1

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

National Library
of Canada

Bibliothèque nationale
du Canada

Canadian Theses Service

Services des thèses canadiennes

Ottawa, Canada
K1A 0N4

CANADIAN THESES

NOTICE

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

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

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

Previously copyrighted materials (journal articles, published tests, etc.) are not filmed.

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

**THIS DISSERTATION
HAS BEEN MICROFILMED
EXACTLY AS RECEIVED**

THÈSES CANADIENNES

AVIS

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

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

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

Les documents qui font déjà l'objet d'un droit d'auteur (articles de revue, examens publiés, etc.) ne sont pas microfilmés.

La reproduction, même partielle, de ce microfilm est soumise à la Loi canadienne sur le droit d'auteur, SRC 1970, c. C-30.

**LA THÈSE A ÉTÉ
MICROFILMÉE TELLE QUE
NOUS L'AVONS REÇUE**

DIGITAL SPEECH CODING

FOR

LAND MOBILE RADIO

by

Robert D. Hoyle, B.Eng.

A thesis submitted to the
Faculty of Graduate Studies and Research
in partial fulfillment of the requirements
for the degree of
Master of Engineering

Department of Systems and Computer Engineering
Carleton University
Ottawa, Ontario

© April 1986

Permission has been granted to the National Library of Canada to microfilm this thesis and to lend or sell copies of the film.

The author (copyright owner) has reserved other publication rights, and neither the thesis nor extensive extracts from it may be printed or otherwise reproduced without his/her written permission.

L'autorisation a été accordée à la Bibliothèque nationale du Canada de microfilmer cette thèse et de prêter ou de vendre des exemplaires du film.

L'auteur (titulaire du droit d'auteur) se réserve les autres droits de publication; ni la thèse ni de longs extraits de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation écrite.

ISBN 0-315-29850-2

The undersigned recommends to the Faculty of Graduate
Studies and Research acceptance of the thesis

"Digital Speech Coding for Land Mobile Radio "

submitted by Robert D. Hoyle, B.Eng., in partial fulfillment
of the requirements for the degree of Master of Engineering.

Thesis Supervisor

Chairman,
Department of Systems and
Computer Engineering

Carleton University

April 1986

ABSTRACT

Subjective quality measurements on several digital speech coders were performed using the "Mean Opinion Score (MOS)" method. The three speech coding methods tested were: continuously variable slope delta modulation (CVSDM) coding, adaptive predictive coding (APC), and residually excited linear predictive (RELP) coding. Several versions of each coder, with transmission rates in the range of 7.5 to 16.1 kbit/sec, were simulated. Five different channel conditions, including three derived from land mobile radio field experiments, were applied to the speech coders' encoded output to study the effects. The results show that of the three coders, the CVSDM coder is the most robust to channel errors, but produces reconstructed output speech of unacceptable quality. The 14.6 kbit/sec RELP coder produces relatively good output speech quality, exhibits a mild degree of robustness to mobile radio channel errors, and is slightly less complex than the APC coder. Of the three digital speech coders tested, the RELP coder is the most suitable for use with land mobile radio.

ACKNOWLEDGEMENTS

I would like to express my gratitude to my thesis supervisor, Dr. D. D. Falconer, for the opportunity to work under him, and for his advice and guidance throughout the course of this thesis.

I would like to acknowledge the work of B. McPhail who provided me with the error pattern data files that he extracted from the digital, mobile radio field trials.

TABLE OF CONTENTS

	<u>PAGE</u>
ABSTRACT	(i)
ACKNOWLEDGEMENTS	(ii)
TABLE OF CONTENTS	(iii)
LIST OF ILLUSTRATIONS	(vi)
LIST OF TABLES	(ix)

CHAPTER ONE : INTRODUCTION

1.1 Review of Digital Speech Coding for Mobile and Portable Radio	1-1
1.2 Thesis Objectives	1-7
1.3 Thesis Organization	1-7

CHAPTER TWO : THE MOBILE RADIO CHANNEL, DIGITAL SPEECH
CODING AND THE INITIAL INVESTIGATION

2.1 The Mobile Radio Propagation Mechanism and Impairments	2-1.
2.2 Digital Speech Coding	
2.2.1 Requirements for Mobile Radio	2-5.
2.2.2 Review of Digital Speech Coding Algorithms	2-6
2.2.3 Recent Trends in Speech Coding	2-11
2.3 Selection of Suitable Speech Coding Algorithms ...	2-14

CHAPTER THREE : BACKGROUND TO THE SPEECH CODING
ALGORITHMS, THEIR DEVELOPMENT AND
CONCURRENT TESTING

3.1	Developmental Laboratory Set-up	3-1
3.2	Development and Testing of the Speech Coding Algorithms	3-5
3.3	Coder 1 : Continuously Variable Slope Deltamodulation (CVSDM)	
3.3.1	Introduction to Deltamodulation Coding ...	3-9
3.3.2	Basic Deltamodulation Coding System and Advanced Topics	3-10
3.3.3	Continuously Variable Slope Deltamodulation Coding	3-20
3.3.4	Final Configurations of the CVSDM Coder ..	3-27
3.4	Coder 2 : Adaptive Predictive Coder with Variable Length Coding of the Residual	
3.4.1	Introduction to Adaptive Predictive Coding (APC)	3-31
3.4.2	Basic APC System	3-31
3.4.3	Prediction of Speech	3-35
3.4.4	Additions to the APC Structure	3-41
3.4.5	Encoding the Predictor Parameters	3-51
3.4.6	Efficient Quantization of the Prediction Residual	3-58
3.4.7	Contending with Channel Errors	3-74
3.4.8	Final Configurations of the APC Coder	3-79

3.5	Coder 3 : Residually Excited Linear Prediction Coder (RELP)	
3.5.1	Introduction to Residually Excited Linear Prediction Coding (RELP)	3-84
3.5.2	Basic RELP System	3-85
3.5.3	Implementation of RELP coder with Pitch Prediction and Perturbed Spectral Folding	3-93
3.5.4	Contending with Channel Errors	3-108
3.5.5	Final Configurations of the RELP Coder	3-109

CHAPTER FOUR : OBJECTIVE AND SUBJECTIVE EVALUATIONS OF THE SPEECH CODERS

4.1	Acquisition of the Mobile Radio Channel Data	4-1
4.2	Extraction of the Bit Error Patterns	4-3
4.3	Results of the Objective Testing	4-7
4.4	Results of the Subjective Testing	4-19
4.5	Discussion of the Results	4-30
4.6	Conclusions and Recommendations of a Digital Speech Coding Algorithm for the Mobile Radio Channel	4-38

CHAPTER FIVE : SUMMARY AND SUGGESTIONS FOR FURTHER RESEARCH

5.1	Summary	5-1
5.2	Suggestions for Further Research	5-4

APPENDIX A	A-1
------------	-----

REFERENCES	R-1
------------	-----

LIST OF ILLUSTRATIONS

1.1-1	Hexagonal cell configuration	1-2
2.1-1	Primary modes of multi-path propagation	2-2
2.1-2	Relationship between received signal envelope and bit errors	2-4
2.2-1	Speech quality vs. bit rate for three different speech coding methods	2-12
3.1-1	Block diagram of the developmental laboratory set-up	3-2
3.1-2	Salient features of the developmental set-up	3-4
3.2-1	Block diagram of the speech coding simulator ..	3-6
3.3-1	Block diagram of single-integration DM	3-11
3.3-2	Block diagram of an analog implementation of single-integration DM	3-13
3.3-3	LDM waveform examples	3-16
3.3-4	Block diagram of the CVSDM : (a) encoder and (b) decoder	3-21
3.3-5	Overload detector logic	3-22
3.4-1	Block diagram of the basic open-loop predictive coding system : (a) coder and (b) decoder	3-33
3.4-2	Block diagram of the closed-loop predictive coding system : (a) coder and (b) decoder	3-34
3.4-3	Block diagram of speech pre-emphasis	3-43
3.4-4	APC system with Noise Feedback Coding filter $F(z)$	3-46
3.4-5	Closed-loop APC system with both spectral and pitch predictors	3-48

