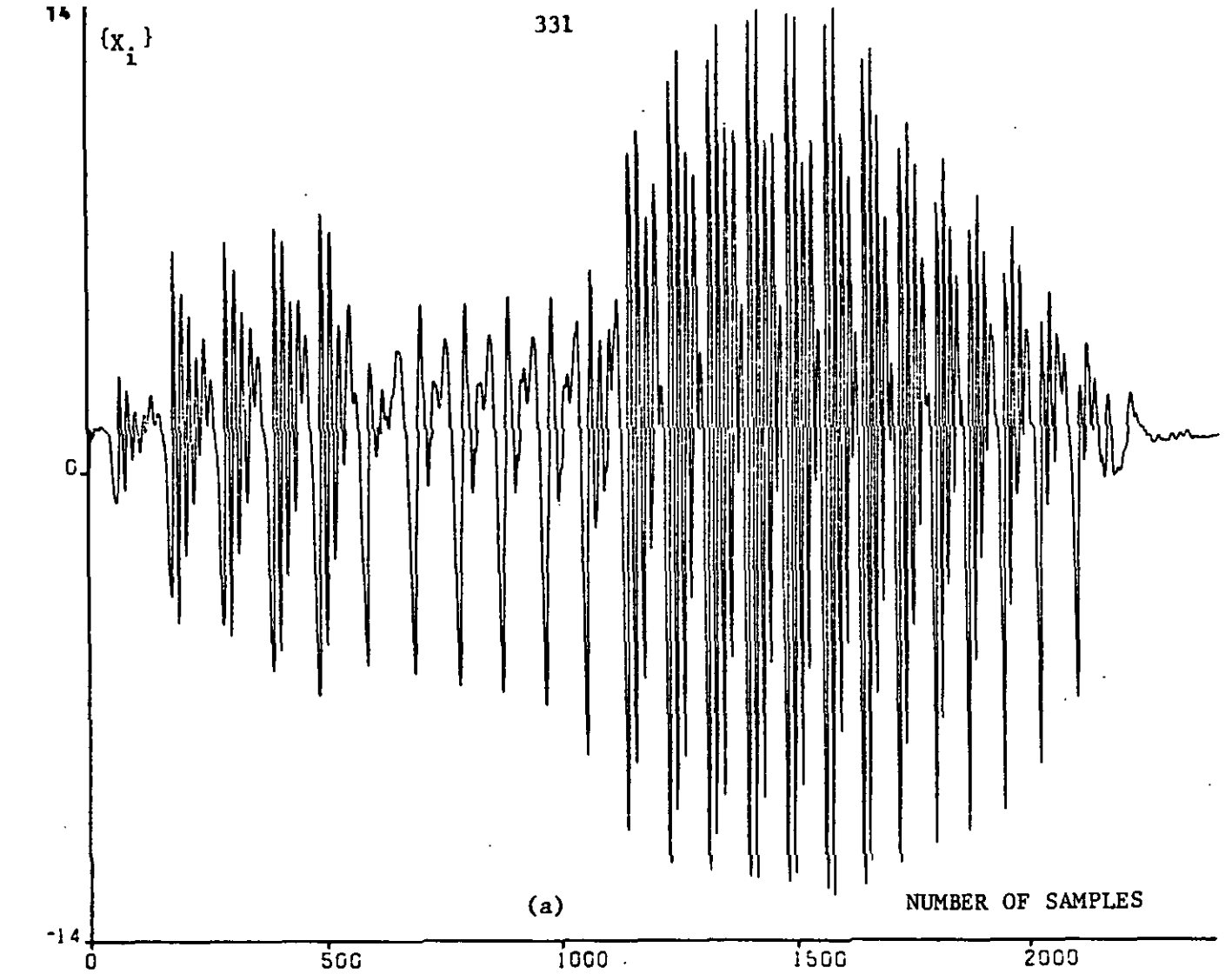
7.5.1 Pitch Extraction Algorithm

It is well-known that pitch extraction is an essential component in a variety of speech processing systems. Besides providing valuable insights into the nature of the excitation source for speech production, the pitch information is required in almost all speech analysis-synthesis (vocoder) systems. Because of its importance, a wide range of algorithms for pitch extraction have already been proposed,[136] from which two broad categories follow, namely, time-domain pitch extraction techniques, i.e., data reduction method, and frequency-domain techniques, i.e., ceptrum method, see Section 2.2.4.

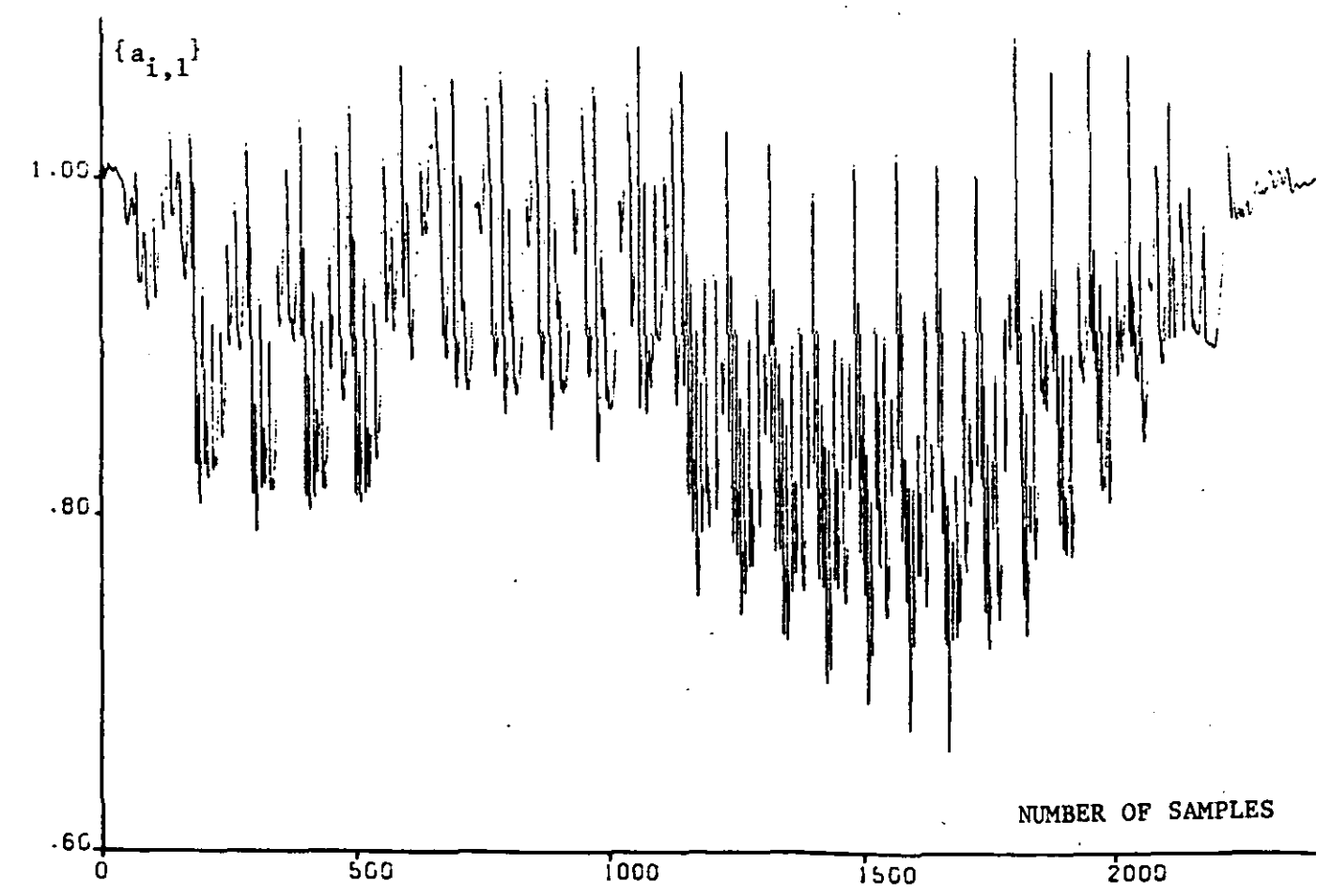
In this respect plots showing the variation of the prediction coefficient, a_1, of a first order SGEP algorithm indicate that the value of the coefficients increases sharply at the onset of each vocal excitation pulse. As a consequence, the first-order SGEP algorithm can be also used for a pitch extraction. In order to locate the peak pulses in the speech signal, see Figure 7.1(a), we initially computed the sequence of a_1 coefficients, $\{a_{i1}\}$, see Figure 7.1(b) and then smoothed it by averaging the a_1 values over the blocks of 25 and 200 speech samples and these sequences were denoted by $\{_{25}(XJ)_i\}$ and $\{_{200}(XJ)_i\}$ respectively, (see Figure 7.1(c) and (d)). These sequences were manipulated as follows:

$$\{(XJ)'_i\} = \{a_i\}-\{_{200}(XJ)_i\} \tag{7.1}$$

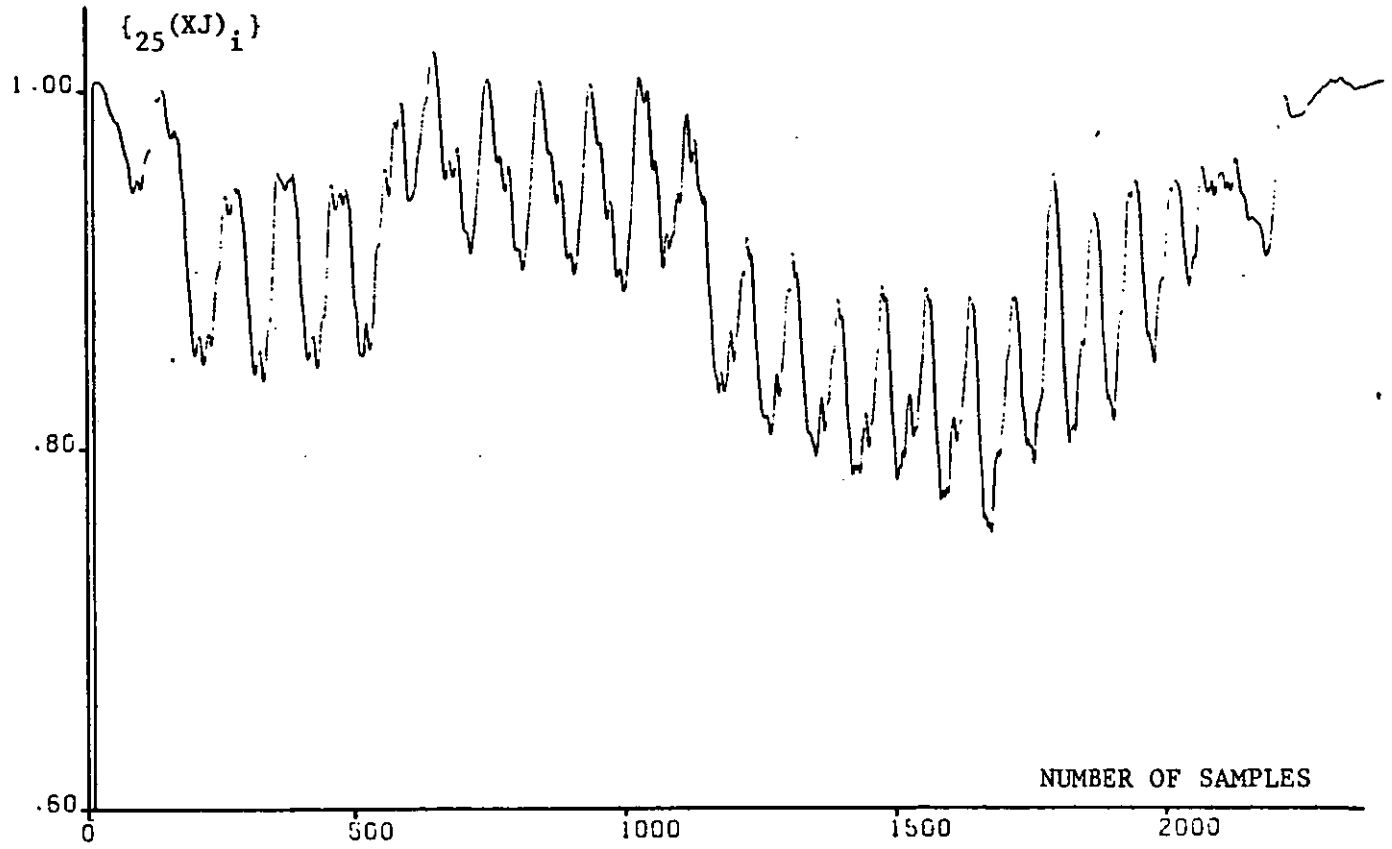
$$\{(XJ)''_i\} = \{_{25}(XJ)_i\}-\{_{200}(XJ)_i\} \; . \tag{7.2}$$


14
{X_i}
0
-14
0
500
1000
1500
2000
NUMBER OF SAMPLES

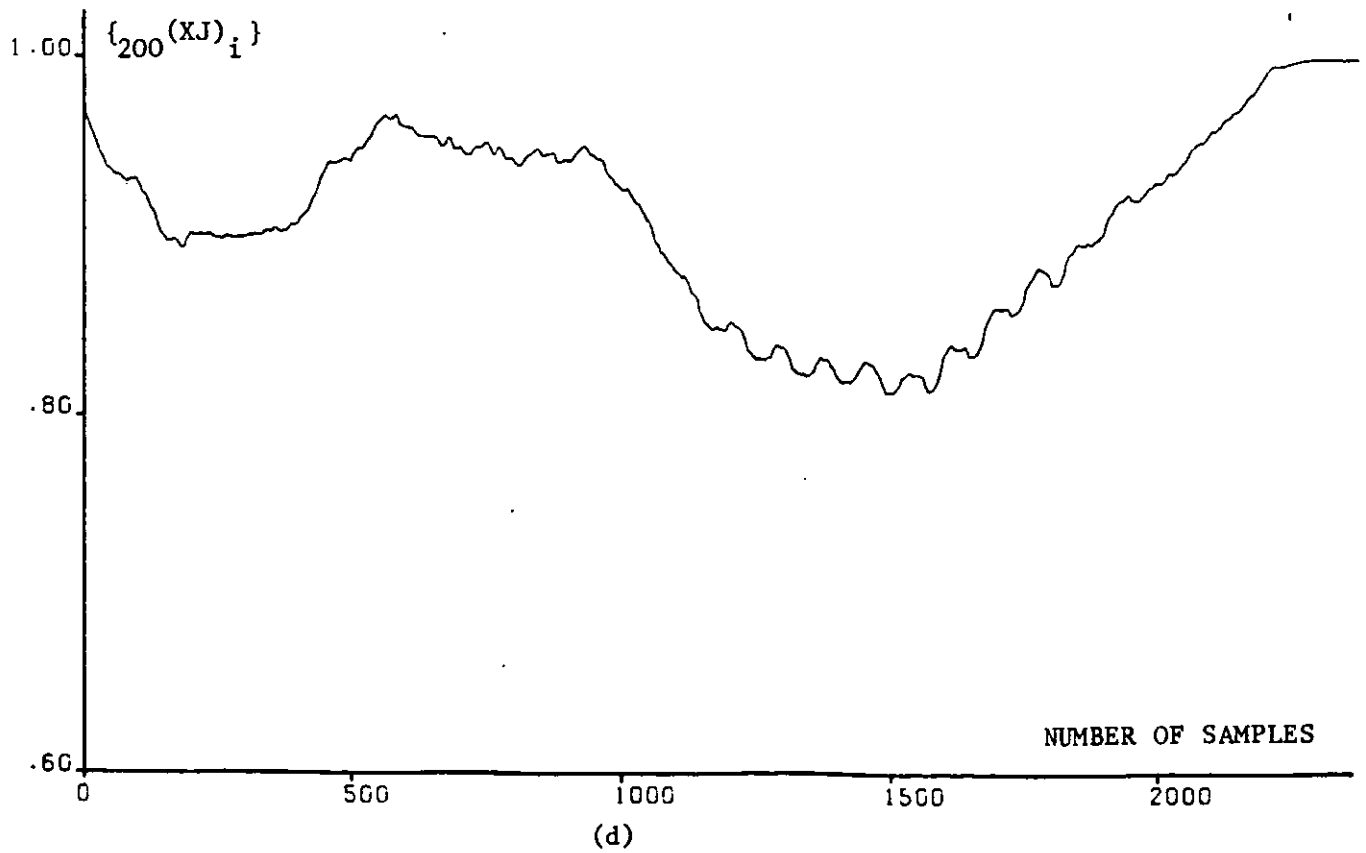
(a)


{a_i,1}
1.00
.80
.60
0
500
1000
1500
2000
NUMBER OF SAMPLES

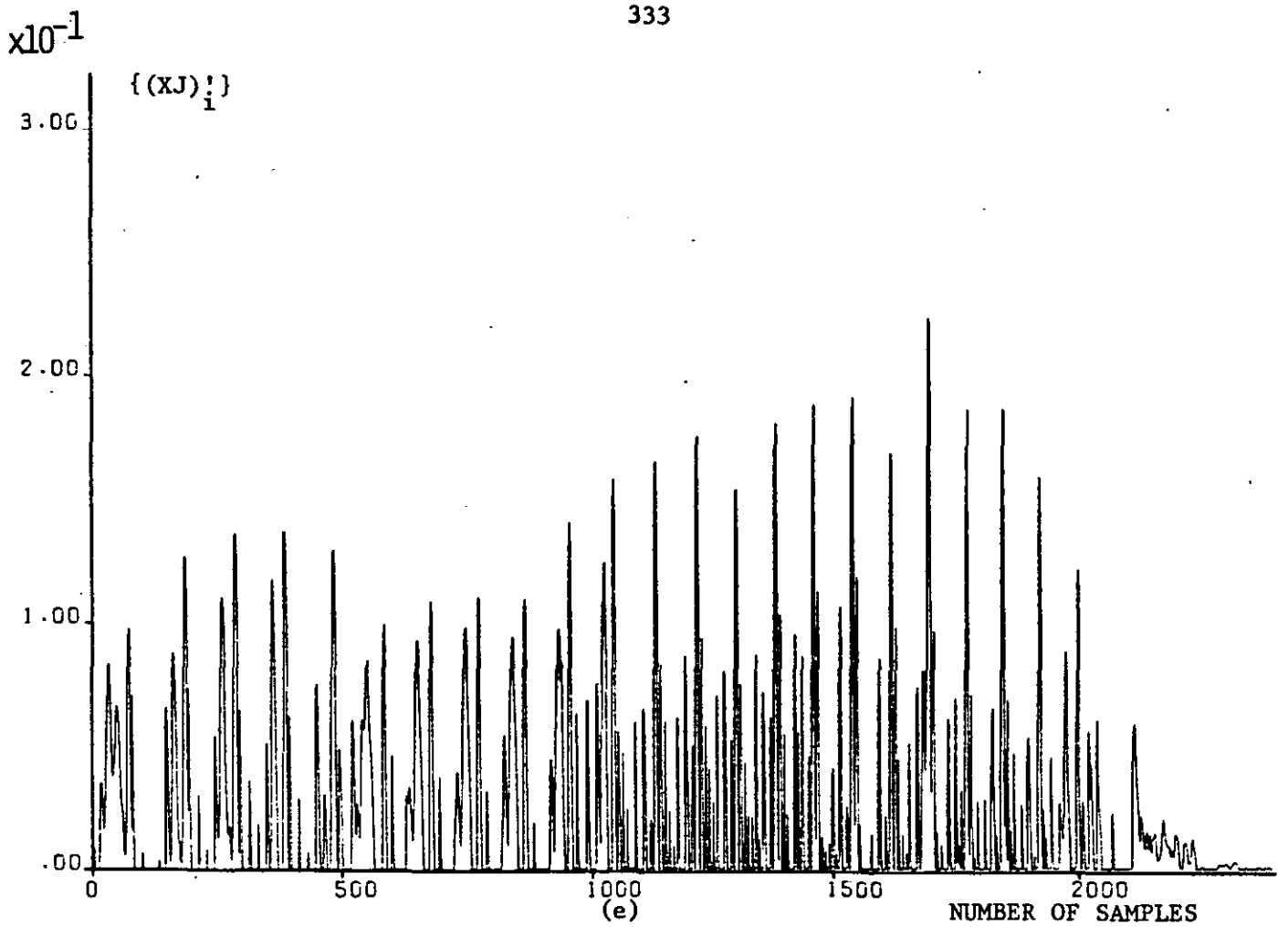
(b)


{25(XJ)i}
1.00
.80
.60
NUMBER OF SAMPLES
0
500
1000
1500
2000

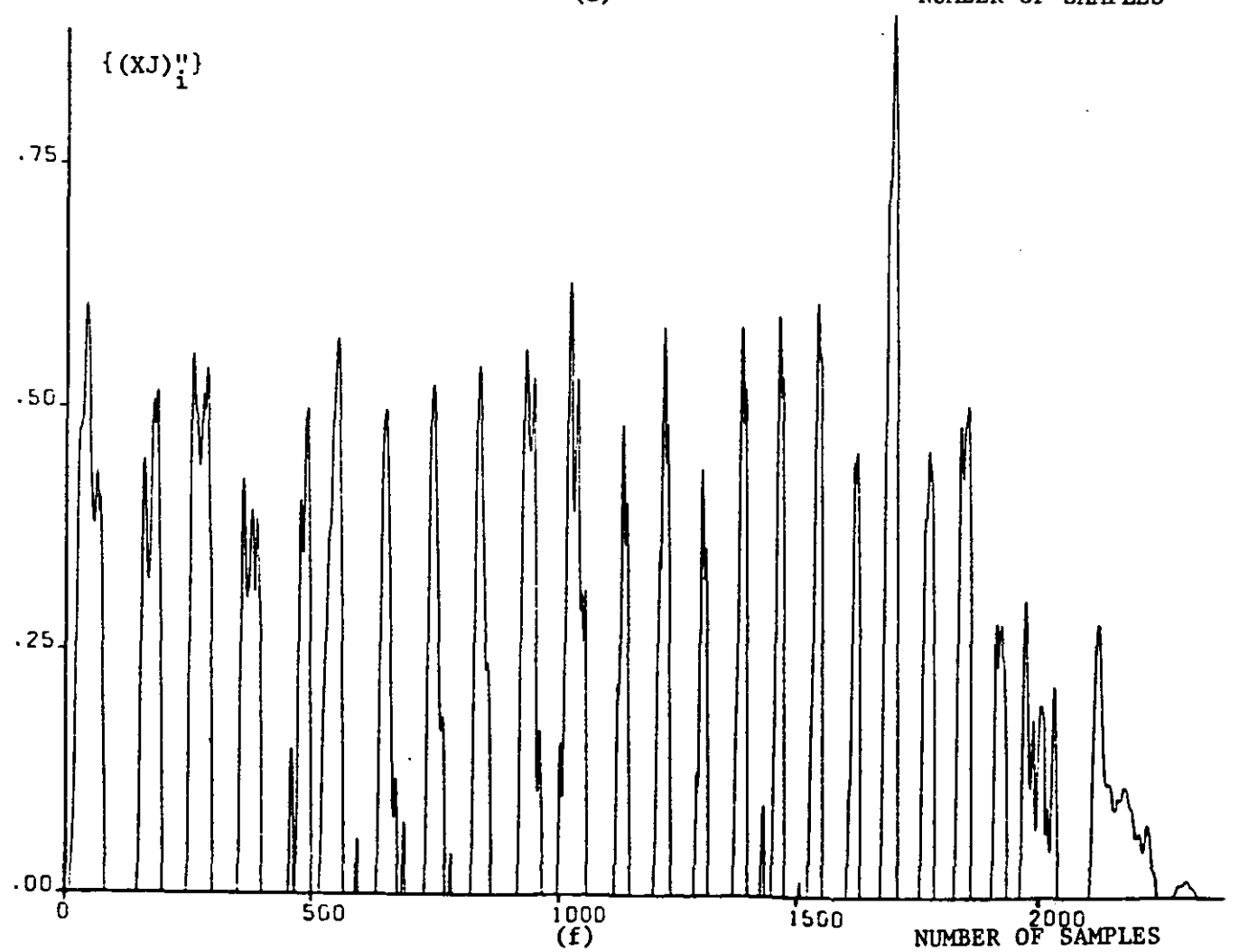
(c)


{200(XJ)i}
1.00
.80
.60
NUMBER OF SAMPLES
0
500
1000
1500
2000

(d)


x10^-1
{(XJ)'_i}
3.00
2.00
1.00
.00
0
500
1000
1500
2000
NUMBER OF SAMPLES

(e)


{(XJ)''_i}
.75
.50
.25
.00
0
500
1000
1500
2000
NUMBER OF SAMPLES

(f)

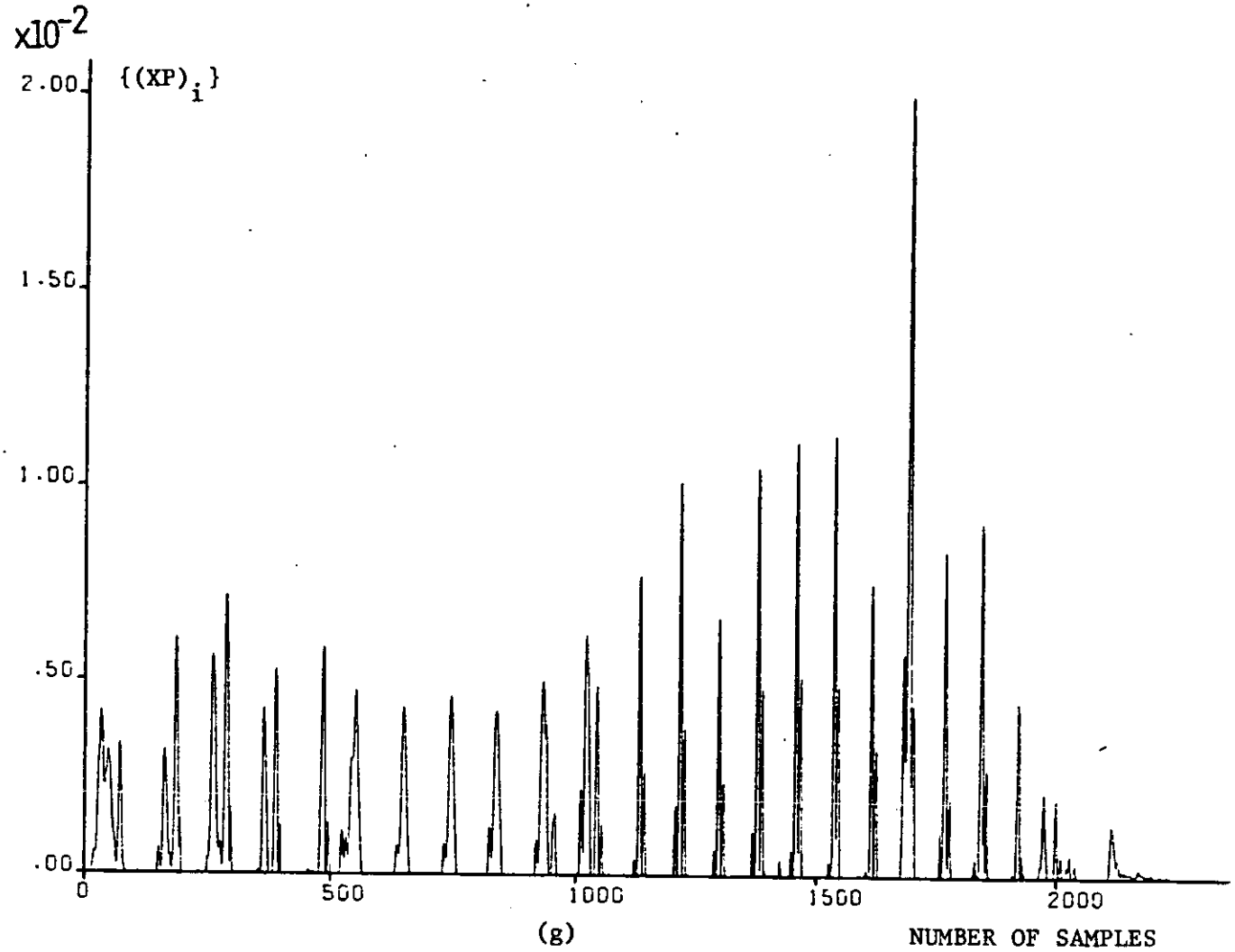


FIGURE 7.1: Waveforms for pitch extraction algorithm using SGEP

a. Segment of input speech signal
b. Variation of $\{a_{i,1}\}$ with the number of samples
c. Smoothed version of $\{a_{i,1}\}$ over the block of 25 samples
d. Smoothed version of $\{a_{i,1}\}$ over the block of 200 samples
e. Sequence of $\{(XJ)'_i\}$
f. Sequence of $\{(XJ)''_i\}$
g. Sequence of $\{(XP)_i\}$

Then the sequences represented by Equations (7.1)-(7.2) were half-rectified and multiplied to yield a sequence, $\{(XP)_i\}$. These are shown in Figure 7.1 curves (e),(f),(g) respectively.

Comparison of the input sequence, $\{X_i\}$, see Figure 7.1(a), with that of $\{(XP)_i\}$ indicates that the latter appears to have considerable potential to extract the pitch.

Further investigations should examine the performance of the proposed pitch extraction technique and compare with the existing algorithms[136] in terms of their accuracy and speed of execution on the computer.

7.5.2 Pole-Zero Predictor

It should be recalled that most of the research in LPC analysis has been focussed on all-pole models. However, the presence of unvoiced and nasal sounds suits a pole-zero model, sometimes known as auto-regressive moving average (ARMA), whose mathematical treatment is rather complicated. We feel that the modelling of vocal tract, incorporating pole-zero recursive filter, can be achieved by using a modified SGEP algorithm to update the filter coefficients, such a pole-zero filter will perhaps accurately model any possible coupling between the vocal tract and the nasal cavity.

7.5.3 Higher-Order CSP Schemes

The algorithm of the second-order switched predictor described in Chapter VI can be extended to implement a higher-order switched predictor. Further investigations should examine the effect of the second correlation coefficient of the speech signal, c_2, as well as c_1, on the evaluation of the look-up tables. We feel that an effect of c_2, in constructing the look-up tables, will also improve the performance of the previously reported codecs employing CSP scheme.

APPENDICES

APPENDIX A

QUANTIZATION BASED ON A MINIMUM MEAN-SQUARE ERROR CRITERION

The aim of this Appendix is to determine the optimum quantizer step points and levels. The input to the quantizer is assumed to be a sequence of zero mean, unit variance random variable and the mean square quantization error is minimized with respect to the quantizer step points and levels. If the variance of the input signal to the quantizer is σ_x^2 (in PCM and σ_e^2 in DPCM case) instead of one, the quantizer parameters now is obtained by multiplying those of the unit variance quantizer by a factor of σ_x (or σ_e in DPCM).

Let $P_d(x)$ be the probability density function of the input signal to the quantizer and N_Q denotes the total number of quantization levels. Output decision levels are represented by $\hat{x}_1, \hat{x}_2, \ldots, \hat{x}_{N_Q}$. It is temporarily assumed that the input to the quantizer x is a continuous variable. Let $x_1, x_2, \ldots, x_{N_Q+1}$ be the quantizer input decision levels such that, (see Figure A.1),

$$x_1 \leq x \leq x_{N_Q+1} \tag{A.1}$$

where $$x_1 < x_2 \ldots < x_{N_Q} < x_{N_Q+1}$$

and $$-x_1 = x_{N_Q+1} = \infty$$

The error power of the quantizer, σ_q^2, can be defined as[62]

$$\sigma_q^2 = \sum_{k=1}^{N_Q} \int_{x_k}^{x_{k+1}} (x-\hat{x}_k)^2 P_d(x)\,dx. \tag{A.2}$$

The necessary conditions for minimum mean-square quantization noise can be obtained by differentiating σ_q^2 with respect to the x_k's and $\hat{x}_k$'s and setting the derivatives equal to zero:

$$\frac{\partial \sigma_q^2}{\partial x_k} = \frac{\partial}{\partial x_k}\int_{x_{k-1}}^{x_k} (x-\hat{x}_{k-1})^2 P_d(x)dx + \frac{\partial}{\partial x_k}\int_{x_k}^{x_{k+1}} (x-\hat{x}_k)^2 P_d(x)dx \tag{A.3}$$

Therefore,

$$\frac{\partial \sigma_q^2}{\partial x_k} = (x_k-\hat{x}_{k-1})^2 P_d(x_k) - (x_k-\hat{x}_k)^2 P_d(x_k) = 0 \tag{A.4}$$

where $k=2,3,\ldots,N_Q$

Thus,

$$x_k = \frac{\hat{x}_{k-1} + \hat{x}_k}{2} \tag{A.5}$$

Also,

$$\frac{\partial \sigma_q^2}{\partial \hat{x}_k} = 2\int_{x_k}^{x_{k+1}} (x-\hat{x}_k) P_d(x)dx = 0 \tag{A.6}$$

therefore

$$\int_{x_k}^{x_{k+1}} (x-\hat{x}_k) P_d(x)dx = 0 \tag{A.7}$$

Equation (A.5) imposes the first condition for minimum mean-square error that x_k should lie half way between $\hat{x}_k$ and $\hat{x}_{k-1}$. Equation (A.7) shows $\hat{x}_k$ to be the centroid of the area of $P_d(x)$ between x_k and x_{k+1}. It is assumed that the probability function $P_d(x)$ is unchanged by the use of a quantizer. Equations (A.5) and (A.7) describe the overall relationships of the optimum quantizer.

One method of solving these equations is to apply a search procedure. For example, for a given number of levels, N_Q, with respect to $x_1=-\infty$ and arbitrary selection of x_2, the most negative quantizer output $\hat{x}_1$ can be obtained. If $\hat{x}_{N_Q}$ is the centroid of the area of $P_d(x)$

between x_{N_Q} and ∞, $\hat{x}_{N_Q}$ is chosen correctly. If the $\hat{x}_{N_Q}$ does not satisfy the Equation (A.4), then $\hat{x}_1$ must be reselected.