3.4-6	Block diagram of the open-loop APC coder used to derive the open-loop prediction residual signal $u(n)$	3-57
3.4-7	Calculation of the bit rate of the side information for the APC coder with and without pitch prediction	3-59
3.4-8	APC residual waveform examples	3-61
3.4-9	Closed-loop prediction residual quantizer	3-63
3.4-10	Switched quantizer, consisting of a center-clipping multi-level quantizer and a one-bit quantizer	3-72
3.4-11	Transmission data block	3-80
3.4-12	Calculation of the bit rate of the final configuration of the APC coder	3-80
3.4-13	APC simulation results using test speech sample one	3-83
3.5-1	Block diagram of a basic RELP speech coder : (a) transmitter and (b) receiver	3-86
3.5-2	Typical spectra for decimation by an integer factor L	3-88
3.5-3	Typical spectra for the generation of the full-band prediction residual	3-91
3.5-4	Frequency response of a lowpass filter with a decimation factor of 2	3-96
3.5-5	Block diagram of a RELP transmitter incorporating pitch prediction in a closed-loop configuration	3-97
3.5-6	Derivation of the open-loop prediction residual $u(n)$	3-99
3.5-7	Calculation of the bit rate of the side information for the RELP coder	3-101
3.5-8	Block diagram of the RELP decoder incorporating pitch prediction and perturbed spectral folding	3-103

3.5-9	Flowchart of the process for randomly perturbing the non-zero samples of the spectrally folded baseband residual signal	3-105
3.5-10	Calculations of the bit rate of the final configuration of the RELP coder	3-110
3.5-11	Simulation performance of the three RELP coder configurations using test speech sample one	3-111
4.2-1	Method to approximately translate a 4.8 kbit/sec error pattern to a higher rate	4-8
4.3-1	Plot of segmental SNR vs. average bit error rate	4-14
4.3-2	Plot of segmental SNR vs. bit rate	4-16
4.4-1	Subjective testing set-up	4-21
4.4-2	Five-point quality scale or alternative impairment scale used to evaluate the quality of the output speech samples	4-21
4.4-3	Plot of mean opinion score vs. average bit error rate	4-25
4.4-4	Plot of mean opinion score vs. bit rate	4-27

LIST OF TABLES

1.1-1	Frequency spectrum allocations for land mobile radio in the 150 to 950 MHz band	1-2
3.3-1	Segmental SNR results of the CVSDM simulations for sampling rates of : (a) 12.0 kHz and (b) 16.1 kHz	3-30
3.4-1	Maximum and Minimum limits for the log area ratio quantizer for : (a) no pre-emphasis and (b) fixed pre-emphasis ($\alpha=0.4$)	3-54
3.4-2	Bit allocation and number of quantizer levels for encoding the log area coefficients	3-54
3.4-3	Bit allocations and range of quantization for the pitch predictor	3-56
3.4-4	Frequency distribution of the quantizer levels employed for an eleven-level center-clipping quantizer	3-63
3.4-5	Self-synchronizing code for the quantizer of Figure 3.4-9	3-66
3.4-6	Segmental SNR performance for various reconstruction level values of the one-bit quantizer	3-73
3.5-1	Specifications for the three lowpass filters which perform baseband extraction	3-96
3.5-2	Table of optimum decision values $x(n)$ and reconstruction $y(n)$ values for the Laplacian quantizer	3-99
4.2-1	Statistics of the error patterns generated from the mobile radio experimental field data	4-5
4.3-1	Respective bit rates of the final configurations of the speech coders	4-10

4.3-2	Five error files used to simulate five different error conditions	4-10
4.3-3	Objective test results using the segmental SNR measure	4-13
4.4-1	(a) Mean opinion score (MOS) subjective test results	4-23
4.4-1	(b) Standard deviations of the mean opinion score subjective test results	4-24
4.4-2	Mean opinion score rankings for various channel error conditions	4-34

CHAPTER ONE

INTRODUCTION

1.1 Review of Digital Speech Coding for Mobile and Portable Radio

Mobile radio communications has had a long history which began with early maritime use for emergency and safety communications. However, there have been two major problems that have plagued mobile radio communication development in the past. These problems have been: inadequate radio frequency allocation for the growing demand; and technological problems in dealing with the mobile propagation mechanism which not only degrades the signal transmission performance, but also limits the service area.

Table 1.1-1 lists the frequency spectrum allocations for private sector land mobile radio in the 25-950 MHz band. (from [1] and [2]). Presently, most of the existing land mobile systems are FM analog voice systems. To ensure high quality FM voice transmission, the channel spacing (RF bandwidth) is specified as 30 kHz in the 150 MHz band or 25 kHz in the 450 and 850 MHz bands. Due to the limited number of channels available, several efficient frequency utilization schemes have been devised.

- (1) 151 - 156 MHz
- (2) 450 - 470 MHz
- (3) 896 - 902 MHz
- (4) 935 - 941 MHz

Table 1.1-1

Frequency spectrum allocation for land
mobile radio in the 150 to 950 MHz band.

Figure 1.1-1

Hexagonal cell configuration.

Cellular mobile radio is one practical solution that is currently being used in many automobile radio telephone systems around the world. Such systems have been introduced with favourable reception in the United States (1982), Japan (1979), Sweden (1981), Canada (1985), as well as other countries. Cellular radio is a spectrum utilization scheme accommodating many users, efficiently achieved by high-density geographical co-channel reuse. The service area is broken up into a number of smaller areas, or cells (Figure 1.1-1) [3]. Each cell has its own base station for communication with mobile users in its own cell. The base stations are in turn connected to mobile switching centers which are in turn connected to the central exchanges of the wireline telephone network. Since each station transmits over a local area, it is possible to reuse the same frequency simultaneously in other cells of the service area. Cellular systems can accommodate more users by "cell-splitting"; however, each new cell site requires an additional \$25,000 U.S. to \$500,000 U.S. in equipment cost [4].

Current cellular systems make use of conventional FM analog-voice techniques. A further sixfold increase in spectrum efficiency could be realized by switching to narrowband transmission techniques. One example of a narrowband modulation technique is amplitude-companded

single-sideband (ACSSB) [5]. ACSSB compandors compress outgoing 30 kHz signals into 5 kHz and then expand the signal once they are received. However, the adoption of ACSSB transmission is unlikely in view of the entrenchment of the current FM-based systems.

However, there is a growing consensus that digital transmission techniques will eventually predominate. In March of 1985, the Public Telephone and Telegraph (PTT) administrations of the Federal Republic of Germany and France approved a proposal by the consortium SEL/AEG/SAT for an all-digital mobile telephone system in the 900 MHz band [6]. In their system, all voice channels will be transmitted on only one frequency in time-division multiple access (TDMA). A cellular radio approach will be used to manage the scarce bandwidth. They maintain that in comparison to the present analog systems, digital systems offer: a cost advantage of between three and five for the subscriber set and base station; and increased resiliency to multipath propagation effects of the mobile radio channel.

Digital voice seems inevitable in view of the need for integration of voice and data in mobile systems. In present systems, the baseband spectrum of the digital data signal must first be manipulated to make it look like an analog voice signal before injection into standard FM equipment. However, high bit rates of up to 9.6 kbit/sec can only be

achieved by employing multilevel modems at the expense of higher cost and high bit error rate. Practical bit rates for data using these analog techniques are actually around 4 kbit/sec.

Far more efficient digital modulation schemes exist whereby data bit rates as high as 20 kbit/sec can theoretically be obtained. Of these digital modulation techniques, time frequency modulation (TFM) [7] and Gaussian-filtered minimum-shift keying (GMSK) [8] hold the most promise.

Presently, the wireline telephone companies use log-companded pulse code modulation (PCM) for voice digitization. Log-companded PCM can produce very good quality speech at 64 kbit/sec. However, the bit rate requirement of a speech coder for mobile radio communication must be somewhat less than 64 kbit/sec. In fact, a bit rate of 16 kbit/sec or less combined with efficient modulation techniques, such as TFM, must be used if the conventional 25 kHz channel spacing is to be adhered to.

Recent advances in speech coding algorithms for mobile radio makes one believe that a 25 kHz channel spacing can be achieved using digital techniques. For example, a digital transceiver with 50 kHz channel spacing has been designed using 32 kbit/sec adaptive delta modulation (ADM) coding and

GMSK modulation [9]. The speech quality produced is very good even under severe transmission conditions. At the other end of the bit rate scale, a 2.4 kbit/sec linear predictive coder (LPC) has been used in conjunction with frequency shift keying (FSK) modulation to achieve 6 kHz channel spacing [10]. However, the LPC coder produces speech of unnatural (synthetic) quality and is severely degraded by mobile radio propagation errors.

Fortunately, several speech coding algorithms capable of transmitting good quality speech at 16 kbit/sec have been devised [11]. Although speech coders such as adaptive predictive coding (APC) [12] are more complex than ADM or LPC coders, they combine good quality speech production with a mild robustness to channel errors.

The importance of digital speech is growing due to the need for secure communications. Data encryption can be easily performed on digital data. There are also many higher-level mechanisms for combating the mobile radio multipath propagation problem that can directly benefit from using digital speech. Spread Spectrum is one such method [13]. Finally, a digital mobile radio communications service would also have the capability of "plugging into" the growing wealth of digital services networks such as ISDN, the integrated services digital network [14].

7

1.2 Thesis Objectives

The main objective of this thesis is to evaluate and compare several digital speech coding algorithms with transmission rates of less than or equal to 16 kbit/sec that could be suitable for use with mobile radio communication and the associated impairments of the channel. This objective is accomplished as follows:

1. Several efficient speech coding algorithms are investigated and three are chosen for study.
2. The chosen speech coders are simulated using a microcomputer.
3. Several mobile radio channel conditions are simulated from actual mobile radio experimental data in the 820 MHz band and applied to the speech encoder output.
4. The performance of the speech coders under the aforementioned conditions is evaluated using objective and subjective testing methods.
5. From this performance comparison, a recommendation can be made regarding a suitable digital speech coding method.

1.3 Thesis Organization

This thesis is divided into three main chapters plus a concluding chapter. Chapter two contains background

information on the propagation mechanism of the mobile radio channel and a review of several speech coding algorithms that satisfy the bandwidth and robustness requirements of the channel. The chapter is completed with the selection of three speech coding algorithms for further study: adaptive delta modulation (ADM), adaptive predictive coding (APC), and residually excited linear prediction (RELP).

Chapter three contains background information on the three speech coding algorithms. Final configurations of the three speech coders are developed based on the investigation and initial experimental work. The methods used to develop and test the speech coding simulations are also detailed.

In the fourth chapter, an evaluation of the three speech coding algorithms under various channel conditions is performed. The extraction of the mobile radio channel error patterns is first performed. These patterns, along with error-free and random error patterns, are simulated to derive decoded speech output. The speech output samples are gathered and subjective tests are carried out using volunteers. Next, the results of the objective (SNR performance) testing and subjective (mean opinion score) testing are evaluated, and a final recommendation of a speech coding algorithm suitable for the mobile radio channel is made.

Finally, in chapter five, a summary of the research is presented together with suggestions for future research in the related fields.

CHAPTER TWO

THE MOBILE RADIO CHANNEL, DIGITAL SPEECH CODING AND THE INITIAL INVESTIGATION

2.1 The Mobile Radio Propagation Mechanism and Impairments

The signal propagation mechanism in a mobile radio environment has two major characteristics: multipath propagation and relative movement between the transmitter and receiver. In general, no direct line-of-sight path exists between the transmitter and receiver in metropolitan environments because of obstructions by buildings and other terrain irregularities and thus, the signal reaches the receiver via numerous indirect paths. Figure 2.1-1 illustrates four primary modes of multipath propagation. The resulting received signal consists of a composite of many components with different amplitudes and phases. The amplitude of the received signal has been shown to have a Rayleigh distribution [15].

The mobile radio service environment generally consists of the base station, which is fixed and located near the center of the service area, and the mobile unit, which is free to wander around within the service area. A stationary vehicle will observe an almost time invariant signal, although occasional variations can occur due to

Figure 2.1-1

Primary modes of multi-path propagation.

environmental changes around the receiver. A moving vehicle will encounter a rapidly fluctuating signal due to multipath propagation. Relative movement between the base and mobile units also gives rise to channel impairments due to the effect known as "Doppler spread". Figure 2.1-2 illustrates a typical received signal envelope from a mobile radio environment. At several points in the diagram, the signal level drops down, or fades, to the level at which errors can occur from noise at the receiver's front end. The fade duration is generally longer than the duration of one bit interval, which causes error bursts of several bits in length to occur. There are also many errors that are caused by ignition noise from the vehicles in the street, including the user's vehicle unless it has been carefully suppressed. In the VHF and UHF bands, signal fading is far more important than the ignition noise as it is responsible for nearly 80% of the total errors in the channel. Figure 2.1-2 illustrates the relationship between the received signal fades and data bit errors as well as showing errors due to ignition noise.

The average error rate of the mobile radio channel can be relatively high in comparison to other channels because of the signal fading and ignition noise problems. The ignition noise problem can be solved through suppression but the signal fading is a more difficult problem to solve.

Figure 2.1-2

Relationship between received signal envelope and bit errors

Diversity transmission/reception is the most common method used to combat fading [16]. Diversity is the use of more than one "branch" of transmission or reception in the communication path, to safeguard against the failure of any single branch. Different paths can be used in frequency (frequency diversity), in space (spatial diversity), or time (temporal diversity). For example, spatial diversity could be achieved by using different sites or antenna positions to create a multiplicity of base-to-mobile paths.

Even with the use of diversity techniques, one could expect average error rates as high as 10^{-2} . However, during most periods, the average error rate could be expected to be around 10^{-3} or even 10^{-4} . Only, during specific periods in time, when the base station and mobile site are separated by a large distance, or the mobile site is stationary and situated in a deep fade, could the average error rate hover near one percent.

2.2 Digital Speech Coding

2.2.1 Requirements for Mobile Radio

A number of basic requirements can be listed for a suitable speech coding algorithm based upon the requirements of the mobile radio channel, described in the previous section and in chapter one. The first requirement is that

the output data rate of the speech coder must be less than or equal to 16 kbit/sec. because of the bandwidth limitations of the mobile radio channel. Also, the speech coder must be able to produce output speech of good, or telephone-toll quality, even with the presence of the added impairments associated with the channel. Thus, the coder has to be fairly robust to burst-type error occurrences, due to multipath fading, and high average error rates of up to one percent. Lastly, the speech coding algorithm should be relatively simple so that a future VLSI implementation of the coder could be realized.