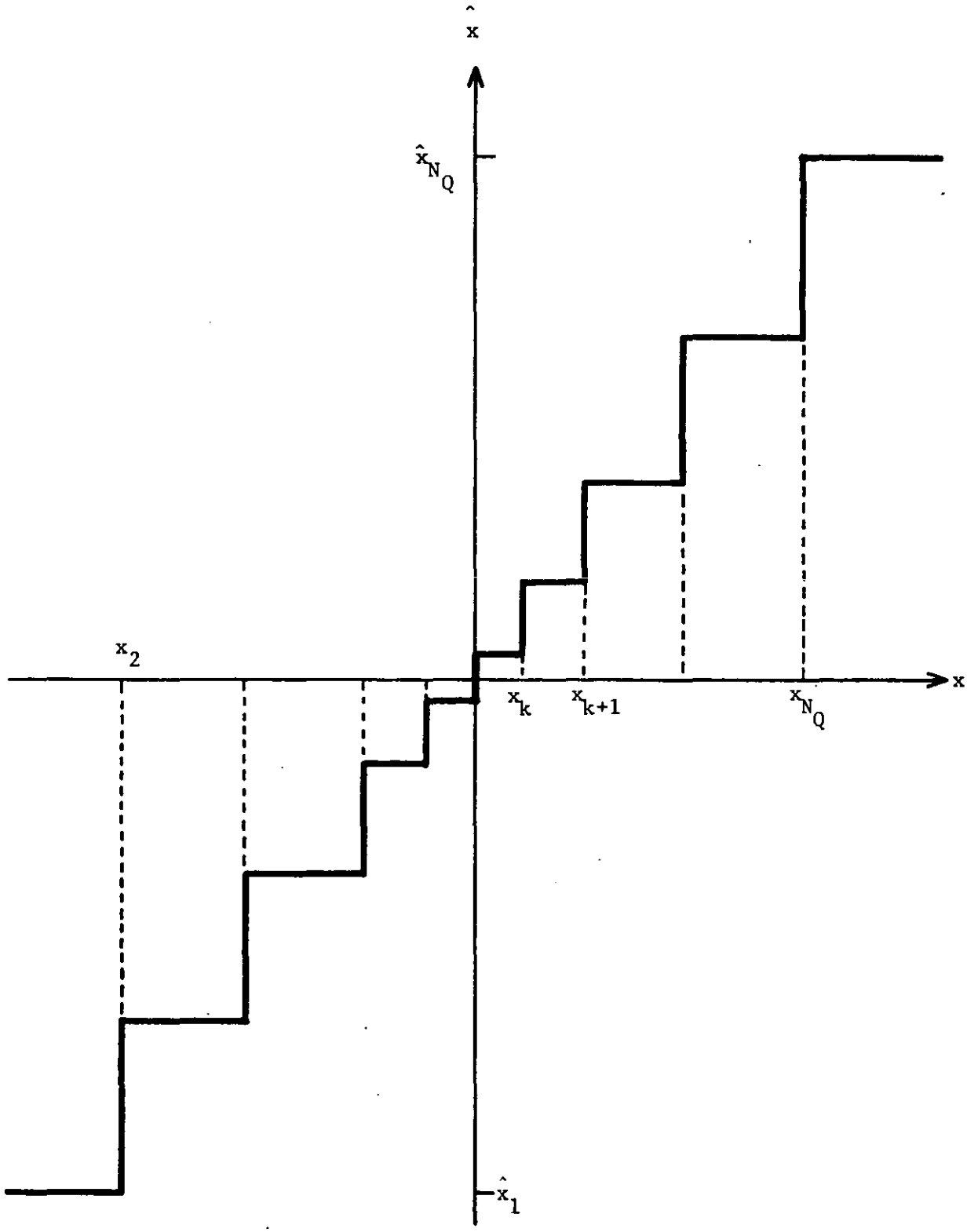


FIGURE A.1: Non-uniform quantization

APPENDIX B

QUANTIZATION NOISE POWER FOR GAUSSIAN AND LAPLACIAN P.D.F's

Zero-mean, p.d. function for *"Gaussian"* signal is given by

$$P_d(e) = \frac{1}{\sqrt{2\pi}\sigma_e} \varepsilon^{-\frac{e^2}{2\sigma_e^2}} \tag{B.1}$$

where e is the quantizer input, i.e., prediction error in DPCM and σ_e is the r.m.s. value of the error. From Equation (2.17), Panter-Dite[43] has shown that the minimum mean square quantizing error is given by

$$\sigma_q^2 = \frac{2}{3N_Q^2}\left[\int_0^V P_d^{1/3}(e)\, de\right]^3 \tag{B.2}$$

Let $\left\{\int_0^V P_d^{1/3}(e)\, de\right\}$ be I and hence

$$\sigma_q^2 = \frac{2}{3N_Q^2} I^3 \tag{B.3}$$

$$I = \int_0^V \left(\frac{1}{\sqrt{2\pi}\sigma_e}\right)^{1/3} \varepsilon^{-\frac{e^2}{6\sigma_e^2}} de \tag{B.4}$$

or

$$= \left(\frac{1}{\sqrt{2\pi}\sigma_e}\right)^{1/3} \int_0^V \varepsilon^{-\frac{e^2}{6\sigma_e^2}} de \tag{B.5}$$

Let $\frac{e}{\sqrt{3}\sigma_e}$ be t and then $de = dt\sqrt{3}\sigma_e$. Consequently, the limits of Equation (B.5) become,

$$\int_{e=0}^{e=V} \varepsilon^{-e^2/6\sigma_e^2} de = \int_{t=0}^{t=V/\sqrt{3}\sigma_e} \varepsilon^{-t^2/2} \sqrt{3}\,\sigma_e\, dt \tag{B.6}$$

From Equations (B.5) and (B.6),

$$I = \left(\frac{1}{\sqrt{2\pi}\sigma_e}\right)^{1/3} \sqrt{3}\,\sigma_e\,\sqrt{2\pi}\left\{\operatorname{erf}\left(\frac{V}{\sqrt{3}\,\sigma_e}\right) - 0.5\right\} \tag{B.7}$$

Assuming that $V/\sigma_e >> 1$ (where $\pm V$ is the overload limit of the quantizer):

$$\operatorname{erf}(V/\sqrt{3}\sigma_e) \to 1 \tag{B.8}$$

From Equations (B.7) and (B.8),

$$I \simeq \left(\frac{1}{\sqrt{2\pi}\sigma_e}\right)^{1/3} \cdot \frac{\sqrt{3}\sigma_e\sqrt{2\pi}}{2} \tag{B.9}$$

Hence, (B.9) in (B.3)

$$\sigma_q^2 \simeq \frac{2}{3N_Q^2} \cdot \frac{3\sqrt{3}\sigma_e^3}{\sqrt{2\pi}\sigma_e} \cdot \frac{2\pi\sqrt{2\pi}}{8} \tag{B.10}$$

$$\sigma_q^2 \simeq \frac{\pi\sqrt{3}}{2N_Q^2}\,\sigma_e^2$$

or

$$\sigma_q^2 \simeq \frac{2.73}{N_Q^2}\,\sigma_e^2 \tag{B.11}$$

As $Q_G = \sigma_e^2/\sigma_q^2$, i.e., SNR of the quantizer

$$Q_G \simeq \frac{N_Q^2}{2.73} \quad \text{and} \quad N_Q = 2^b \tag{B.12}$$

b is the bits/sample and then

$$\text{SNRQ(dB)} \simeq 10 \log_{10} Q_G = 6b - 4.35 \text{ dB}. \tag{B.13}$$

This is the same as Equation (2.44)

In proving Equation (B.13), it is assumed that N_Q is large, $P_d(e)$ is even function for p.d.f. of the input to the quantizer and $P_d(e)$ is zero outside the interval ($\pm V$), which represents the range of the quantizer input.

p.d.f. of Laplacian signal having symmetrical property and zero-mean is given by

$$P_d(e) = \frac{1}{\sqrt{2}\sigma_e} \varepsilon^{-\frac{\sqrt{2}}{\sigma_e}|e|} \tag{B.14}$$

In a similar fashion to that obtained for Gaussian signals,

$$I = \int_0^V P_d^{1/3}(e)de = \left(\frac{1}{\sqrt{2}\sigma_e}\right)^{1/3} \int_0^V \varepsilon^{\frac{\sqrt{2}}{3\sigma_e}|e|} de$$

For $V/\sigma_e >> 1$, it is proved that

$$\int_0^V \varepsilon^{-\frac{\sqrt{2}}{3\sigma_e}|e|} de \simeq \frac{3\sigma_e}{\sqrt{2}} \tag{B.15}$$

Therefore,

$$\sigma_q^2 \simeq \frac{2}{3N_Q^2} \cdot \frac{1}{\sqrt{2}\sigma_e} \cdot \frac{27\sigma_e^3}{2\sqrt{2}} \tag{B.16}$$

$$\sigma_q^2 \simeq \frac{9\sigma_e^2}{2N_Q^2} \tag{B.17}$$

or

$$Q_L \simeq \frac{2N_Q^2}{9} \quad , \quad N_Q = 2^b \tag{B.18}$$

where b is the bits/sample and then,

$$SNRQ(dB) = 10 \log_{10} Q_L \simeq 6b - 6.53 \tag{B.19}$$

The same assumptions in the previous case still hold for obtaining Equation (B.19).

Furthermore, from reference (60), we know that a DPCM system SNR(dB) is upper bounded by

$$SNR(dB) \lesssim 6b + SNRI \tag{B.20}$$

where $b=CI/f_s$ is the number of bits/sample.

Equation (B.20) is obtained under the assumptions of a Gaussian input, the predictor input is uncontaminated, i.e., the quantization noise in the feedback loop has been neglected and the prediction coefficients are known exactly (hence SNRI is involved in Equation (B.20)).

Comparing Equation (B.13) with Equation (B.20) reveals that DPCM is only 4.35 dB below the bound for Gaussian prediction error and Gaussian input sequence. In the case of Laplacian signals, DPCM is 6.53 dB below the bound, see Equations (B.19) and (B.20). However, it is possible to approach the bound by employing Entropy Coding (EC). For example, for Gaussian signals, 4.35 dB may be reduced to 2.35 dB by using EC.[60]

APPENDIX C

CALCULATION OF THE PREDICTION COEFFICIENTS FOR THE SECOND ORDER FIXED PREDICTORS AND THEIR RELATIONSHIP WITH THE FIRST ORDER PREDICTORS

Consider the model of Figure C.1, let us assume for simplicity, that only the two previous samples are available, i.e. x_{i-1}, x_{i-2}.

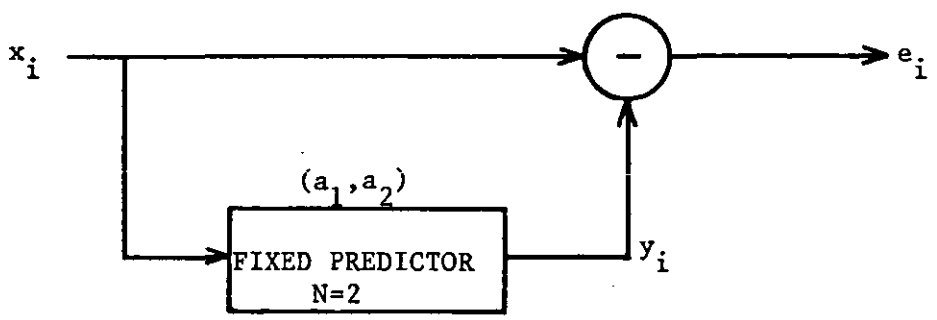


FIGURE C.1: Simple model for prediction

Then, the prediction error at ith instant is formed as

$$e_i = x_i - a_1 x_{i-1} - a_2 x_{i-2} \qquad \text{(C.1)}$$

where a_1 and a_2 are the fixed coefficients.

From Equation (C.1),

$$e_i^2 = x_i^2+a_1^2x_{i-1}^2+a_2^2x_{i-2}^2+2a_1a_2x_{i-1}x_{i-2}$$
$$-2a_1x_ix_{i-1}-2a_2x_ix_{i-2} \qquad (C.2)$$

Also, we know that,

$$\langle x_i^2\rangle \tilde{=} \langle x_{i-1}^2\rangle \tilde{=} \langle x_{i-2}^2\rangle = C_0$$
$$\langle x_ix_{i-1}\rangle \tilde{=} \langle x_{i-1}\cdot x_{i-2}\rangle = C_1$$
$$\langle x_ix_{i-2}\rangle = C_2 \qquad (C.3)$$

where C_0,C_1 and C_2 are the values of autocorrelation function of the signal at displacements of 0,1 and 2 sampling periods respectively. Thus, Equation (C.2), in mean-square error sense may be written as

$$\langle e_i^2\rangle = (1+a_1^2+a_2^2)C_0 + 2a_1(a_2-1)C_1-2a_2C_2 \qquad (C.4)$$

where $\langle(.)\rangle$ is the time average of (.). Since C_0 is the signal power and $c_r=C_r/C_0$ is the normalized rth correlation coefficient, Equation (C.4) can be expressed as,

$$\frac{\langle e_i^2\rangle}{C_0} = SNR^{-1} = (1+a_1^2+a_2^2)+2a_1(a_2-1)c_1-2a_2c_2 \qquad (C.5)$$

For

$$\frac{\partial(SNR^{-1})}{\partial a_1}\Big|_{min} = 2a_1+2(a_2-1)c_1 \to 0 \qquad (C.6)$$
$$\frac{\partial(SNR^{-1})}{\partial a_2}\Big|_{min} = 2a_2+2a_1c_1-2c_2 \to 0 \qquad (C.7)$$

and from Equations (C.6) and (C.7),

$$a_1 = \frac{c_1(1-c_2)}{1-c_1^2}\ ; \qquad a_2 = \frac{c_2-c_1^2}{1-c_1^2} \qquad (C.8)$$

Using these optimum values of a_1 and a_2, SNR^{-1} in Equation (C.5) becomes,

$$SNR^{-1} = 1-c_1^2 - \frac{(c_1^2-c_2)^2}{1-c_1^2} \qquad (C.9)$$

For the right most term to be zero, Equation (C.9) is reduced to that proved in Equation (3.38). Then

$$\frac{(c_1^2-c_2)^2}{1-c_1^2} = 0$$

and

$$c_2 = c_1^2 \qquad (C.10)$$

Equation (C.10) in Equation (C.8) results in

$$a_1 = c_1 \; ; \quad a_2 = 0 \qquad (C.11)$$

Notice that, performance of the second order predictor is equal to the first order predictor if $c_2=c_1^2$.

APPENDIX D

BAND-LIMITED LOW PASS DIGITAL FILTER

The pre- and post-encoding band-limiting operation was performed using Recursive Butterworth low pass digital filters.(137)

The gain characteristic of the Nth order Butterworth filter is given by the expression,

$$H(z) = \frac{1}{\left\{1 + \left[\frac{\tan\pi fT}{\tan\pi f_c T}\right]^{2N}\right\}^{\frac{1}{2}}} \qquad (D.1)$$

where f_c and T are the cut-off frequency and the sampling period respectively. The higher the value of N the better is the approximation of the filter's gain characteristic to an ideal low-pass characteristic. The co-ordinates of the N poles which lie in the unit circle are given by the following equations:

$$w_m = \frac{1-\tan^2\pi f_c T}{d} \qquad (D.2)$$

$$v_m = \frac{2\tan(\pi f_c T)\sin\alpha}{d} \qquad (D.3)$$

where

$$\alpha = \frac{(2m+1)\pi}{2N}, \quad m=0,1,\ldots,(2N-1) \quad \text{for N even}$$

$$\alpha = \frac{m\pi}{N}, \quad \text{for N odd}$$

$$d = 1-2\tan(\pi f_c T)\cos\alpha + \tan^2\pi f_c T$$

Hence,

$$z_m = w_m + jv_m \qquad (D.4)$$

As an example, consider the following design specifications:

Sampling frequency, f_s = 10 kHz

Cut-off frequency, f_c = 3.4 kHz

Gain at zero frequency = 0 dB

Gain between 3.8 kHz and 4 kHz $\lesssim$ -22 dB

The order of filter can be found using Equation (D.1), i.e.,

$$-22 \text{ dB} = 10 \log_{10} X$$

$$X = 1/162$$

then,

$$162 - 1 = \left[\frac{\tan\frac{\pi\times 3.8}{10}}{\tan\frac{\pi\times 3.4}{10}}\right]^{2N} \qquad (D.5)$$

$$161 = (1.3885218)^{2N}$$

$$2N = \frac{\log_{10} 161}{\log_{10} 1.3885218} = 15.48$$

$$N = 7.74 \sim 8 \qquad (D.6)$$

Using Equations (D.2) and (D.3) for N=8, i.e., m=0 to 15, we obtain the following 16 poles as shown in Table D.1 (all the zeros occur at z=-1).