2.2.2 Review of Digital Speech Coding Algorithms

.. Digital speech coding is a set of techniques that transforms an analog speech waveform to a digital representation of that signal. Speech signals contain considerable redundancy because of the physical mechanism of the human vocal tract and the way language is structured. Speech coding takes advantage of these redundancies in order to produce a lower-rate bit stream while still maintaining the key properties of the original speech signal. Speech coding techniques are generally classified into three types: waveform coding, source coding, and hybrid methods. A concise review of several coding methods from each category will now be presented.

Waveform coders, as the name implies, essentially strive for facsimile reproduction of the signal waveform [17]. Linear pulse code modulation (PCM) is probably the simplest waveform coder. PCM, uniformly quantizes the analog signal using a rectangular grid in time and amplitude. One variation on PCM, called μ -law PCM, uses a non-uniform, logarithmic scale quantization process. Good quality speech can be obtained at a bit rate of 64 kbit/sec. μ -law PCM is widely used in the telephony world where relatively cheap voice codecs, based on this algorithm, abound.

Differential waveform coders take advantage of the high sample-to-sample correlation property of speech, where efficiency is obtained by quantizing the sample-to-sample difference of the speech signal rather than the signal itself. Two popular differential waveform coders are deltamodulation (DM) and adaptive differential PCM (ADPCM). ADPCM offers telephone-toll quality speech at 32 kbit/sec and has been recommended by the CCITT as a standard for speech coders operating at this bit rate [18]. DM, essentially a one-bit per sample version of ADPCM, can produce reasonable quality speech down to 16 kbit/sec if adaptive step-size techniques are used. Continuously variable slope DM (CVSDM) is a member of this class, using syllabic companding of the quantizer step-size [19]. CVSDM

is simple and robust to channel errors, making it attractive for use with mobile radio. Current directions of research for waveform coders have focused on methods of efficiently encoding the difference signal. One method that is getting a lot of attention is that of vector quantization [20].

At the other end of the bit rate scale for speech coders are the source coders. Source coders, such as LPC vocoders, generally transmit in the range of 2.4 to 4.8 kbit/sec. The basic principle of the LPC vocoder is to approximate the speech production model [21]. Speech parameters defining the vocal tract filter, voiced/unvoiced/silence decisions, and pitch are analysed and encoded at the transmitter. Unfortunately, the performance of the vocoder is often talker-dependent and very sensitive to channel impairments. The output speech that it produces has a synthetic (less than natural) quality. Thus, vocoders will not be considered for this study.

The last set of speech coding techniques to be presented are the hybrid methods. As the name implies, these coders are a meld of techniques from both waveform and source coding.

The adaptive predictive coder (APC) extends the concept of differential coding to the time varying spectral structure of speech by using higher order differences or

adaptive prediction [22]. The resulting rms level of the difference signal is reduced substantially compared to that of the original speech. Prediction based on the past 4 to 20 samples of speech, is generally referred to as short-time prediction. Additionally, long-time prediction, due to correlations from pitch-period to pitch-period, can be utilized to further minimize the difference signal. The APC coder is a relatively complex algorithm, but has the capability of providing good quality speech at bit rates of 12 to 16 kbit/sec.

Sub-band coding (SBC) subdivides the speech signal into a number of separate frequency bands and independently codes these signals [23]. Quadrature-mirror band-pass filtering is the preferred method for separating the signal into the separate frequency bands. These signals are then downsampled to their Nyquist rate and coded using ADPCM. The frequency bands are allocated available bits according to their information content. Sub-band coding delivers good quality speech at bit rates higher than 14 kbit/sec. The complexity of the coder is moderate.

An improvement can be made over SBC by increasing the number of frequency bands and dynamically varying the number of bits per sample from one band to another. The filter bank is created effectively using a fast Fourier transform (FFT) algorithm. This method of speech coding is called

adaptive transform coding (ATC) [24]. ATC is capable of adapting dynamically to the spectral properties of speech by means of the bit allocation of the algorithm. LPC-derived coefficients are quantized and transmitted along with the FFT-derived information for synthesis at the receiver. ATC is, in effect, a frequency-domain counterpart to the time-domain APC coder. Both methods have a similar degree of complexity and performance.

Multi-pulse LPC is a hybrid coding technique that improves upon the basic LPC vocoder by avoiding the difficulties of the unvoiced/voiced decisions [25]. By varying the amplitude and temporal position of the excitation pulses used in the contemporary vocoder, improved speech quality is achieved at bit rates above 10 kbit/sec.

Residually excited LPC (RELPC) is another hybrid approach. The RELPC coder takes advantage of the residual speech information that the LPC process discards and produces a more robust, higher-quality output speech [26]. The analysis function derives the LPC parameters and extracts the baseband portion of the residual signal. The coder then transmits the LPC parameters along with an encoded, decimated version of the baseband signal. The receiver then artificially regenerates the full-band residual signal, typically by non-linear distortion of the baseband signal. The RELPC coder begins to produce

reasonable speech quality at bit rates above 4.8 kbit/sec.

Figure 2.2-1 illustrates the speech quality vs. bit rate performance of the three basic speech coding techniques. The upper left-hand corner on the diagram indicates the range in bit rate and speech quality where a suitable speech coding algorithm for mobile radio may be found. Thus, only some waveform coding and hybrid coding algorithms can be considered.

2.2.3 Recent Trends in Speech Coding

The references provided in the previous section for the various speech coding algorithms only mark the starting point for the research and implementations that have since followed. The follow-up to the initial ground-breaking research has usually been characterized by: first, more research, exploring improvements in all aspects of the original algorithm; followed by real-time implementations of the speech coders using commercial digital signal processor (DSP) devices; finally, customized VLSI implementations of the coder.

DSP processors generally have some or all of the following features: internal program and data storage; serial data input/output; main arithmetic unit for fast 16 bit-by-16 bit multiplication; instruction-execution cycle

Figure 2.2-1

Speech quality vs. bit rate for three different speech coding methods.

times of 200 ns. The low cost and high-versatility of DSP's has made them an ideal means for implementing a variety of speech coding algorithms. Recent trends in DSP chip design has seen an incremental improvement in performance over the first generation devices [27]. Thus, the hardware technology appears to be pushing the development of new speech coding algorithms.

Recent implementations of speech coders using commercial DSP devices include: an APC codec using three NEC 7720 DSP's [28]; a RELP codec using two TI 32010 DSP's [29]; a sub-band codec using two NEC 7720 DSP's [30]; a multi-pulse LPC codec using five NEC 7720 DSP's [31]; and an ADPCM/PCM transcoder using a single TI 32010 DSP [32]. Recent developments in VLSI realizations of speech coders include: a multi-pulse LPC synthesizer using an AT&T DSP-20 and a custom memory chip [33]; an ADPCM LSI codec [34]; a pattern matching processor implemented in VLSI for use, in combination with commercial DSP's, as a vector quantizer [35]; and an LPC vocoder chip set consisting of three VLSI chips and two LSI chips [36].

Based upon these recent trends in speech coding applications, it is apparent that it might soon be possible to realize an all digital mobile radio transceiver using custom DSP devices.