The poles are in complex conjugate pairs. They are:

$z_0 - z_{15}$
$z_1 - z_{14}$
$z_2 - z_{13}$
$z_3 - z_{12}$
$z_4 - z_{11}$
$z_5 - z_{10}$
$z_6 - z_9$
$z_7 - z_8$

The locations of the poles in z-plane are shown in Figure D.1.

m	α(in rad)	d	w_m	v_m
0	0.19635	0.74065	-3.117176	0.958263
1	0.58905	1.28386	-1.798277	1.574284
2	0.98175	2.28759	-1.009243	1.322299
3	1.37444	3.59897	-0.641499	0.991416
4	1.76714	5.01847	-0.460047	0.710989
5	2.15984	6.32989	-0.364735	0.477871
6	2.55254	7.33361	-0.314815	0.275602
7	2.94524	7.87682	-0.293105	0.090104
8	3.33794	7.87682	-0.293105	-0.090104
9	3.73064	7.33361	-0.314815	-0.275602
10	4.12334	6.32989	-0.364735	-0.477871
11	4.51604	5.01847	-0.460047	-0.710989
12	4.90874	3.59897	-0.641499	-0.991416
13	5.30143	2.28759	-1.009243	-1.322299
14	5.69414	1.28386	-1.798277	-1.574284
15	6.08684	0.74065	-3.117176	-0.958263

TABLE D.1: Co-ordinates of poles

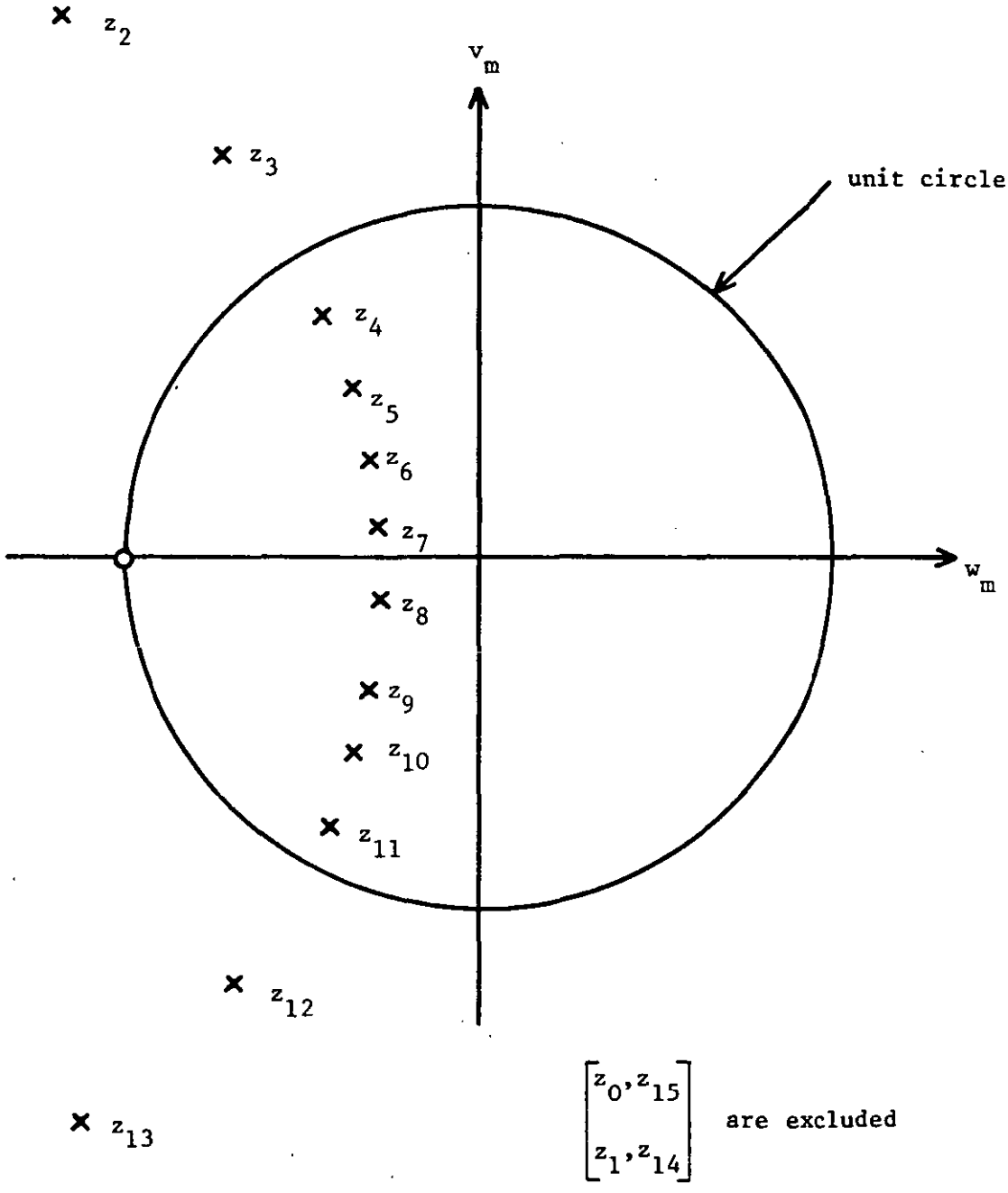


FIGURE D.1: Location of poles and zeroes on z-plane

From the above, the following conjugate pairs have $|z_m|>1$,

$$z_0 - z_{15}$$
$$z_1 - z_{14}$$
$$z_2 - z_{13}$$
$$z_3 - z_{12}$$

Since $|z_m|>1$, they lie outside the unit circle and for stability reasons we exclude them in the design procedure. Now if we let,

$$z_4 = z_A \rightarrow z_{11} = z_A^*$$
$$z_5 = z_B \rightarrow z_{10} = z_B^*$$
$$z_6 = z_C \rightarrow z_9 = z_C^*$$
$$z_7 = z_D \rightarrow z_8 = z_D^*$$

then, we have 4 pairs of poles within unit circle with the following co-ordinates:

$$z_A = -0.460047 \pm j0.710989$$
$$z_B = -0.364735 \pm j0.477871$$
$$z_C = -0.314815 \pm j0.275602$$
$$z_D = -0.293105 \pm j0.090104$$

Each pair of poles can form second order recursive filter whose transfer function is given by

$$H_1(z) = \frac{(1+z^{-1})^2}{(1-z_A z^{-1})(1-z_A^* z^{-1})} \qquad (D.7)$$

$$= \frac{1+2z^{-1}+z^{-2}}{1-(z_A+z_A^*)z^{-1}+z_A z_A^* z^{-2}} \qquad (D.8)$$

Substituting $z_A+z_A^* = -0.920094$

$z_A . z_A^* = 0.717148$

in Equation (D.8) we obtain,

$$H_1(z) = \frac{1+2z^{-1}+z^{-2}}{1+0.920094z^{-1}+0.717148z^{-2}} \qquad (D.9)$$

and its implementation is shown in Figure D.2.

Proceeding in the same way for the remaining three pairs of poles we get,

$$H_2(z) = \frac{1+2z^{-1}+z^{-2}}{1+0.729470z^{-1}+0.361392z^{-2}} \qquad (D.10)$$

$$H_3(z) = \frac{1+2z^{-1}+z^{-2}}{1+0.629631z^{-1}+0.175065z^{-2}} \qquad (D.11)$$

$$H_4(z) = \frac{1+2z^{-1}+z^{-2}}{1+0.586210z^{-1}+0.094029z^{-2}} \qquad (D.12)$$

The overall transfer function of the 8th order filter is

$$H(z) = H_1(z).H_2(z).H_3(z).H_4(z) \qquad (D.13)$$

Therefore, the complete block diagram of the filter results from the substitution of Equations (D.9)-(D.12) in Equation (D.13) and it is shown in Figure D.3. The gain factor, A0, 0.065314 arises from the unity gain at zero frequency, i.e., $\omega=0$ and hence $z=1$. Consequently

$$H_1(z) = \frac{1+2+1}{1+0.920094+0.717148}$$

$$= \frac{4}{2.6372} \quad . \qquad (D.14)$$

Similarly, from Equations (D.10)-(D.12),

$$H_2(1) = \frac{4}{2.0900} \quad , \qquad (D.15)$$

$$H_3(1) = \frac{4}{1.8046} \quad , \qquad (D.16)$$

and

$$H_4(1) = \frac{4}{1.6802} \quad , \qquad (D.17)$$

In order to obtain unity gain at zero frequency, we want

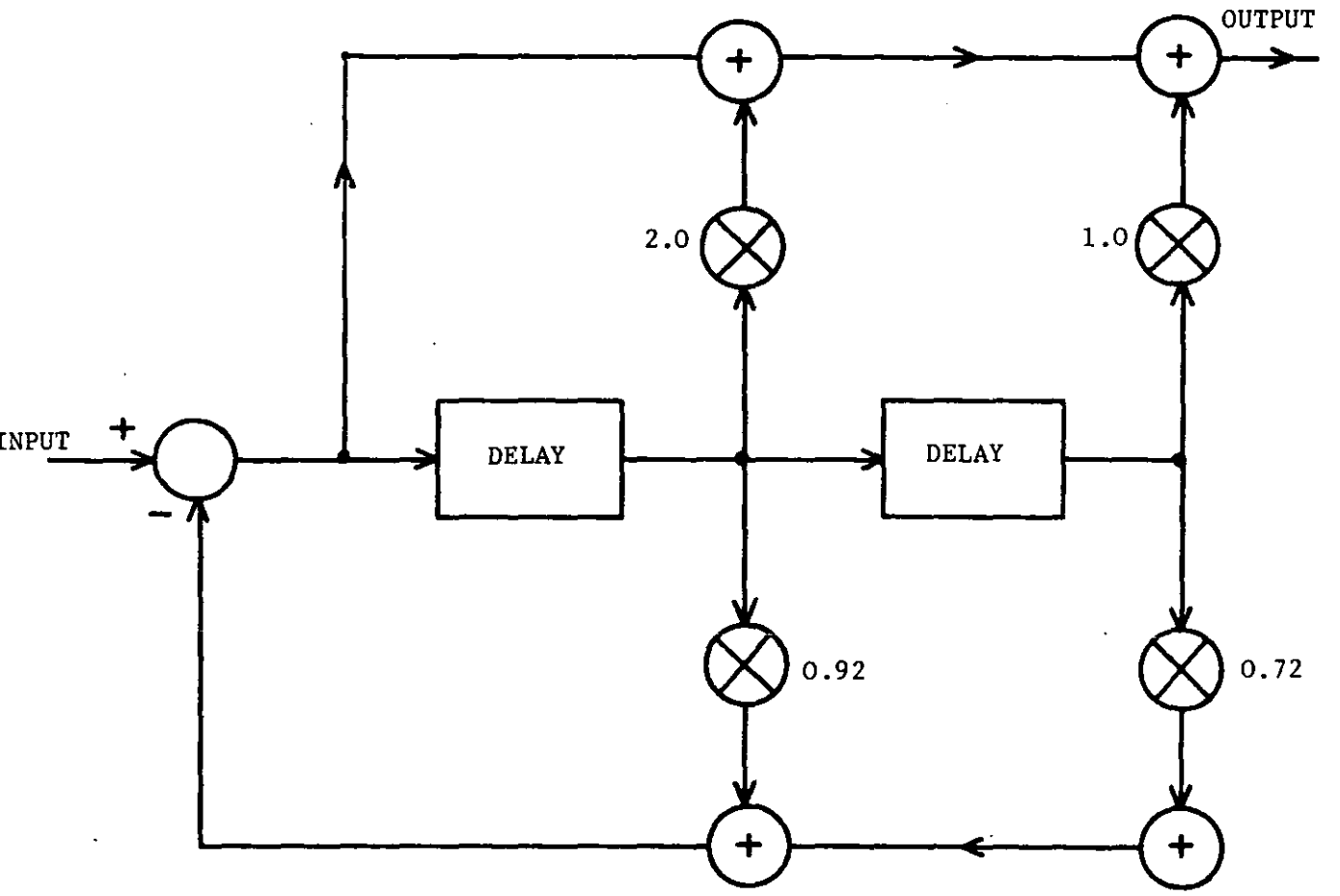


FIGURE D.2: The 2nd-order recursive filter

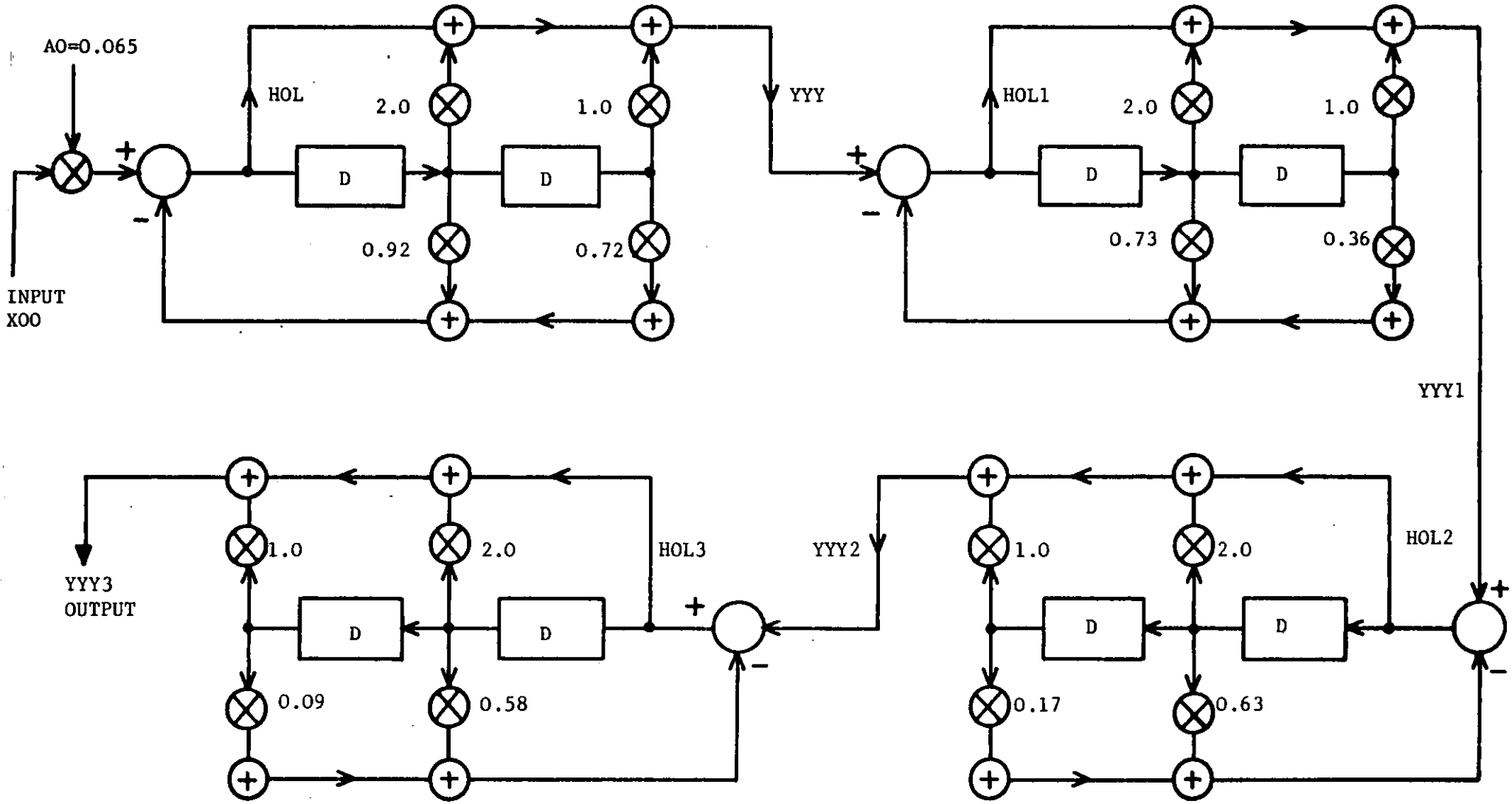


FIGURE D.3: Block diagram of 8th-order Butterworth digital filter

$$AO\{H_1(1).H_2(1).H_3(1).H_4(1)\} = 1 \qquad (D.18)$$

and then $AO\{256/16.72\} = 1$

$AO = 0.065314$

where AO is the gain factor for unity gain at zero frequency.

The subroutine of a filter is shown in LIST 1 (sampling frequency, f_s=10 kHz and cut-off frequency, f_c=3.4 kHz) where XOO is an input sample, YYY3 is output sample and A is real coefficient array. LIST 2 presents also the subroutine of a filter,but for f_s=8 kHz and f_c=3.4 kHz.

List 3 presents a more general programme for filter design where

CC = Cut-off frequency in kHz,

CCC = Sampling frequency in kHz,

and the remaining parameters are explained in the subroutines.