2.3 Selection of Suitable Speech Coding Algorithms

The requirements for a suitable speech coding algorithm were outlined in sub-section 2.2.1. It was decided the speech coder must have the following properties: bit rate of less than or equal to 16 kbit/sec; relatively low complexity; deliver good quality speech even in the presence of channel impairments associated with mobile radio. The coder complexity issue could be open to some subjectiveness; however, the number of multiplications and additions required to implement the algorithm, is generally regarded as a good measure of the coder complexity.

To measure coder performance and robustness to channel errors, it is necessary to compare, subjectively, the output speech of all of the candidate speech coding algorithms. Objective tests, although simpler to perform than subjective tests, are not as reliable since objective measures, such as the signal-to-noise ratio (SNR), only perform well for certain types of coders, such as waveform coders.

Fortunately, several reports on comprehensive subjective evaluations of speech coders are available. Two separate evaluations that are of interest to this study are those performed by Daumer [37] and Svean [11]. In his study, Daumer subjectively evaluated six speech coders of which four have relevance to this study. These four coders

were: ADPCM [38], sub-band coding [23], APC [12], and ATC [39]. His results show that 24 kbit/sec SBC, 25 kbit/sec ADPCM, and both ATC and APC at 16 kbit/sec all achieved subjective scores only slightly below toll-quality. At a subjective score indicating speech of "fair or better" quality were: 16 kbit/sec SBC, 16 kbit/sec ADPCM, and 12 kbit/sec APC.. The APC coder outperformed the ATC coder at similar bit rates.

In his study, Svean subjectively evaluated four speech coders; ATC, APC, SBC, and RELP. His results show that 16 kbit/sec APC, 16 kbit/sec SBC, and 9.85 kbit/sec RELP all achieved subjective scores indicating that the speech they produced was of "fair to good" quality.

In one study, the issue of complexity was addressed by proposing that the complexity of a speech coding algorithm is directly proportional to the number of DSP devices required to implement the algorithm [40]. A low-complexity design was defined as one in which one DSP device could be used. Examples of such algorithms are adaptive delta modulation (ADM) and ADPCM. On the other end of the bit rate scale were "high-complexity" coders such as ATC and APC which both required more than three digital signal processors. Sub-band coding was seen to be a "medium-complexity" coder.

Based upon the evidence provided by the studies carried out on subjective testing and the complexity issue, three speech coding algorithms were chosen for study. The coders chosen were: adaptive delta modulation coding (ADM), adaptive predictive coding (APC), and residually excited linear predictive coding (RELPE). The ADM coder was viewed as being a relatively simple algorithm to develop and quite robust to channel errors. The APC coder, although relatively complex, has been a proven high-performer, and was thus chosen. The RELPE coder, although being relatively new on the scene, was felt to be slightly less complex than APC, with the possibility of providing similar speech quality at a slightly lower bit rate.

CHAPTER THREE

BACKGROUND TO THE SPEECH CODING ALGORITHMS, THEIR DEVELOPMENT AND CONCURRENT TESTING

3.1 Developmental Laboratory Set-up

Figure 3.1-1 shows a block diagram of the laboratory set-up used to develop the speech coding simulations. A Texas Instrument Professional Computer (TI-PC) was used to write all of the speech coding software and store the experimental data. The experimental data consisted of: binary speech data files, mobile radio channel data files, speech coder parametric set-up files, and simulation result files.

The binary speech data files were created as follows. The speaker's voice was first recorded on analog tape (AMPEX-406) using a dynamic-type microphone (SHURE 560) and a reel-to-reel tape recorder (REVOX A77) set at 7.5 IPS. The tape was then played back, sent through a band-pass filter (Krohn-Hite 3202), analog-to-digital converted (TECMAR PC-MATE with 12-bit accuracy), and stored in RAM memory. The band-pass filter was used to anti-alias the tape playback before the sampling process. The use of 12-bit linear A/D conversion gives similar performance to using an 8-bit $\mu 255$ law codec with subsequent $\mu 255$ to 13-bit

Figure 3.1-1:
Block diagram of the developmental laboratory set-up.

linear conversion. Figure 3.1-2 lists several salient features of the equipment used in Figure 3.1-1. Using all of the available memory, it was possible to store up to 15 seconds of continuous speech, assuming a 8 kHz sampling rate.

Test speech sample files were created for use in the initial debugging and configuring of the speech coder simulations. Recordings were made of a speaker (male) with an average pitch period of 5.9 msec. The speaker was asked to say the following three sentences:

1. "The pipe began to rust while new."
2. "Add the sum to the product of these three."
3. "Open the crate but do not break the glass."

These sentences are well balanced including voiced speech, plosives, fricatives, etc. Three test speech sample files were created using the following sampling rates:

Test speech sample 1 - 6.666 kHz

Test speech sample 2 - 12.048 kHz

Test speech sample 3 - 16.128 kHz

The duration of the three sentences lasts about 7.4 seconds.

Texas Instrument Professional Computer (TI-PC)

- 256 Kbytes of RAM
- one 5 and 1/4 DS-DD floppy disk drive
- 5 Mbyte Winchester hard disk drive
- MS-DOS V. 3-13 operating system

TECMAR PC-MATE Lab Master

- 12-bit analog-to-digital and digital-to-analog conversion

Krohn-Hite Model 3202 Variable Bandpass Filter

- 20 dB per octave roll-off

Revox Model A77 Reel-to-reel Tape Deck

- 3.75 and 7.5 IPS recording speeds

Shure Model 560 Microphone

- moving coil (dynamic)
- omni-directional
- 40 to 10 000 Hz frequency response

Figure 3.1-2

Salient features of the developmental set-up

3.2 Development and Testing of the Speech Coding Algorithms

All of the speech coding simulation software was written in Microsoft pascal, with the exception of some low-level bit manipulation subroutines which were written in Microsoft assembler.

The speech coding simulator program takes an input ~~binary~~ speech file, and simulates the tandem operation of the transmitter, channel, and receiver to produce the output binary speech file. Figure 3.2-1 illustrates how this operation is carried out. The channel data is passed to the coder simulator as a data file indicating the bit positions that are in error. A data file consisting of various coder parameters is also passed to the program.

The speech coding structure was separated into a number of logical blocks and coded in software. For each block, separate programs were written to test the functionality of the algorithm used. Once tested separately, the algorithm would be incorporated into the final coder structure and undergo another round of testing.

For example, the linear prediction software used with the APC and RLP coder was taken from an existing subroutine by Gray and Markel [41]. Included with this software is a test program that can be used to test the autocorrelation subroutine that solves for the predictor coefficients. The

Figure 2.14-1

test program generates a test data sequence which allows one to check the solved coefficients against expected results.

The AMDF method of pitch period estimation is another logical block used in speech coding. A program was written to test the AMDF method with actual speech data. During voiced segments of the speech, the AMDF pitch period estimate remained fairly constant, proving that the algorithm functioned correctly.

A program was written to test the functionality of a digital lowpass filter, used for baseband extraction. The program performed the lowpass filter operation on input data files. By using a sinusoidal signal as an input data file, with its frequency close to the cut-off frequency of the filter, it was possible to confirm whether the passband and stopband of the filter functioned correctly.

Once an algorithm was tested separately, the algorithm would be incorporated into the speech coder simulation, and tested again. For example, after placing the autocorrelation-based predictor into the APC structure, the performance of the predictor was determined by calculating the prediction gain. It was found that the prediction gain closely approximated known results.