LIST 1

COMPUTER SIMULATION OF 8TH-ORDER BUTTERWORTH DIGITAL FILTER, f_s=10 kHz

f_c=3.4 kHz

SUBROUTINE FILTER (XXO,YYY3,A)

```
DIMENSION A(8)
XOO=XOO*0.065314
HOL=XOO-(0.920094*A(1)+0.717148*A(2))
YYY=HOL+2.0*A(1)+1.0*A(2)
A(2)=A(1)
A(1)=HOL
HOL1=YYY-(0.729470*A(3)+0.361392*A(4))
YYY1=HOL1+2.0*A(3)+1.0*A(4)
A(4)=A(3)
A(3)=HOL1
HOL2=YYY1-(0.629631*A(5)+0.175065*A(6))
YYY2=HOL2+2.0*A(5)+1.0*A(6)
A(6)=A(5)
A(5)=HOL2
HOL3=YYY3-(0.586210*A(7)+0.094028*A(8))
YYY3=HOL3+2.0*A(7)+1.0*A(8)
A(8)=A(7)
A(7)=HOL3
RETURN
END
```

LIST 2

COMPUTER SIMULATION OF 8TH-ORDER BUTTERWORTH DIGITAL FILTER, f_s=8 kHz

f_c=3.4 kHz

SUBROUTINE FILTER (XOO,YYY3,A)

```
DIMENSION A(8)
XOO=XOO*0.294497
HOL=XOO-(1.63702*A(1)+0.837274*A(2))
YYY=HOL+2.0*A(1)+1.0*A(2)
A(2)=A(1)
A(1)=HOL
HOL1=YYY-(1.42308*A(3)+0.597149*A(4))
YYY1+HOL1+2.0*A(3)+1.0*A(4)
A(4)=A(3)
A(3)=HOL1
HOL2=YYY1-(1.29368*A(5)+0.451927*A(6))
YYY2=HOL2+2.0*A(5)+1.0*A(6)
A(6)=A(5)
A(5)=HOL2
HOL3=YYY2-(1.23300*A(7)+0.383827*A(8))
YYY3=HOL3+2.0*A(7)+1.0*A(8)
A(8)=A(7)
A(7)=HOL3
RETURN
END
```

LIST 3

```
CCE1   DATE 27 . 01 . 1982   TIME 08:57


      DIMENSION A1(4), A2(4), B1(4), B2(4)
      COMPLEX ALPHA(30), BETA(30)
      WRITE(1,20)
20    FORMAT(' F.C.. SAMPL. RATE ')
      READ(1,*)CC,CCC
C     CC  = CUTOFF FREQUENCY
C     CCC = SAMPLING RATE
      FC=CC/CCC
      CALL BUTTER(8,FC,ALPHA,BETA)
      CALL COEFF(4,ALPHA,BETA,A1,A2,B1,B2,A0)
      WRITE(1,10)A1,A2,B1,B2
10    FORMAT(4F12.8)
      WRITE(1,15)A0
15    FORMAT(5X,E8.3)
      CALL EXIT
      END
$INSERT  BUTTER
      SUBROUTINE BUTTER(N,FC,ALPHA,BETA)
C
C     THIS SUBROUTINE COMPUTES THE POLES AND ZEROES OF A
C     BUTTERWORTH LOWPASS DIGITAL FILTER.
C
C     INPUTS ARE:  N = ORDER OF FILTER
C                 FC = CUTOFF FREQUENCY AS A FRACTION OF
C                      THE CLOCK FREQUENCY
C     OUTPUTS ARE: ALPHA = COMPLEX ARRAY CONTAINING THE
C                          TRANSFER FUNCTION ZEROES IN
C                          ITS FIRST N LOCATIONS
C
C                  BETA = COMPLEX ARRAY CONTAINING THE
C                         TRANSFER FUNCTION POLES IN ITS
C                         FIRST N LOCATIONS
C

      COMPLEX ALPHA(30), BETA(30)
      WC=3.141592654*FC
      TAN2=2.0*SIN(WC)/COS(WC)
      TANSQ=0.25*TAN2**2
      IF(N.EQ.1)GOTO 2
      IN=MOD(N,2)
      N1=N+IN
      N2=(3*N+IN)/2-1
      DO 1 M=N1,N2
      A=3.141592654*FLOAT(2*M+1-IN)/FLOAT(2*N)
      ANUM=1.0-TAN2*COS(A)+TANSQ
      U=(1.0-TANSQ)/ANUM
      V=TAN2*SIN(A)/ANUM
      I=(N2-M)*2+1
      J1=I+IN
      J2=I+IN+1
```

CCE1 (CONT.)

```
      BETA(J1)=CMPLX(U,V)
1     BETA(J2)=CMPLX(U,-V)
      IF(IN)3,3,2
2     BETA(1)=CMPLX(((1.0-TANSQ)/(1.0+TAN2+TANSQ)),0.0)
3     DO 4 I=1,N
4     ALPHA(I)=(-1.0,0.0)
      N1=N+1
      DO 5 I=N1,30
      ALPHA(I)=(0.0,0.0)
5     BETA(I)=(0.0,0.0)
      RETURN
      END

      SUBROUTINE COEFF(N,ALPHA,BETA,A1,A2,B1,B2,A0)
C
C     THIS SUBROUTINE COMPUTES THE COEFFICIENTS IN A SERIAL
C     FORM REALISATION OF A DIGITAL FILTER.
C
C     INPUTS ARE:      N = NUMBER OF SECTIONS IN FILTER
C                  ALPHA = ARRAY HOLDING FILTER ZEROES
C                   BETA = ARRAY HOLDING FILTER POLES
C
C     OUTPUTS ARE: A1,A2,B1,B2: ARRAYS HOLDING SECTION
C                               COEFFICIENTS
C                         A0: GAIN COEFFICIENT FOR UNITY GAIN
C                             AT ZERO FREQUENCY
C
      COMPLEX ALPHA(30),BETA(30)
      DIMENSION A1(15),A2(15),B1(15),B2(15)
      A0=1.0
      DO 1 I=1,N
      I1=2*I-1
      I2=2*I
      A1(I)=REAL(-ALPHA(I1)-ALPHA(I2))
      A2(I)=REAL(ALPHA(I1)*ALPHA(I2))
      B1(I)=REAL(-BETA(I1)-BETA(I2))
      B2(I)=REAL(BETA(I1)*BETA(I2))
1     A0=A0*(1.0+A1(I)+A2(I))/(1.0+B1(I)+B2(I))
      IF(ABS(A0).LT.(1.0E-6))A0=1.0
      A0=1.0/A0
      RETURN
      END
```

APPENDIX E

CUMMISKEY'S SEQUENTIAL ALGORITHM[62]

The generalized equation for the adaptation of the kth prediction coefficient of a sequential prediction at the (i+1)th sampling instant is given as:

$$a_{i+1,k} = a_{i,k} - g \cdot \frac{\partial(FU)}{\partial a_{i,k}} \tag{E.1}$$

If the error function to be minimized is the absolute error, i.e., $FU=|\langle e_i \rangle|$ and since

$$FU = |\langle e_i \rangle| = \langle e_i \ \mathrm{sgn}(e_i) \rangle \tag{E.2}$$

then the kth component of the gradient is given by

$$\frac{\partial(FU)}{\partial a_{i,k}} = \frac{\partial \langle (x_i - \sum_{k=1}^{N} a_k x_{i-k}) \mathrm{sgn}(e_i) \rangle}{\partial a_{i,k}} \tag{E.3}$$

or

$$\simeq -(x_{i-k})\mathrm{sgn}(e_i) \tag{E.4}$$

The approximation sign ($\simeq$) comes in since, to differentiate Equation (E.3), $|\langle e_i \rangle|$ has been approximated to the absolute value of the sample error, $|e_i|$. Equation (E.4) gives the derivative of the absolute error with respect to kth coefficient. The substitution of Equation (E.4) in Equation (E.1), together with g=h yields

$$a_{i+1,k} = a_{i,k} + h \ x_{i-k} \ \mathrm{sgn}(e_i) \tag{E.5}$$

In terms of vectoral quantity, the total gradient vector, (FU) is written as

$$\nabla(FU) \simeq - \hat{X}_i \ \mathrm{sgn}(e_i) \tag{E.6}$$

and therefore the general form of sequential prediction becomes

$$\hat{A}_{i+1,k} = \hat{A}_i + h\hat{X}_i \, \mathrm{sgn}(e_i) \tag{E.7}$$

where $\hat{X}_i = [x_{i-1}, x_{i-2}, \ldots, x_{i-N}]^T$

and

$k = 1,2,\ldots,N$

h is constant and controls the rate of adaptation.

As it was discussed in Chapter IV, see Section 4.4, h is expressed as

$$h = \frac{A}{B + \zeta(x_i, M)} \tag{E.8}$$

where A,B are constants and $\zeta(x_i, M)$ is a function of the M previous speech samples, that is

$$\zeta(x_i, M) = \sum_{k=1}^{M} |x_{i-k}| \tag{E.9}$$

When SAP algorithm, minimizing $\langle e_i^2 \rangle$ is employed in DPCM and compared with that of Cummiskey's sequential algorithm defined by Equations (E.1)-(E.9), the following conclusions can be drawn:

a. Absolute error criterion reduces hardware complexity, and computationally, for N-order predictor, it requires (2N+3) multiplications and 3N additions per sample.

b. In terms of SNR, SAP with $\langle e_i^2 \rangle$ slightly out-performs Cummiskey's relationship.

c. In the presence of transmission errors Equation (E.7) is less affected than the SAP algorithm, since the former one uses the polarity of e_i rather than the actual amplitude of e_i as in the SAP algorithm.(60)

APPENDIX F

LATTICE PREDICTOR AND PARCOR COEFFICIENTS

This type of predictor, rather than employing usual transversal filter structure, has a different form. Its coefficients are computed sequentially. Itakura-Saito[32] studied in detail and the structure of this sort has been used in vocoder type systems. The beauty of lattice predictors is that rather than calculating the predictor coefficients, PARtial CORrelation - PARCOR - or reflection coefficients are computed. These coefficients are less sensitive for the transmission purposes. Such a predictor in ADPCM loop is shown in Figure F.1.

In Figure F.1, the redundancies of the speech signal are removed stage by stage-cascaded form. The coefficient $b_{i,k}$, where k=1,2,...,N, at kth stage,at ith sampling instant,is optimized to minimize the error term $e_{i,k+1}$ and in this way, the final output to the quantizer, $e_{i,N+1}$ is minimum. The error sample, $e_{i,N+1}$ at ith sampling instant is given by,

$$e_{i,N+1} = x_i - \sum_{k=1}^{N} b_{i,k}\hat{F}_{i,k} \qquad (F.1)$$

and consequently the final output $y_{i,N+1}=y_i$ of a lattice predictor is

$$y_i = y_{i,N+1} = \sum_{k=1}^{N} b_{i,k}\hat{F}_{i,k} \qquad (F.2)$$

where $\hat{F}_{i,k}$'s (information that is available to the receiver), are shown in Figure F.1. The energy in the intermediate prediction error, $e_{i,k+1}$, at the output of each filter stage is to be minimized individually. Hence,

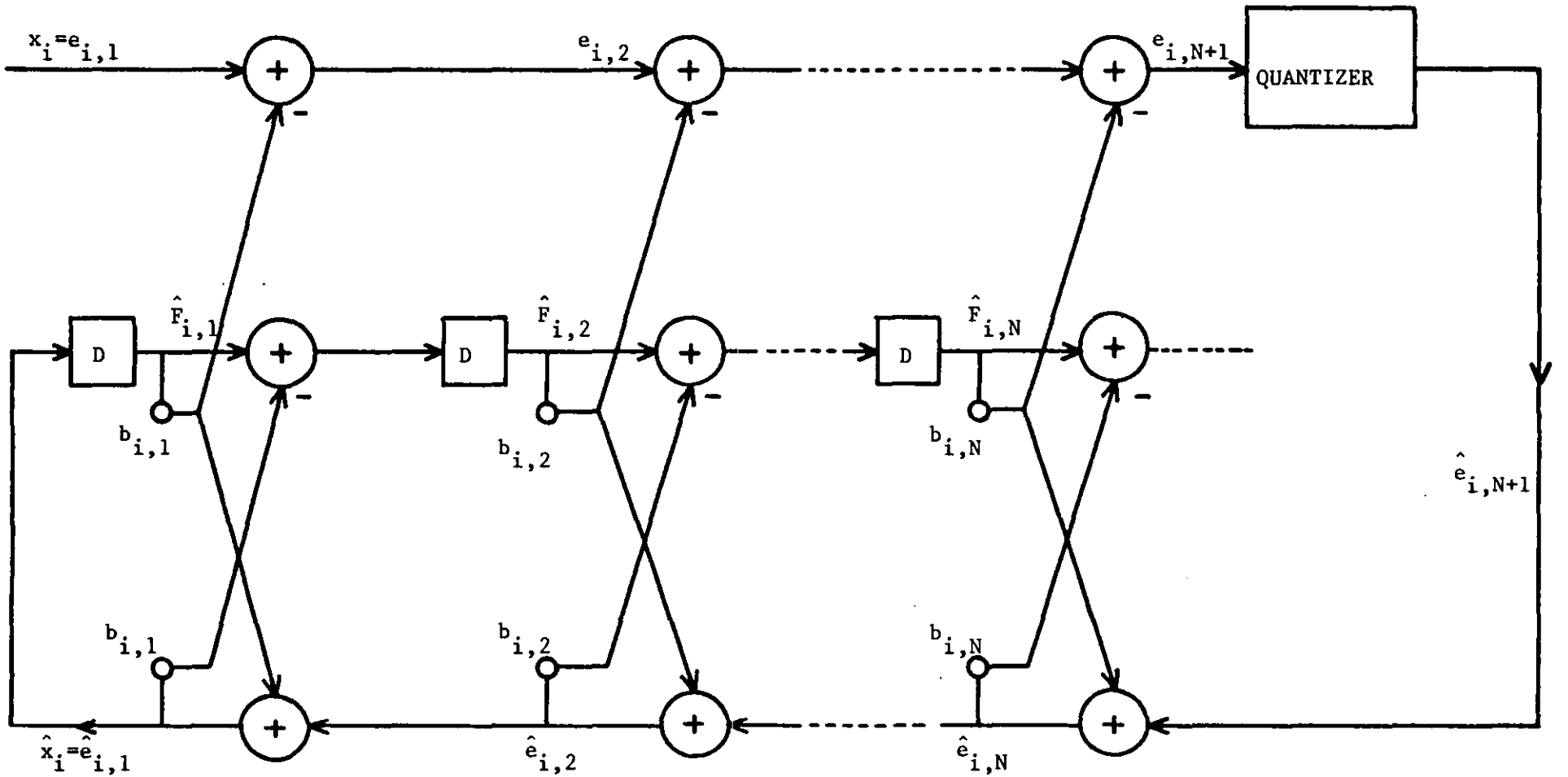


FIGURE F.1: Lattice predictor structure in a differential encoder

$$e_{i,k+1} = e_{i,k} - b_{i,k} \hat{F}_{i,k}, \quad k=1,2,\ldots,N \tag{F.3}$$

Given the initial states of $\hat{F}_{i,k}$'s, new values are found by the following relationships:

$$\hat{e}_{i,k} = \hat{e}_{i,k+1} + b_{i,k} \hat{F}_{i,k} \tag{F.4}$$

$$\hat{F}_{i,k+1} = D(\hat{F}_{i,k} - b_{i,k}\hat{e}_{i,k}) \neq D\hat{F}_{i,k} \tag{F.5}$$

and $$\hat{F}_{i,1} = D(\hat{e}_{i,1}), \text{ where } D = \text{Delay}.$$

This method eliminates the inversion of matrix and the coefficients approach to the optimum values faster than those of autocorrelation methods. Further PARCOR technique has greater numerical accuracy in computation and stability checking criterion ($|b_{i,k}|<1$) is easy. However, there are large numbers of computations involved. This is because, N multiplications for each output sample of direct methods, now are replaced by 2N multiplications. The stability test mentioned above is widely used by employing the relationships between the direct method coefficients $a_{i,k}$'s and PARCOR coefficients, $b_{i,k}$'s as studied in reference (26). For this reason, initially $a_{i,k}$ coefficients of predictor are converted into the coefficients of inverse filter, $\tilde{a}_{i,k}$'s, (input speech samples, applied to the inverse predictor, directly produces prediction error samples). For N-order predictor, at ith sampling instant, $\tilde{a}_k$'s are computed as follows:

$$\tilde{a}_1 = 1.0$$

$$\tilde{a}_k = -a_{k-1}, \quad \text{where } 2 \leq k \leq N+1. \tag{F.6}$$

Then recursion formula from reference (26), see page 95, can be employed, i.e.,

$$b_k = \tilde{a}_{k,k}$$

$$\tilde{a}_{j,k-1} = \frac{\tilde{a}_{j,k} - \tilde{a}_{k,k}\tilde{a}_{k-j}}{1-b_k^2} \qquad 1 \leq j \leq k-1 \tag{F.7}$$

where k goes from N to N-1, down to 1. It is necessary and sufficient that parameters, b_k's at ith instant, must satisfy the condition,

$$-1 \leq b_k \leq 1. \tag{F.8}$$

Thus if any of the b_k violate Equation (F.8), then it is known that there are roots outside the unit circle, i.e., predictor is unstable. At this stage we can attempt to correct the stability in the following manner: After each b_k is calculated, its absolute value is compared to 1. If $|b_k| \leq 1.0$, the algorithm proceeds to calculate the next coefficients. If $|b_k| > 1$, coefficients are restricted to $\text{sgn}(0.98, b_k)$, then we can use inverse recursion formula in order to obtain predictor coefficients,[26] i.e.,

$$\tilde{a}_{k,k} = b_k$$

$$\tilde{a}_{j,k} = \tilde{a}_{j,k-1} - b_k\tilde{a}_{k-j,k-1} \tag{F.9}$$

where $1 \leq j \leq k-1$ and (F.9) is being solved for k=1,2,...,N and with the final value of a_k's being defined by Equation (F.6). All the procedure described here is defined by the following subroutines (LIST 4).