To monitor the performance of different configurations of the speech coders, the segmental signal-

to-noise ratio (SEGSNR) was used as an objective measure. For a full discussion concerning objective measurements of coder performance, refer to Appendix A. The SEGSNR method was chosen over the more time consuming method of subjective testing. The performance gain of adding a particular enhancement to a speech coder could be monitored using the SEGSNR measure. The effects of varying the different parameters of the speech coder could also be monitored using the SEGSNR measure. For example, the SEGSNR level was used to measure the coder performance as different values of the prediction order of the spectral predictor were employed. It was found that a prediction order of eight gave sufficient performance for the speech data used. The SEGSNR level, and thus the performance, did not increase appreciably for prediction orders greater than eight.

3.3 Coder 1 : Continuously Variable Slope Deltamodulation (CVSDM)

3.3.1 Introduction to Deltamodulation Coding

Deltamodulation (DM) coding is an important member of the class of differential waveform coders. Deltamodulation can be viewed as being a one-bit or two-level version of DPCM. This property of having a one-bit codeword eliminates the need for word framing at the transmitter and receiver. Thus, DM is attractive as a digital voice encoder because of its simplicity and robustness to channel errors. There are many variations of the DM encoder which have proceeded the original invention [42]. Different versions of DM can be effected by utilizing various prediction and quantization schemes.

Two important features of the DM coder are: signal oversampling and coarse quantization. Signal oversampling increases the prediction gain of the coder since neighbouring speech samples become more correlated. The perceptual effects of one-bit quantization forces one to consider using high bit rate coders and/or adaptive quantization schemes. At bit rates in the order of 32 to 48 kbit/sec, DM systems with adaptive quantizers produce reconstructed speech of very good quality [43]. However, speech coders with bit rates under 16 kbit/sec must be considered for mobile radio. Syllabically adaptive DM

systems can transmit communications quality speech at bit rates as low as 9.6 kbit/sec.

In section 3.3.2, a simple delta-modulation system will be introduced. As well, topics relating to DM systems such as quantization noise, adaptive quantization, etc., will be discussed. Section 3.3.3 will present a delta-modulation coder with syllabic companding of the quantizer step-size logic. This coder, known as continuously variable slope delta-modulation (CVSDM), is one of the three speech coding systems that were used during the final subjective testing portion of this study.

3.3.2 Basic Delta-modulation Coding System and Related Topics

In Figure 3.3-1, a simple Delta-modulation (DM) coder is shown. The coder uses first-order prediction. The DM block diagram is a special case of the more general DPCM block diagram except for the inclusion of an output low pass filter. The quantizer has two reconstruction levels of ± 1 . The output of the quantizer is sent straight to the digital channel. At the transmitter, the quantizer output is first multiplied by the step-size $\delta(n)$ before entering the prediction stage which is shown boxed in Figure 3.3-1. Non-adaptive or fixed step-size delta-modulation is also known by the name of linear delta-modulation (LDM). Adaptive

Figure 3.3-1

Block diagram of single-integration DM

deltamodulation (ADM) is a system where the step-size is a time-variable quantity. The predictor, in single-integration DM is a first-order device with predictor tap coefficient $h_1 \leq 1$. Figure 3.3-2 depicts an analog implementation of the single-integration DM coder of Figure 3.3-1 [44]. Note that the integrator in Figure 3.3-2 takes the place of the boxed portion in Figure 3.3-1. If $h_1 = 1$ then "ideal integration" is effected; in this case, $\hat{x}(n)$ is simply a delayed version of the previous output $y(n-1)$. If $h_1 < 1$, then "leaky integration" is effected which produces a more subdued predictor. However, the decrease in overall quality of the reconstructed speech is more than adequately compensated for by the increase in transmission error resistance.

Both digital and mixed analog-digital implementations are possible with the DM coder. However, only digital implementations of the DM coder will be discussed from this point on because of the ease of realizing such coders in simulation.

The function of the output filter will now be discussed. DM coders typically oversample the input speech $x(n)$ in lieu of the coarse one-bit coding of the quantizer. A quantity that relates the extent of oversampling is the oversampling ratio F defined as

Figure 3.3-2

Block diagram of an analog implementation of the single integration DM coder.

$$F = f_s / 2W$$

(3.3-1)

where f_s is the sampling frequency of the input speech and W is the bandwidth of the input speech. As a result of this oversampling, the unfiltered output speech $y(n)$ is characterized by quantization noise and distortion components that extend beyond the bandwidth of the input speech W . The output filter, a lowpass filter with corner frequency of W , rejects these out-of-band components between W and $f_s/2$.

The objective quality of the DM coder can be calculated using the segmental SNR measure. The noise component of the SNR can be calculated using either the reconstruction error $r(n)$, or the in-band reconstruction error $r_f(n)$. These are defined by

$$r(n) = x(n) - y(n) \quad ; \quad r_f(n) = x_f(n) - y_f(n)$$

where $x_f(n)$ and $y_f(n)$ are the results of filtering $x(n)$ and $y(n)$ respectively, with the output lowpass filter of Figure 3.3-1. The object of filtering $x(n)$ is to duplicate the delay introduced in filtering $y(n)$. Using $r_f(n)$ (and thus $y_f(n)$) to calculate the segmental SNR rather than $r(n)$ (and thus $y(n)$) makes sense because $y_f(n)$ is, in fact, the output speech. However, in practice, the signal $r(n)$ is much easier to use in calculations than $r_f(n)$. Thus, the signal $r(n)$ was employed to calculate the segmental SNR.

The reconstruction error $r(n)$ in deltamodulation coders can be separated into two different components: slope overload distortion and granular noise. Figure 3.3-3 illustrates LDM waveforms that result for different values of the step-size constant δ . In Figure 3.3-3(a), the encoder is correctly tracking the input $x(n)$. The step-size ($\delta=5$) is matched to the slope characteristics of the input $x(n)$. If the input slope statistics are not known, waveform encoding in LDM will be less accurate. The arbitrarily selected step-size could be either too small, as in the slope overload example of Figure 3.3-3(b); or too large as in the granular noise example of Figure 3.3-3(c). Even if the input speech statistics are known over a long period of time, it will still be impossible to match the step size to the local input slope at all times. - The step-size is matched well for the well-behaved waveform of Figure 3.3-3(a), but will not be matched well for most other random waveforms.

The slope tracking problem with random waveforms can be diminished if adaptive deltamodulation (ADM) is employed. ADM coders allow the step-size to follow the variations in the input slope in an adaptive manner. There are two classes of ADM coders that are based on how fast the input slope is tracked. Slowly adaptive or syllabically adaptive coders estimate the average slope of a waveform segment that

Figure 3.3-3

LDV waveform examples. In (a) the step size δ is well matched to the input slope. The step size in (b) is too small for the input slope. In (c) the step size is too large for the input slope. (After [43].)

is sufficiently well-behaved to be considered stationary. A step-size is chosen such that the input waveform is optimally tracked. With instantaneously adaptive coders, the attempt is to track the instantaneous or local input slope by sudden changes in the step-size.

The two types of error distortions presented above, namely, slope overload distortion and granular noise, have different perceptual annoyance values. Slope overload distortion is often less perceptible than granular noise in the reconstructed output speech. This is so because, slope overload distortion errors tend to be input-correlated whereas granular noise errors tend to be input-uncorrelated [45]. The spectra of granular noise tends to be more heavily weighted towards the high frequencies (highpass). Although the out-of-band granular noise will be rejected by the output filter, there will still be perceptible granular noise in the high in-band frequencies, near W , where the input signal power is low.