LIST 4

```
CCE   DATE 27 . 01 . 1982   TIME 09:06


C
C
C       MAIN PROGRAM
C       N=ORDER OF THE PREDICTOR
C       AR(J)=ARRAY FOR INVERSE PREDICTOR COEFFICIENTS
C       A(J)= ARRAY FOR PREDICTOR COEFFICIENTS
C       RC(J)=ARRAY FOR PARCOR COEFFICIENTS
C       AA(J)=ARRAY FOR PREDICTOR COEFFICIENTS AFTER
C             STABILITY CORRECTION PROCESS
C
C
C
      AR(1)=1.00
      DO 1 J=2,N+1
      AR(J)=-A(J-1)
1     CONTINUE
      CALL REFLEC(AR,RC,N)
      CALL REFLECT(RC,ARR,N)
      DO 2 J=1,N
      AA(J)=-ARR(J+1)
2     CONTINUE
      CALL EXIT
      END
C
C
C       SUBROUTINE REFLEC CONVERTS AR(J)'S INTO RC(J)'S
C       INVERSE PREDICTOR COEFFICIENTS INTO PARCOR COEF.
C
C
      SUBROUTINE REFLEC(AR,RC,N)
      DIMENSION AR(10),RC(10),BX(20)
      N=2
      MP1=N+1
      ALPHA=1.0
      DO 30 J6=1,N
      MR=N+1-J6
      MRP1=MR+1
      D=1.0-AR(MPR1)*AR(MPR1)
      ALPHA=ALPHA/D
      DO 10 K=1,MR
      MM=MR+2-K
10    BX(K)=AR(MM)
      DO 20 K=1,MR
20    AR(K)=(AR(K)-AR(MRP1)*BX(K))/D
      RC(MR)=AR(MPR1)
30    CONTINUE
      DO 40 IK=1,N
40    IF(ABS(RC(IK)).GE.1.0) RC(IK)=SIGN(0.9888,RC(IK))
      RETURN
      END
```

CCE (CONT.)

```
C
C
C       SUBROUTINE REFLECT CONVERTS RC(J)'S INTO ARR(J)'S
C       PARCOR COEFFICIENTS INTO CORRECTED INVERSE FILTER COEF.
C
C
      SUBROUTINE REFLECT(RC,ARR,N)
      DIMENSION RC(10),ARR(10),BR(20)
      ARR(1)=1.00
      ARR(2)=RC(1)
      DO 30 MINC=2,N
      DO 10 J20=1,MINC
      JB=MINC-J20+1
10    BR(J20)=ARR(JB)
      DO 20 IP=2,MINC
20    ARR(IP)=ARR(IP)+RC(MINC)*BR(IP-1)
      ARR(MINC+1)=RC(MINC)
30    CONTINUE
      RETURN
      END
```

APPENDIX G

DURBIN'S SEQUENTIAL ALGORITHM

For the autocorrelation method, the matrix equation for obtaining the predictor coefficients (see Equation (3.17)) is of the form,

$$C_r = \sum_{k=1}^{N} a_k C_{rk} \quad , \quad 1 \leq k \leq N$$

where C_r is the rth autocorrelation function.

The most efficient method for solving Toeplitz autocorrelation matrix equations is Durbin's recursive procedure[11,28] which can be stated as follows:

$$E_0 = C_0 \tag{G.1}$$

$$b_k = [C_k - \sum_{j=1}^{k-1} a_{j,(k-1)} C_{k-j}]/E_{k-1} \tag{G.2}$$

where $1 \leq k \leq N$

$$a_{k,k} = b_k \tag{G.3}$$

$$a_{j,k} = a_{j,(k-1)} - b_k a_{(k-j),(k-1)} \tag{G.4}$$

where $1 \leq j \leq k-1$

$$E_k = (1-b_k^2)E_{k-1} \quad . \tag{G.5}$$

Once Equations (G.1)-(G.5) are solved recursively for k=1,2,...,N and the final solution is given as

$$a_j = a_{j,N} \quad , \quad 1 \leq j \leq N \tag{G.6}$$

where $a_{j,N}$ is the jth predictor coefficient for a predictor of order N.

In order to illustrate the above procedure, consider an example of obtaining the predictor coefficients for a predictor of order 3. The original matrix equation is defined as,

$$\begin{pmatrix} C_0 & C_1 & C_2 \\ C_1 & C_0 & C_1 \\ C_2 & C_1 & C_0 \end{pmatrix} \begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix} = \begin{pmatrix} C_1 \\ C_2 \\ C_3 \end{pmatrix} \qquad (G.7)$$

Using Equations (G.1)-(G.6) we obtain:

STEP 1: $E_0 = C_0$

(i) $b_1 = C_1/E_0 = C_1/C_0$

(ii) $a_{1,1} = b_1$

(iii) $E_1 = (1-b_1^2)E_0 = \dfrac{C_0^2-C_1^2}{C_0}$.

STEP 2: (i) $b_2 = [C_2-a_{1,1}C_1]/E_1$

$$= \frac{C_2C_0-C_1^2}{C_0^2-C_1^2}$$

(ii) $a_{2,2} = b_2$

$a_{1,2} = a_{1,1}-b_2a_{1,1}$

$$= \frac{C_0C_1-C_1C_2}{C_0^2-C_1^2}$$

(iii) $E_2 = (1-b_2^2)E_1$.

STEP 3: (i) $b_3 = [C_3-a_{1,2}C_2-a_{2,2}C_1]/E_2$

(ii) $a_{3,3} = b_3$

$a_{1,3} = a_{1,2}-b_3a_{2,2}$

$a_{2,3} = a_{2,2}-b_3a_{1,2}$

(iii) $E_3 = (1-b_3^2)E_2$.

Hence, the 3rd-order predictor coefficients are defined as

$$[a_1, a_2, a_3] = [a_{1,3}, a_{2,3}, a_{3,3}] .$$

The number of multiplications/additions required per block size of W, for obtaining N prediction coefficients are also considered. Assuming that the division is computationally equivalent to two multiplications, the whole process can be summarized as shown in Table G.1.

FUNCTION	MULTIPLICATIONS	ADDITIONS
$C_0 = \sum_{i=1}^{W} x_i^2$	W	W-1
C_1	W-1	W-2
C_2	W-2	W-3
C_N	W-N	W-(N+1)
TOTAL 1	$\frac{1}{2}(N+1)(2W-N)$	$\frac{1}{2}(N+1)[2W-(N+2)]$
From the above example for N=3:		
STEP 1: (i)	2	0
(ii)	0	0
(iii)	3	1
STEP 2: (i)	3	1
(ii)	1	1
(iii)	3	1
STEP 3: (i)	4	2
(ii)	2	2
(iii)	3	1
STEP N: (i)	N+1	N-1
(ii)	N-1	N-1
(iii)	3	1
TOTAL 2	N^2+4N	N^2
TOTAL 1 + TOTAL 2	$\frac{1}{2}[N^2+(7+2W)N+2W]$	$\frac{1}{2}[N^2+(2W-3)N+2W-2]$

TABLE G.1: Computational requirement per block of W samples for calculation of N prediction coefficients from Durbin's algorithm

```
THESIS1   DATE 15 . 03 . 1982   TIME 16:05


C
C
C      PROGRAMS FOR
C                    DPCM-AQF-FFOP
C                    DPCM-AQF-FSOP
C                    FBADPCM-AQF
C                    ADPCM-AQF-CSP(4)-FSOP
C
C
       IMPLICIT INTEGER*4(I-N),REAL*8(A-H,O-Z)
       DIMENSION X(256),C(5),CD(5),A(8),B(20),SG1(10),A1(5)
       DIMENSION SNR1(200),DBS(200),KK(10),CT(200)
C
C      B VALUES CORRESPOND TO POWER LEVELS OF INPUT SPEECH SIGNAL
C
       B(1)=200.0
       B(2)=250.0
C
C      IM VALUES CORRESPOND TO DIFFERENT ENCODERS IM=1 IS FOR FIXED
C      SECOND ORDER PREDICTOR,IM=2 IS FOR FORWARD BLOCK ADAPTIVE
C      PREDICTOR WHILE IM=3 IS FOR SWITCHED PREDICTOR
C
C       SYSTEM PARAMETERS ARE AS FOLLOWS:
C       AQ1=QUANTIZER OPTIMIZING COEFFICIENT
C       AQ1 IS 1.0,0.50,0.33 FOR 2,3,4 BITS/SAMPLE QUANTIZATION
C       AQ2=MAXIMUM QUANTIZER INPUT RANGE CONSTANT
C       AQ2 IS 1.0,3.0 AND 7.0 FOR 2,3 AND 4 BITS/SAMPLE QUANTIZATION
C       L1 AND L2 CORRESPOND TO POWER LEVEL
C       NB IS THE TOTAL NUMBER OF BLOCKS EACH HAVING 256 SAMPLES
C       LK1,LK2 ARE THE VALUES OF IM,VARIES BETWEEN 1 AND 3
C       IM=FOR FSOP,IM=2 FOR FBA ,IM=3 FOR SWITCHED FIXED,CSP(4)-FSOP
C
       READ(1,*)AQ1,AQ2,L1,L2,NB,LK1,LK2
       DO 26 IM=LK1,LK2
       DO 25 K=L1,L2
       DO 1 J=1,8
1      A(J)=0.00
       CB=0.00
       ERB=0.00
       XO=0.000
       DO 21 INC=1,NB
       READ(5,*)(X(J),J=1,256)
       C1=0.0
       ERR1=0.0
       KR=1
       KRR=256
       CALL CORRE(X,KR,KRR,C,CT,INC)
       CALL DURBIN(C,A1)
       AUTO=CT(INC)
       KRR=KRR+3
```

THESIS1 (CONT.)

```
      YM=0.0
      DO 3 IK=1,256
      IF(IK-1)3,3,2
2     RR1=X(IK)-0.864*X(IK-1)
      RR=RR1/B(K)
      YM=YM+RR*RR
3     CONTINUE
      YM=YM/256.0
      YM=DSQRT(YM)
      Z12=A01*YM
      CALL TRANS(Z12,ST,KK)
      Z2=ST
      Z1=A02*ST
      IF(IM-2)4,13,5
4     A1(1)=1.515
      A1(2)=-0.752
      GO TO 13
5     IF(AUTO.GT.0.7000) GO TO 6
      GO TO 7
6     A1(1)=1.523
      A1(2)=-0.720
      GO TO 13
7     IF(AUTO.LE.0.700.AND.AUTO.GT.0.400) GO TO 8
      GO TO 9
8     A1(1)=0.950
      A1(2)=-0.3450
      GO TO 13
9     IF(AUTO.LE.0.400.AND.AUTO.GT.0.000) GO TO 10
      GO TO 11
10    A1(1)=0.4930
      A1(2)=-0.210
      GO TO 13
11    IF(AUTO.LE.0.000) GO TO 12
      GO TO 13
12    A1(1)=-0.6311
      A1(2)=-0.3622
13    DO 20 J=KR,KRR
      Y=X(J)
      Y=Y/B(K)
      N22=2
      NC=N22-1
C
C     FOR FFOP THE FOLLOWING CONDITIONS ARE REPLACED BY
C     IF(INC-1)14,14,14 AND IF(J-N22)15,15,15
C
      IF(INC-1)14,14,17
14    IF(J-N22)15,15,17
15    YY=Y-0.864*X0
      CALL Q(YY,Y1,Z1,Z2,INDIC)
      X0=0.864*X0+Y1
```

THESIS1 (CONT.)

```
      XOO=Y-XO
      CALL FILT(XOO,Y5,A)
      C1=C1+Y*Y
      ERR1=ERR1+Y5*Y5
      CB=CB+Y*Y
      ERB=ERB+Y5*Y5
      DO 16 I1=1,NC
      SG1(N22+1-I1)=SG1(N22-I1)
16    SG1(1)=XO
      GO TO 20
17    EST=0.00
      DO 18 I1=1,N22
18    EST=EST+SG1(I1)*A1(I1)
      YY=Y-EST
      CALL Q(YY,Y1,Z1,Z2,INDIC)
      XO=EST+Y1
      XOO=Y-XO
      CALL FILT(XOO,Y5,A)
      C1=C1+Y*Y
      ERR1=ERR1+Y5*Y5
      CB=CB+Y*Y
      ERB=ERB+Y5*Y5
      DO 19 I1=1,NC
      SG1(N22+1-I1)=SG1(N22-I1)
19    SG1(1)=XO
20    CONTINUE
      P1=C1/ERR1
      P3=C1/256.0
      SNR1(INC)=10.0*DLOG10(P1)
      DBS(INC)=10.0*DLOG10(P3)
21    CONTINUE
      SNRE=0.00
      DBSI=0.00
      DO 22 JK=1,NB
      SNRE=SNRE+SNR1(JK)
      DBSI=DBSI+DBS(JK)
22    CONTINUE
      BNN=FLOAT(NB)
      SNRSEG=SNRE/BNN
      DBSI=DBSI/BNN
      SNRT=10.0*DLOG10(CB/ERB)
      BN=NB*256
      P10=CB/BN
      DBST=10.0*DLOG10(P10)
      WRITE(1,23)SNRSEG,SNRT
23    FORMAT(1H ,'SNRSEG=',F12.4,5X,'SNRT=',F12.4)
      WRITE(1,24)DBSI,DBST
24    FORMAT(1H ,'DBSI  =',F12.4,5X,'DBST=',F12.4)
      REWIND 5
25    CONTINUE
```

THESIS1 (CONT.)

```
26      CONTINUE
        CALL EXIT
        END
C
C        THIS SUBROUTINE CALCULATES AUTOCORRELATION COEFFICIENTS
C
        SUBROUTINE CORRE(X,KR,KRR,C,CT,INC)
        IMPLICIT INTEGER*4(I-N),REAL*8(A-H,O-Z)
        DIMENSION X(256),C(5),CT(200),CD(5)
        DO 2 J=1,3
        C1=0.00
        DO 1 K=KR,KRR
        A1=X(K)*X(K+J-1)
        C1=C1+A1
1       CONTINUE
        C(J)=C1
        C(J)=C(J)/256.0
        KRR=KRR-1
2       CONTINUE
        B=C(1)
        DO 3 J=1,3
        CD(J)=C(J)/B
3       CONTINUE
        CT(INC)=CD(2)
C
C       CD(2) IS THE FIRST SHIFT AUTOCORRELATION COEFFICIENT
C
        RETURN
        END
C
C       THIS SUBROUTINE EMPLOYS DURBIN'S SEQUENTIAL ALGORITHM FOR
C       COMPUTING N TH. ORDER PREDICTOR COEFFICIENTS PER 256 SAMPLES
C
        SUBROUTINE DURBIN(C,A1)
        IMPLICIT INTEGER*4(I-N),REAL*8(A-H,O-Z)
        DIMENSION A1(5),ALPHA(5,5),E(8),AI(8),C(5)
        N=2
        E(1)=C(1)
        AI(1)=C(2)/C(1)
        ALPHA(1,1)=AI(1)
        E(2)=(1.0-AI(1)*AI(1))*E(1)
        DO 30 I=2,N
        AI(I)=0.00
        IMM=I-1
        DO 10 J=1,IMM
10      AI(I)=AI(I)+ALPHA(J,I-1)*C(I-J+1)
        AI(I)=(C(I+1)-AI(I))/E(I)
        ALPHA(I,I)=AI(I)
        DO 20 J=1,IMM
20      ALPHA(J,I)=ALPHA(J,I-1)-AI(I)*ALPHA(I-J,I-1)
```