For an LDM coder, the step-size δ can be varied such that an equal increase in either slope overload distortion or granular noise can be produced. The SNR will drop more significantly for the case where the amount of slope overload distortion is increased over the case where the amount of granular noise is increased. This occurs because slope overload distortion contributes more heavily to the

total error variance. However, as mentioned earlier, granular noise is more perceptible than slope overload distortion and so the SNR measure fails as a subjectively-meaningful objective measure of the DM coder's performance. The segmental SNR is a more useful measure because it compensates for the under-emphasis of weak signal performance of the conventional SNR calculation. In using the segmental SNR calculation to derive a final implementation of a DM coder, it is hoped that the correct balance can be found between slope overload distortion and granular noise to produce reconstructed speech of optimal quality.

The emphasis in DM coding systems is on simplicity, especially with the predictor subsystem which is almost always non-adaptive. One elementary variation on the single integration type predictor that was introduced in Figures 3.3-1 and 3.3-2 is a system where two integrators take the place of the single integrator. The basic principle of double integration DM (DM-DI) is to encode the input waveform using a difference signal $d(n)$ that follows the second derivative rather than the first derivative, as in single integration DM (DM-SI). By cascading two integrators in place of a single integrator, the DM-DI system can track an input waveform much faster than a DM-SI system, implying that both slope overload distortion and granular noise can

be lessened.

The benefits of the higher order prediction of DM-DI coders become negligible or negative for very low sampling rates. In an experimental study, it was found that DM-DI coders outperform DM-SI coders only at sampling frequencies above 18 kHz [46]. Since only coders with bit rates below 16 kbit/sec are allowable for this study, DM-DI systems were ruled out.

To produce good quality delta modulated speech at bit rates less than 16 kbit/sec it is necessary to use adaptive algorithms for the prediction error quantizer. Most of these adaptive algorithms involve varying the step size of the quantizer according to some rule. The step size variation can either be instantaneous or syllabic. Instantaneous step-size variation is characterized by average step size changes of one to several dB per waveform sample. Syllabic step size variation is characterized by average step size changes of a fraction of a dB per waveform sample [47]. Properly designed instantaneous ADM coders can track speech waveform variations smoothly; however, practical ADM speech coders use syllabic adaptation of the step size because these systems have the additional feature of being more robust to transmission errors. Since the average error rate of a typical mobile radio channel can reach one percent, only ADM coders using syllabic adaptation

of the step size were considered. Continuously variable slope delta modulation (CVSDM) is a class of ADM coders employing syllabic companding of the step size. The algorithm used to change the step size is based on observing the most recent (3 or 4) bits on the channel.

After studying several ADM speech coders using syllabic companding of the quantizer step size, the CVSDM speech coder was chosen as the preferred DM system because of its relative simplicity and ease of digital realization. In the next section (3.3.3), the CVSDM coder will be described in more detail.

3.3.3 Continuously Variable Slope Delta Modulation (CVSDM)

Continuously variable slope delta modulation (CVSDM) coders are a class of ADM coders that vary the step size parameter at a rate on the order of a pitch period (5 to 10 ms). A block diagram of the CVSDM coder is shown in Figure 3.3-4. The diagram illustrates how an overload detector is used to adjust the step-size of the coder. If the last three bits on the channel $b(n)$ are either all +1 or all -1, then an overload condition has been detected and the output of the detector is V_0 . When no overload condition has been detected, then the output of the detector is zero. Figure 3.3-5 details the logic of the overload detector. Figure 3.3-4 also illustrates how "leaky" integrators I_1 and I_2 are

Figure 3.3-4

Block diagram of the CVSD : (a) encoder and (b) decoder.

Figure 3.3-5

Overload detector logic

used in both the encoder and decoder. Integrator I_1 is the approximation signal integrator while integrator I_2 is the step-size integrator. The function of the integrator can be implemented digitally by the use of a linear approximation equation. The equations used to update the step size $\delta(n)$ are given by:

$$\begin{aligned} \text{Overload: } \delta(n) &= \beta \delta(n-1) + (1-\beta)(V_0 + V_{\min}) \\ \text{No overload: } \delta(n) &= \beta \delta(n-1) + (1-\beta)V_{\min} \end{aligned} \quad (3.3-2)$$

where n indicates the n th sampling instant, β is the step-size update constant, V_0 is the output gain of the overload detector when an overload condition is detected, and $V_{\min}$ is a constant which helps define the minimum step size ($\delta_{\min}$). An overload condition signals a step-size increase of

$$(1-\beta)(V_0 + V_{\min}) = \delta_0 + \delta_{\min}$$

For no overload conditions, the step size decreases at a rate given by β , but never below a positive step size of

$$(1-\beta)V_{\min} = \delta_{\min}$$

The step-size update constant (β) can be derived from the time constant (t_c) of the step-size integrator, and the sampling frequency (f_s) as follows:

$$\beta = \exp \left\{ \frac{-1}{t_c \times f_s} \right\} \quad (3.3-3)$$

Thus, the rate at which the step size is updated depends primarily on the value of the step-size update constant. Step-size adaptations are syllabic when the value of β is close to 1, and when the gain δ_0 (during step size increases) is a small fraction of the maximum value $\delta_{\max}$. If, δ_0 is sufficiently large, then step-size increases can be relatively rapid, and the result would be an adaptation logic that can be characterized as instantaneous for step size increases, and syllabic for step size decreases. This arrangement has perceptual advantages in speech coding especially in the context of transmission errors.

Using "leaky" integration also ensures that the step size is upper-bounded by a maximum step size value ($\delta_{\max}$). This maximum value can be derived from equation 3.3-2 and is given by

$$\delta_{\max} = V_0 + V_{\min} \quad (3.3-4)$$

Ensuring that the step size is upper bounded helps to cope with stability problems that could arise at the receiver due to transmission errors on the channel. The value of $\delta_{\max}$ also controls the amount of slope overload distortion, while the value of $\delta_{\min}$ controls the amount of granular noise.

Another important quantity is the maximum step-size ratio M , which is given by

$$M = \delta_{\max} / \delta_{\min} \quad (3.3-5)$$

A typical value of M is on the order of 200 [19], with smaller values being appropriate in the context of transmission errors.

An approximation signal $y(n)$ is generated in the encoder feedback loop and at the n th sampling instant it is given by

$$y(n) = a y(n-1) + (1-a) H \cdot b(n-1) \delta(n-1) \quad (3.3-6)$$

where a is the approximation signal update constant, H is the approximation gain constant, $b(n)$ is the output on the digital channel (± 1), and $\delta(n)$ is the step size constant. The approximation signal $y(n)$ is upper-bounded by the value

$$y_{\max} = H \delta_{\max}$$

Thus H controls this upper bound. The approximation signal update constant a can be calculated using equation (3.3-2) with the time constant (t_c) set equal to the time constant of the approximation signal integrator (I_1). The constant a controls the rate at which the approximation signal $y(n)$ is updated.

The receiver attempts to reconstruct the transmitter's approximation signal $y(n)$ using only the received digital channel data $b(n)$ and the global signal

update equations. However, the received data could differ from the transmitted data $b(n)$ due to channel noise thus, making the reconstructed signal $y'(n)$ different from the desired signal $y(n)$.

The CVSDM coder uses one-bit coding and is therefore not as susceptible to bit errors as frame-oriented speech coders such as APC. Although the CVSDM coder is robust to channel errors, it is still not as robust to errors as a non-adaptive DM coder because, in addition to affecting reconstructed output $y'(n)$ calculations, channel errors also affect the step size δ' calculations. In fact, because of the inherent three-bit memory of the overload detector, single-bit errors will affect, directly, three step size calculations. Also, the use of backward estimation techniques (using channel bit data instead of input speech) to calculate the step size changes implies that errors will tend to propagate. However, the error effects on the calculations will vanish with time due to use of "leaky" integration.