THESIS1 (CONT.)

```
      E(I+1)=(1.00-AI(I)*AI(I))*E(I)
30    CONTINUE
      DO 40 J=1,N
40    A1(J)=ALPHA(J,N)
      RETURN
C
C
C     THIS SUBROUTINE  ENCODES QUANTIZER STEP SIZE AT 8 BITS
C
      SUBROUTINE TRANS(Z12,ST,KK)
      IMPLICIT INTEGER*4(I-N),REAL*8(A-H,O-Z)
      DIMENSION KK(10)
      DEL=0.00387573
      DO 2 I=1,256
      FI=I
      IF(Z12-(DEL*FI))1,2,2
1     GO TO 3
2     CONTINUE
3     Z22=DEL*(FI-0.5)
      N=Z22/DEL
      KK(1)=N/128
      IF(KK(1).NE.0) N=N-128
      KK(2)=N/64
      IF(KK(2).NE.0) N=N-64
      KK(3)=N/32
      IF(KK(3).NE.0) N=N-32
      KK(4)=N/16
      IF(KK(4).NE.0) N=N-16
      KK(5)=N/8
      IF(KK(5).NE.0) N=N-8
      KK(6)=N/4
      IF(KK(6).NE.0) N=N-4
      KK(7)=N/2
      IF(KK(7).NE.0) N=N-2
      KK(8)=N
      Y2=128*KK(1)+64*KK(2)+32*KK(3)+16*KK(4)+8*KK(5)+4*KK(6)
     1+2*KK(7)+KK(8)
      ST=(Y2+0.5)*DEL
      RETURN
C
C
C     THIS SUBROUTINE JUST QUANTIZES THE INPUT ERROR SAMPLE
C
      SUBROUTINE Q(X1,X,Z1,Z2,INDIC)
      IMPLICIT INTEGER*4(I-N),REAL*8(A-H,O-Z)
      INDIC=0
      Y=Z1
      W1=Y+Z2
      W=-W1
      Z3=Z2/2.0
```

THESIS1 (CONT.)

```
      IF(X1-Y)1,9,9
1     IF(X1+Y)10,10,2
2     IF(X1)3,11,6
3     IF(X1-W-Z2)12,12,4
4     IF(X1-W-Z2-0.00001)12,12,5
5     W=W+Z2
      GO TO 3
6     IF(X1-W1+Z2)7,13,13
7     IF(X1-W1+Z2+0.00001)8,13,13
8     W1=W1-Z2
      GO TO 6
9     INDIC=1
      GO TO 13
10    INDIC=1
      GO TO 12
11    X=Z3
      GO TO 14
12    X=W+Z3
      GO TO 14
13    X=W1-Z3
14    CONTINUE
      RETURN
      END
C
C     THIS SUBROUTINE FILTERS THE OUTPUT ERROR SAMPLE AND IT IS
C     DESIGNED FOR 8KZ SAMPLING FREQUENCY AND 3.4KHZ CUT-OFF
C
      SUBROUTINE FILT(YYYY,YOUT3,AF)
      IMPLICIT INTEGER*4(I-N),REAL*8(A-H,O-Z)
      DIMENSION AF(8)
      YYYY=YYYY*0.2940
      HOLL=YYYY-(1.63702*AF(1)+0.83727*AF(2))
      YOUT=HOLL+2.0*AF(1)+AF(2)
      AF(2)=AF(1)
      AF(1)=HOLL
      HOLL1=YOUT-(1.42308*AF(3)+0.59716*AF(4))
      YOUT1=HOLL1+2.0*AF(3)+AF(4)
      AF(4)=AF(3)
      AF(3)=HOLL1
      HOLL2=YOUT1-(1.29367*AF(5)+0.45193*AF(6))
      YOUT2=HOLL2+2.0*AF(5)+AF(6)
      AF(6)=AF(5)
      AF(5)=HOLL2
      HOLL3=YOUT2-(1.23200*AF(7)+0.38383*AF(8))
      YOUT3=HOLL3+2.0*AF(7)+AF(8)
      AF(8)=AF(7)
      AF(7)=HOLL3
      RETURN
      END
```

- END -

THESIS2 DATE 15 . 03 . 1982 TIME 16:16

```
C
C
C       SGEP OR CSP(4)-SGEP WITH DPCM-AQF
C
C
        IMPLICIT INTEGER*4(I-N),REAL*8(A-H,O-Z)
        DIMENSION X(256),C(5),CD(5),A(8),B(20),SG1(10),A1(5)
        DIMENSION SNR1(200),DBS(200),KK(10),CT(200)
        DIMENSION YBAR1(5),CN(5)
C
C       B VALUES CORRESPOND TO POWER LEVELS OF INPUT SPEECH
C
        B(1)=200.0
        B(2)=250.0
C
C       IM VALUES CORRESPOND TO DIFFERENT ENCODERS IM=1 IS FOR N22
C       ORDER SGEP WHILE IM=2 FOR SWITCHED SGEP
C
C       AQ1=1.0,0.50,0.33 FOR 2,3,4 BITS QUANTIZER
C       AQ2=1.0,3.0,7.0 FOR 2,3,4 BITS QUANTIZER
C       NB=TOTAL NUMBER OF SPEECH BLOCKS
C       LK1 AND LK2 CORRESPOND TO TYPE OF SYSTEM(IM)
C       BBB,P7,P88,R7 ARE SGEP ADAPTATION CONSTANTS
C
        READ(1,*)AQ1,AQ2,L1,L2,NB,LK1,LK2,BBB
        DO 30 IM=LK1,LK2
        DO 29 K=L1,L2
        ITHK=0
        P7=5.0
        PSGEP=5.0
        P77=P7-1.0
        P88=P77/P7
        R7=10.0
        DO 1 J=1,8
1       A(J)=0.00
        N22=2
        CB=0.00
        ERB=0.00
        XO=0.000
        DO 2 N=1,N22
        CN(N)=1.00/(R7*(N**(1.00/P7)))
2       CONTINUE
        A1(1)=0.947
        A1(2)=0.000
        DO 25 INC=1,NB
        READ(5,*)(X(J),J=1,256)
        C1=0.0
        ERR1=0.0
        KR=1
        KRR=256
```

THESIS2 (CONT.)

```
      CALL CORRE(X,KR,KRR,C,CT,INC)
      AUTO=CT(INC)
      KRR=KRR+3
      YM=0.0
      DO 4 IK=1,256
      IF(IK-1)4,4,3
3     RR1=X(IK)-0.864*X(IK-1)
      RR=RR1/B(K)
      YM=YM+RR*RR
4     CONTINUE
      YM=YM/256.0
      YM=DSQRT(YM)
      Z12=A01*YM
      CALL TRANS(Z12,ST,KK)
      Z2=ST
      SN=ST/A01
      SNN=SN*SN
      PDDD=SNN+BBB
      PDD=PSGEP/PDDD
      Z1=A02*ST
      IF(IM-1)16,16,5
5     IF(AUTO-0.700)8,8,6
6     ITH=1
      IF(ITH-ITHK)7,16,7
7     A1(1)=1.523
      A1(2)=-0.720
      GO TO 16
8     IF(AUTO-0.400)11,11,9
9     ITH=2
      IF(ITH-ITHK)10,16,10
10    A1(1)=0.950
      A1(2)=-0.3450
      GO TO 16
11    IF(AUTO-0.000)14,14,12
12    ITH=3
      IF(ITH-ITHK)13,16,13
13    A1(1)=0.4930
      A1(2)=-0.210
      GO TO 16
14    ITH=4
      IF(ITH-ITHK)15,16,15
15    A1(1)=-0.6311
      A1(1)=-0.3622
16    DO 24 J=KR,KRR
      Y=X(J)
      Y=Y/B(K)
      N22=2
      IF(INC-1)17,17,20
17    IF(J-N22-1)18,18,20
18    YY=Y-0.864*X0
```

THESIS2 (CONT.)

```
      CALL Q(YY,Y1,Z1,Z2,INDIC)
      X0=0.864*X0+Y1
      X00=Y-X0
      CALL FILT(X00,Y5,A)
      C1=C1+Y*Y
      ERR1=ERR1+Y5*Y5
      CB=CB+Y*Y
      ERB=ERB+Y5*Y5
      DO 19 I1=1,N22
      SG1(N22+2-I1)=SG1(N22+1-I1)
19    CONTINUE
      SG1(1)=X0
      GO TO 24
20    E=SG1(1)-A1(1)*SG1(2)-A1(2)*SG1(3)
      DO 21 K2=1,N22
      EI=CN(K2)*SG1(K2+1)
      E1=E-EI
      E2=E+EI
21    YBAR1(K2)=DABS(E1)-DABS(E2)
      DO 22 KT=1,N22
      AD=(KT)**(P88)
      A1(KT)=A1(KT)-(R7/10.0)*(1.0/AD)*YBAR1(KT)*(PDD)
22    CONTINUE
      EST=A1(1)*SG1(1)+A1(2)*SG1(2)
      YY=Y-EST
      CALL Q(YY,Y1,Z1,Z2,INDIC)
      X0=EST+Y1
      X00=Y-X0
      CALL FILT(X00,Y5,A)
      C1=C1+Y*Y
      ERR1=ERR1+Y5*Y5
      CB=CB+Y*Y
      ERB=ERB+Y5*Y5
      DO 23 I1=1,N22
      SG1(N22+2-I1)=SG1(N22+1-I1)
23    CONTINUE
      SG1(1)=X0
24    CONTINUE
      P1=C1/ERR1
      P3=C1/256.0
      SNR1(INC)=10.0*DLOG10(P1)
      DBS(INC)=10.0*DLOG10(P3)
      ITHK=ITH
25    CONTINUE
      SNRE=0.00
      DBSI=0.00
      DO 26 JK=1,NB
      SNRE=SNRE+SNR1(JK)
      DBSI=DBSI+DBS(JK)
26    CONTINUE
```

THESIS2 (CONT.)

```
      BN=FLOAT(NB)
      SNRSEG=SNRE/BN
      DBSI=DBSI/BN
      SNRT=10.0*DLOG10(CB/ERB)
      BNN=NB*256
      P10=CB/BNN
      DBST=10.0*DLOG10(P10)
      WRITE(1,27)SNRSEG,SNRT
27    FORMAT(1H ,'SNRSEG=',F12.4,5X,'SNRT=',F12.4)
      WRITE(1,28)DBSI,DBST
28    FORMAT(1H ,'DBSI  =',F12.4,5X,'DBST=',F12.4)
      REWIND 5
29    CONTINUE
30    CONTINUE
      CALL EXIT
      END
C
C     THE SUBROUTINES ARE THE SAME AS BEFORE
C     (SEE PROGRAM THESIS1)
C
```

REFERENCES

1. LITTLE, P., *"Oral and written communication"*, Longman Group Ltd., London, 1973.

2. FLANAGAN, J., *"Speech analysis and synthesis"*, Springer-Verlag, 1976.

3. FLOAD, J.E., *"Telecommunications networks"*, Peter Peregrinus, 1975.

4. MARSHALL, G.J., *"Principles of digital communication"*, McGraw Hill, London, 1980.

5. CATTERMOLE, K.W., *"Principles of pulse code modulation"*, Iliffe Books Ltd., London, 1969.

6. AARON, M.R., *"Digital communications"*, IEEE Comm. Magazine, January, 1979, Vol.17, pp.16-26.

7. XYDEAS, C.S., *"Differential encoding techniques applied to speech signals"*, Ph.D. Thesis, L.U.T., December, 1978.

8. STEELE, R., *"Delta modulation systems"*, Pentech Press, London, 1975.

9. SAKANE, F., *"On instantaneously adaptive DM and encoding of video signals"*, Ph.D. Thesis, L.U.T., 1977.

10. HOLMES, J.N., *"A survey of methods for digitally encoding speech signals"*, IERE. Conf. Proceedings in Signals in Telecom., Loughborough, April, 1981.

11. RABINER, L.R., and SCHAFER, R.W., *"Digital processing of speech signals"*, Prentice-Hall, New York, 1978.

12. PATRICK, P.J., STEELE, R., XYDEAS, C.S., *"Voiced/unvoiced band-switching system for transmission of 6 kHz speech over 3.4 kHz telephone channels"*, IERE Journal, Vol.51,No.5, pp.233-235, May, 1981.

13. FLANAGAN, J.L., SCHROEDER, M.R., ATAL, B.S., CROCHIERE, R.E., JAYANT, N.S., TRIBOLET, J.M., *"Speech coding"*, IEEE Trans. Commu. Vol. COM-27, No.4, pp.710-737, April, 1979.

14. RICHARDS, D.L., *"Telecommunication by speech"*, Butterworths, London, 1973.

15. PETERSON, G.E., and BARNEY, H.L., *"Control methods used in a study of the vowels"*, J. Acoustics Soc.Ame. Vol.24, No.2, pp.175-184, March, 1952.

16. SCHROEDER, M.R., *"Vocoders: Analysis and synthesis of speech"*, Proc. IEEE, Vol.54, pp.720-734, May, 1966.

17. MOYE, L.S., *"Digital transmission of speech at low bit rates"*, Electrical Communication, Vol.47, No.4, pp.212-223, November, 1972.

18. HOLMES, J.N., *"The JSRU channel vocoder"*, Proc. IEE, 127, Part F, pp.53-60, March, 1980.

19. GOLD, B., BLANKENSHIP, P.E. and McAULAY, R.J., *"New applications of channel vocoders"*, IEEE Trans.Acous. Speech, Vol.ASSP-29, No.1, pp.13-23, February, 1981.

20. SCHAFER, R.W. and RABINER, L.R., *"Digital representations of speech signals"*, Proc. IEEE, Vol.63, No.4, pp.662-677, April, 1975.

21. HASKELL, B.G. and STEELE, R., *"Audio and video bit-rate reduction"*, IEEE Proc., Vol.69, No.2, February 1981, pp.252-262.

22. STEELE, R., *"Parametric representation of speech signals"*, Monograph 7, L.U.T., 1976.

23. SMITH, C.P., *"Voice communication method using pattern matching for data compression"*, J. Acoustic Soc.Ame., 35, p.805, 1963.

24. CHENG, M., *"The low bit-rate coding of speech signals"*, Ph.D. Thesis, Imperial College, 1976.

25. PATRICK, P.J., *"Private communication"*, Dept. of Elec.Eng., L.U.T.

26. MARKEL, J.D. and GRAY A.H.Jr., *"Linear prediction of speech"*, Springer-Verlag, 1976.

27. GOLD, B., *"Digital speech networks"*, Proc. of IEEE, Vol.65, No.12, pp.1636-1658, December, 1977.

28. MAKHOUL, J., *"Linear prediction: A tutorial review"*, Proc. IEEE, Vol.63, No.4, pp.561-580, April, 1975.

29. BARNWELL, T.P., *"Windowless techniques for LPC analysis"*, IEEE Trans. on Acous.Spe., Vol. ASSP-28, No.4, pp.421-427, August, 1980.

30. McGONEGAL, C.A., RABINER, L.R. and McDERMOTT, B.J., *"Speaker verification by human listeners over several speech transmission systems"*, BSTJ Vol. 57, No.8, pp.2887-2900, October, 1978.

31. ATAL, B.S. and HANAUER, S.L., *"Speech analysis and synthesis by linear prediction of the speech wave"*, J.Acoustic Soc.Ame., Vol.50, No.2, pp.637-655, August, 1971.

32. ITAKURA, F. and SAITO, S., *"On the optimum quantization of feature parameters in the Parcor speech synthesizer"*, IEEE Proc. Conf. Speech Commu. Process, pp.434-437, 1972.

33. JAYANT, N.S., *"Digital coding of speech waveforms: PCM, DPCM and DM quantizers"*, IEEE Proc., Vol. 62, No.5, pp.611-632, May, 1974.

34. REEVES, A.H., *"French Patent"*, 852183, 1938.

35. SMITH, B., *"Instantaneously companding of quantized signals"*, BSTJ, Vol. 36, pp.653-709, May, 1977.

36. JAYANT, N.S., *"Adaptive quantization with a one-word memory"*, BSTJ, Vol. 52, No.7, pp.1119-1144, September 1973.

37. GOODMAN, D.J. and GERSHO, A., *"Theory of adaptive quantizer"*, IEEE Trans. Commu., Vol. COM-22, No.8, pp.1037-1045, August, 1974.

38. NOLL, P., *"Adaptive quantizing in speech coding systems"*, Int. Zurich Seminar on Digital Comm., pp.B3.1-B3.6, March, 1974.

39. GERSHO, A., *"Quantization"*, IEEE Commu. Society Magazine, pp. 16-29, September, 1977.

40. CHOCHIERE, R.E., *"A mid-rise/mid-thread quantizer switch for improved idle-channel performance in adaptive coders"*, BSTJ, Vol.57, No.8, pp.2953-2955, October, 1978.

41. NOLL, P., *"Effects of channel errors on the SNR performance of speech encoding schemes"*, BSTJ, Vol.54, No.9, pp.1615-1636, November, 1975.

42. STEELE, R., *"Lecture notes given at Loughborough University of Technology"*, Dept. of Elec.Eng., L.U.T.

43. PANTER, P.F. and DITE, W., *"Quantizing distortion in pulse-count modulation with non-uniform spacing of levels"*, IRE Proc., Vol.39, pp.44-48, 1951.

44. MAX, J., *"Quantization for minimum distortion"*, IRE Trans. Infor. Theory, Vol. IT-6, No.1, pp.7-12, March, 1960.

45. STROH, R.W. and PAEZ, M.D., *"A comparison of optimum and logarithmic quantization for speech PCM and DPCM systems"*, IEEE Trans. on Commu., Vol. COM-21, No.6, pp.752-757, June, 1973.

46. PAEZ, M.D. and GLISSON, T.H., *"Minimum mean-square error quantization in speech PCM and DPCM systems"*, IEEE Trans. on Commu. Vol. COM-20, No.2, pp.225-230, April, 1972.

47. FLEISCHER, P., *"Sufficient conditions for achieving minimum distortion in a quantizer"*, IEEE Int.Conf. record, pp.104-111, 1964.

48. NOLL, P., *"A comparative study of various quantization schemes for speech encoding"*, BSTJ, Vol. 54, No.9, pp.1597-1614, November, 1975.

49. STROH, R.W., *"Optimum and adaptive differential PCM"*, Ph.D. Dissertation, Brooklyn Institute, June 1970.

50. STEELE, R., and JAYANT, N.S., *"Statistical block protection coding for DPCM-AQF speech"*, IEEE, National Tele.Conf.Proc., NTC80, Houston, Texas, U.S.A., pp.50.2.1-5, November, 1980.

51. CUMMISKEY, P., JAYANT, N.S. and FLANAGAN, J.L., *"Adaptive quantization in differential PCM coding of speech"*, BSTJ, Vol. 52, No.7, pp.1105-1118, September, 1973.

52. CUTLER, C.C., *"Differential quantization of communication signals"*, Patent No. 2,605361, July, 1952.

53. O'NEAL, J.B. Jr., *"Predictive quantizing systems for the transmission of television signals"*, BSTJ, Vol.45, No.5, pp. 689-721, May-June, 1966.

54. NITADORI, K., *"Statistical analysis of DPCM"* Journal of Inst. Electr. Commu.Eng. of Japan, Vol.48, pp.17-26, February, 1965.

55. McDONALD, R.A., *"Signal-to-noise and idle channel performance of DPCM systems - Particular application to voice signals"*, BSTJ, Vol. 45, No.7, pp.1123-1151, September, 1976.

56. O'NEAL, J.B.Jr., *"Signal-to-quantizing noise ratios for differential PCM"*, IEEE Trans. on Commu. Techn. Vol. COM-19, No.4, pp.568-569, August, 1971.

57. O'NEAL, J.B. Jr. and STROH, R.W., *"Differential PCM for speech and data signals"*, IEEE Trans. on Commu. Vol. COM-20, No.5, pp.900-912, October, 1972.

58. ATAL, B.S. and SCHROEDER, M.R., *"Adaptive predictive coding of speech signals"*, BSTJ, Vol. 49, No.8, pp.1973-1986, October, 1970.

59. STEELE, R., *"Linear predictors and differential PCM for speech signals"*, Monograph 2, L.U.T., 1977.

60. GIBSON, J.D., *"Adaptive prediction in speech differential encoding systems"*, IEEE Proc. Vol. 68, No.4, pp.488-525, April, 1980.

61. JAYANT, N.S., *"Pitch adaptive DPCM coding of speech with 2 bits quantization and fixed spectrum prediction"*, BSTJ, Vol. 56, No.3, pp.439-454, March, 1977.

62. CUMMISKEY, P., *"Adaptive DPCM for speech processing"*, Ph.D. Dissertation, Newark College of Engineering, Newark, N.J., 1973.

63. GIBSON, J.D., JONES, S.K. and MELSA, J.L., *"Sequentially adaptive prediction and coding of speech signals"*, IEEE Trans. on Commu. Vol. COM-22, No.1, pp.1789-1797, November, 1974.

64. GIBSON, J.D., *"Sequentially adaptive backward in ADPCM"*, IEEE Trans. on Commu. Vol. COM-26, No.1, pp.145-150, January, 1978.

65. GIBSON, J.D. and CROSS, E.A., *"Fixed-tap ADPCM system divergence and bound on the robust quantizer overload point"*, IEEE Trans. on Commu. Vol. COM-26, No.6, pp.827-832, June, 1978.

66. CROCHIERE, R.E., *"An analysis of 16 kb/s sub-band coder performance: Dynamic range, tandem connections and channel errors"*, BSTJ, Vol.57, No.8, pp.2927-2952, October, 1978.

67. GOODMAN, D.J. and WILKINSON, R.M., *"A robust adaptive quantization"*, IEEE Trans. on Commu. Vol. COM-23, No.11, pp.1362-1365, November, 1975.

68. SCAGLIOLA, C., *"Evaluation of adaptive speech coders under noisy channel conditions"*, BSTJ, Vol.58, No.6, pp.1369-1394, July-August 1979.

69. COHN, D.L. and MELSA, J.L., *"Pitch compensating quantizer"*, Int.Conf. on ASSP, pp.258-261, Philadelphia, U.S.A., April, 1976

70. COHN, D.L. and MELSA, J.L., *"The residual encoder"*, IEEE Trans. on Commu. Vol. COM-23, No.9, pp.935-941, September, 1975.

71. QURESHI, S.U. and FORNEY, D.G., *"A 9.6/16 Kb/s speech digitizers"*, IEEE Int.Conf. on Commu. Vol.11, June, 16-18, pp.30,1975.

72. XYDEAS, C.S., FARUQUI, M.N. and STEELE, R., *"Envelope dynamic ratio quantizer"*, IEEE Trans. on Com. Vol. COM-28, No.5, pp.720-728, May, 1980.

73. O'NEAL, J.B. Jr., *"Entropy coding in speech and television DPCM Systems"*, IEEE Trans. on Int.Theory, Vol. IT-17, pp.758-761, November, 1971.

74. VIRUPAKSHA, K. and O'NEAL, J.B. Jr., *"Entropy coded adaptive differential pulse code modulation for speech"*, IEEE Trans. Commu. Vol. COM-22, pp.777-787, June, 1974.

75. MAKHOUL, J. and BEROUTI, M., *"Adaptive noise spectral shaping and entropy coding in predictive coding of speech"*, IEEE Trans. ASSP, Vol.ASSP-27, No.1, pp.63-73, February, 1979.

76. NOLL, P., *"On predictive quantizing schemes"*, BSTJ, Vol. 57, No.5, pp.1499-1532, May-June, 1978.

77. ATAL, B.S. and SCHROEDER, M.R., *"Predictive coding of speech signals and subjective error criteria"*, Int.Conf. on ASSP, ICASSP, Tulsa, U.S.A., pp.573-576, 1978.

78. ATAL, B.S., and SCHROEDER, M.R., *"Predictive coding of speech signals and subjective error criteria"*, IEEE Trans. on Acoustic Speech Sig. Proc., Vol. ASSP-27, pp.247-254, June, 1979.

79. DELORAINE, E.M., VAN MIERO, S. and DERJAVITCH, B., *"French patent"*, 932140, 1946.

80. CUMMISKEY, P., *"Single integration ADPCM"*, ICASSP, pp.247-250, Philadelphia, U.S.A., April, 1976.

81. ABATE, J.E., *"Linear-adaptive DM"*, IEEE Proc. Vol.55, No.3, pp. 298-308, March, 1967.

82. GOODMAN, D.J., *"DM granular quantization noise"*, BSTJ, Vol.48, pp.1197-1218, May-June 1969.

83. GREENSTEIN, L.J., *"Slope overload noise in LDM with Gaussian input"*, BSTJ, Vol. 52, No.3, pp.387-421, March, 1973.

84. STEELE, R., *"SNR formula for LDM with band-limited flat and RC shaped Gaussian signals"*, IEEE Trans. on Commu. Vol. COM-28, No.12, pp.1977-1984, December, 1980.

85. O'NEAL, J.B. Jr., *"DM modulation quantizing noise analysis"*, BSTJ, Vol.45, No.1, pp.117-141, January, 1966.

86. DE JAGER, F., *"Delta modulation - A new method of PCM transmission using the 1 unit code"*, Philips Res. Rep. pp.442-466, December, 1952.

87. CUMMISKEY, P., *"Single integration ADM"*, BSTJ, Vol. 54, No.8, pp.1463, October, 1975.

88. STEELE, R., *"Delta modulation"*, unpublished report, L.U.T., July, 1977.

89. WINKLER, M.R., *"HIDM"* IEEE Int. Convention Record, No.8, pp. 260-265, 1963.

90. JAYANT, N.S. CUMMISKEY, P. and FLANAGAN, J.L., *"Design and implementation of an ADM"*, Prod. of Int. Conf. on Speech and Processes, Boston, U.S.A., April, 1972.

91. JAYANT, N.S., *"Characteristics of delta modulator"*, IEEE Proc., pp.428-429, March, 1971.

92. KYAW, A.T. and STEELE, R., *"Constant factor DM"*, Elect. Letters, Vol.9, No.4, pp.96-97, February, 1973.

93. DHADESUGOOR, V.R., ZIEGLER, C., and SHILLING, D.L., *"Delta modulator in packet voice networks"*, IEEE Trans. on Commu. Vol. COM-28, 1, pp.33-51, January, 1980.

94. UN, C.K. and LEE, H.S., *"A study of the comparative performance of ADM systems"*, IEEE Trans. on Commu. Vol. COM-28, No.1, pp.96-101, January, 1980.

95. TRIBOLET, J.M., NOLL, P., McDERMOTT, B.J., CROCHIERE, R.E., *"Study of complexity and quality of speech waveform coders"*, ICASSP, Tulsa, U.S.A., pp.589-590, 1978.

96. CROCHIERE, R.E., WEBBER, S.A., FLANAGAN, J.L., *"Digital coding of speech in sub-bands"*, BSTJ, Vol. 55, No.8, pp.1069-1085, October, 1976.

97. CROCHIERE, R.E., *"On the design of SBC for low bit-rate speech communication"*, BSTJ, Vol.65, pp.747-770, May-June, 1977.

98. ZELINSKI, R., and NOLL, P., *"Adaptive transform coding of speech signals"*, IEEE Trans. ASSP, Vol. ASSP-25, No.4, pp.299-309, August, 1977.

99. TRIBOLET, J.M., NOLL, P., McDERMOTT, B.J. and CROCHIERE, R.E., *"A comparison of the performance of four low bit-rate speech waveform coders"*, BSTJ, Vol.58, No.3, pp.699-712, March, 1979.

100. KING, R.A. and GOSLING, W., *"Time encoded speech"*, IEE Conf. Publications, No. 180, pp.140-143, 1979.

101. WIENER, N., *"The extrapolation, interpolation and smoothing of stationary time series"*, Wiley, 1949.

102. KOLMOGOROFF, A., *"Interpolation and extrapolation of stationary series"*, Bulletin de l'Academie des Sciences de l'U.R.S.S. Series Maths., 1942.

103. PRATT, W.K., *"Digital image processing"*, A. Wiley - Interscience Publication, pp.294-295, 1978.

104. NOLL, P., *"Non-adaptive and adaptive differential pulse code modulation of speech signals"*, Polytechnische Tijdscrift, No. 19, pp.623-629, 1972.

105. GIBSON, J.D., *"Predictor optimization and performance improvement in DPCM"*, IEEE Int.Conf.Commu. ICC80, pp.36.6.1-6, 1980.

106. NAG, *"Fortran Library Manual"*, Loughborough University of Technology.

107. STEELE, R., *"Private Communication"*, Bell Laboratories, April, 1981.

108. SAMBUR, M.R. and JAYANT, N.S., *"LPC analysis/synthesis from speech inputs containing quantization noise or additive white noise"*, IEEE Trans. on ASSP, Vol. ASSP-24, No.6, pp.488-494, December, 1976.

109. STEELE, R., GOODMAN, D.J. and McGONEGAL, C.A., *"A difference detection and correction scheme for combating DPCM transmission errors"*, IEEE Trans. Vol. COM-27, No.1, pp.252-255, January, 1979.

110. JAYANT, N.S., *"Step-size transmitting differential coders for mobile telephony"*, BSTJ, Vol.54, No.9, pp.1557-1581, November, 1975.

111. WIDROW, B., *"Technical Report No. 6764-6, Adaptive Filters I: Fundamentals"*, System Theory Labs. Stanford Elect. Labs., Stanford Uni. U.S.A., December, 1966.

112. NAGUMA, J. and NODA, A., *"A learning method for system identification"*, IEEE Trans. on Auto Control, Vol. AC-12, No.3, pp.282-287, June 1967.

113. Final Report Contract, No. DCA 100-72-C-0036, *"Development of a configuration concept of a speech digitizer based on adaptive estimation techniques"*, Information and Control Services Centre, Inst. of Tech. Southern Methodist University, Dallas, Texas, U.S.A., August, 1973.

114. GIBSON, J.D., MELSA, J.L. and JONES, S.K., *"Digital speech analysis using sequential estimation techniques"*, IEEE Trans. Vol.ASSP-23, No.4, pp.362-369, August, 1975.

115. MARK, J.W. and DASIEWICZ, P.P., *"Application of iterative algorithms to adaptive predictive coding"*, Journal of Cybernetics, No.7, pp.279-317, 1977.

116. EVCI, C.C., STEELE, R. and XYDEAS, C.S., *"Sequential gradient estimation predictor for speech signals"*, ICASSP Conf. Proceedings, Washington D.C., U.S.A., pp.723-726, April, 1979.

117. WIDROW, B., and GLOWER, J.R. et al, *"Adaptive noise cancelling: Principles and applications"*, IEEE Proc. Vol.63, No.2, pp. 1692-1716, December, 1975.

118. HIRUY, F., *"Reduction of acoustic noise in speech by adaptive noise cancelling"*, M.Sc. Project, Loughborough University of Technology, September, 1980.

119. SAKRISON, D.J., *"Application of stochastic approximation methods to the system optimization"*, D.Sc. Thesis, Dept. Elec.Eng., M.I.T., June, 1961.

120. GIBSON, J.D., *"Sequentially adaptive prediction in ADPCM speech coders"*, NTC76, Dallas, Texas, pp.29.5.1-29.5.5, November,1976.

121. GIBSON, J.D., BERGLUND, V.P. and SAUTER, L.C., *"Kalman backward adaptive predictor coefficient identification in ADPCM with PCQ"*, IEEE Trans. on Comm. Vol. COM-28, No.3, pp.361-371, March, 1980.

122. UN, C.K., CYNN, M.N., *"A performance comparison of ADPCM and ADM coders"*, NTC 80, Houston, Texas, pp.50.1.1-50.1.5, December, 1980.

123. MITRA, D., *"A generalized adaptive quantization system with a new reconstruction method for noisy transmission"*, IEEE Trans. on Commu. Vol. COM-27, No.11, pp.1681-1689, November, 1979.

124. EINARSSON, G., *"A robust AQ with extended dynamic range"*, IEEE Trans. on Commu. Vol. COM-29, No.6, pp.830-836, June, 1981.

125. FALCONER, D., *"Private communication"*, Bell Labs., N.J., U.S.A, July, 1980.

126. TOU, J.T., *"Sampled-data control systems"*, McGraw-Hill, Elec.Eng. Series, 1959, pp.238-242.

127. CHANG, C.S., *"An improved residual encoder for speech transmission"*, Int.Con. of IEEE on ASSP, Washington D.C., pp.542-545, April,1979.

128. STEELE, R. and JAYANT, N.S., *"Statistical block protection coding for DPCM-AQF speech"*, IEEE Trans. Comm. Vol. COM-28, No.11, pp.1899-1907, Nov. 1980.

129. BERLEKAMP, E.R., *"Algebraic coding theory"*, New York, McGraw-Hill, 1968.

130. EVCI, C.S., STEELE, R. and XYDEAS, C.S., *"DPCM-AQF using second-order adaptive predictors for speech signals"*, IEEE Trans. on ASSP, Vol. ASSP-29, No.3, pp.337-341, June, 1981.

131. GIBSON, J.D., *"Comparisons and analyses of forward and backward adaptive prediction in ADPCM"*, IEEE National Tele. Conf. Proc. NTC78, U.S.A., pp.19.2.1-19.2.5, December, 1978.

132. KNORR, S.G., *"Reliable voiced/unvoiced decision"*, IEEE Trans. on ASSP, Vol. ASSP-27, No.3, pp.263-267, June, 1979.

133. PATRICK, P.J., XYDEAS, C.S., STEELE, R. and CHAM, W.K., *"Wideband quality speech encoders with bit-rates of 16-32 Kb/s"*, Int.Conf. Proc. of IEEE on ASSP, Atlanta, U.S.A., pp.844-847, Aptil, 1981.

134. EVCI, C.C., XYDEAS, C.S. and STEELE, R., *"Sequential adaptive predictors for ADPCM speech encoders"*, IEEE, National Tele. Conf. Proc. NTC81, New Orleans, U.S.A., pp. E8.1.1-5, November, 1981.

135. XYDEAS, C.S. and EVCI, C.C., *"A comparative study of DPCM-AQF speech codecs for bit-rates of 16 to 32 Kb/s"*, to appear in IEEE Int.Conf. Proc. on ASSP, Paris, France, May, 1982, Session S14.

136. RABINER, L.R., CHENG, M.J., et al, *"A comparative performance study of several pitch detection algorithms"*, IEEE Trans. on ASSP, Vol. ASSP-24, No.5, pp.399-418, October, 1976.

137. ACKROYD, M.H., *"Digital filters"*, Butterworths, London, 1973.