This concludes this section on the CVSDM coding system. The next section (3.3.4) will detail the work that was carried out to implement two versions of the CVSDM coder.

3.3.4 Final Configurations of the CVSDM Coder.

The previous section introduced the necessary constants and equations needed to define a CVSDM coder. This section will detail the work that went into developing two CVSDM coder configurations, using the required equations.

The first step in deriving the CVSDM coders was to decide what sampling frequencies should be used. The maximum allowable bit rate for this study was defined earlier as 16 kbit/sec. It was decided that two sampling rates of 12 and 16 kHz would be used. For the output filter of the CVSDM coders, an analog low pass filter with a cut-off frequency of 3.3 kHz was used. This was the same type of filtering used for the anti-aliasing filters of the APC and RELP speech coders. Thus, the oversampling ratios of the two CVSDM coders are 2.4 (16 kbit/sec coder) and 1.8 (12 kbit/sec coder) - relatively low by DM standards.

The next two parameters that were derived were the update constants for the two signal integrators. These constants will be calculated for the 16 kbit/sec CVSDM coder. Using equation 3.3-3, and assuming a step-size integrator time constant of 5 ms, the step size update constant is

$$\beta = \exp \left\{ \frac{-1}{16 \times 10^3 \cdot 5 \times 10^{-3}} \right\} = 0.99$$

A 5 msec time constant provides syllabic or pitch adaptation of the step size δ . Using a time constant of 1 ms for the approximation signal integrator, the approximation signal update constant a can be calculated to be 0.94. This produces a faster rate of adaptation for the approximation signal. The values of β and a for the 12 kbit/sec coder were determined to be 0.98 and 0.92, respectively, using the same time constant data.

The last three parameters of the CVSDM coder to be set to their final values are: V_0 , $V_{\min}$ and H . Recall that the parameters V_0 and $V_{\min}$ determine the minimum and maximum step size, while H determines the upper bound of the approximation signal $y(n)$. Initially, V_0 and $V_{\min}$ were set so as to produce a maximum step size ratio of $M=200$. The parameter H was adjusted to keep $y_{\max}$ within the numerical accuracy limits of the computer.

Simulations were run using various values for the three parameters just described. The aim was to produce two CVSDM coders exhibiting the maximum achievable segmental SNR for the speech data used. The speech data used for this process were speech sample two (sampled at 12.048 kHz) and speech sample three (sampled at 16.128 kHz).

Table 3.3-1 outlines a small portion of the results obtained from the simulations of the two CVSDM coders. Both coders achieve their highest segmental SNR ratio for simulation number four. The parametric values used for simulation four were assigned to the final versions of the two CVSDM coders. Details of the simulations of the CVSDM coders with mobile radio channel error patterns are outlined in chapter four.

(a)

Simulation Number	V min	V max	H	Segmental SNR (dB)
1	12	5500	5.0	5.55
2	10	6000	5.0	5.58
3	16	6000	5.0	5.60
4	16	6000	5.0	5.66
5	16	6000	6.7	5.57
6	18	6000	5.0	5.56
7	12	6500	5.0	5.51

(b)

Simulation Number	V min	V max	H	Segmental SNR (dB)
1	10	4500	5.0	6.51
2	8	5000	5.0	6.51
3	10	5000	3.3	6.35
4	10	5000	5.0	6.59
5	10	5000	6.7	6.55
6	15	5000	5.0	6.48
7	10	5500	5.0	6.42

Table 3.3-1

Results of the initial simulations to determine the UM speech coder with the optimal segmental SNR for sampling rates of: (a) 12.0 kHz and (b) 16.1 kHz.

3.4 Coder 2 : Adaptive Predictive Coder with Variable Length Coding of the Residual

3.4.1 Introduction to Adaptive Predictive Coding

The method of adaptive predictive coding (APC) of speech has been investigated extensively since its introduction fifteen years ago [22]. The APC coder can transmit good quality speech at bit rates of about 16 kbit/sec. [28]. The costs of achieving good quality speech compared to previous speech coding techniques such as delta modulation (DM) coding, are increased complexity and increased susceptibility to channel errors.

However, the recent availability of more sophisticated and less expensive digital hardware, coupled with the use of error detection and correction techniques, has made the APC system more attractive as a solution to the medium bit rate, telephone-quality speech problem.

The succeeding sub-sections will introduce adaptive predictive coding, the associated concepts and structures, and the preliminary results from the simulations. Lastly, a final implementation will be derived.

3.4.2 A Basic Adaptive Predictive Coding System

The delta modulation coder, discussed in section 3.3, is a simplified version of a differential PCM (DPCM) coder.

using one-bit quantization and a first-order predictor. Adaptive predictive coding attempts to exploit further redundancies in the input speech signal by using adaptive quantization and an adaptive Nth-order linear predictor.

Figure 3.4-1 illustrates the structure of the coder and decoder for a basic open-loop predictive coding system. This coding system is impractical because the decoder has no way of tracking the input speech waveform. In order that the coder and decoder parts track (and reconstruct) the input waveform in synchronism, it is essential that the predicted signal $\tilde{s}(n)$ depend on previous quantized values $\hat{s}(n)$ rather than on previous unquantized inputs $s(n)$. Incorporating this requirement leads to the closed-loop predictive coding structure of Figure 3.4-2. The operation of the closed-loop predictive coder will now be summarized.

At the transmitter, the input speech signal is sampled to produce a sequence of sample values $s(1)$, $s(2)$, ..., $s(k)$, The predictor $P(z)$ forms a linear estimate $\tilde{s}(n)$ of each of these sample values based on previously reconstructed output speech samples $\hat{s}(n)$. The estimate $\tilde{s}(n)$ is subtracted from the input speech sample $s(n)$ and the resulting difference, the closed-loop prediction error $e(n)$, is quantized and transmitted along the digital channel to the receiver as $\hat{e}(n)$. The quantized, closed-loop prediction error $\hat{e}(n)$ is added back to the

(a)

(b)

Figure 3.4-1

Block diagram of the basic open-loop predictive coding system : (a) coder and (b) decoder.

(a)

(b)

Figure 3.4-2

Block diagram of the closed-loop predictive coding system : (a) coder and (b) decoder.

predicted value $\tilde{s}(n)$ to form the next reconstructed output sample $\hat{s}(n)$.

At the receiver, the same reconstruction loop used in the transmitter is used to derive the output samples $\hat{s}(n)$. Referring to Figure 3.4-2(b), it is readily seen that, in the absence of channel errors, the coder output is given by

$$\begin{aligned}\hat{s}(n) &= \tilde{s}(n) + \hat{e}(n) \\ &= \tilde{s}(n) + s(n) - \tilde{s}(n) + \delta(n) \\ &= s(n) + \delta(n)\end{aligned}\tag{3.4-1}$$

where $\delta(n)$ is the quantizer error (difference between the output and the input of the quantizer). Thus, the output speech sample from the predictive coder is equal to the input speech sample plus the quantizer error. Therefore, the design of an efficient quantizer is of utmost importance in order to achieve quality output speech. More will be said on the design of the quantizer in section 3.4.5.

3.4.3 Prediction of Speech

Linear predictive analysis is one of the most powerful speech analysis techniques. Linear predictive analysis has become important because of its ability to provide extremely accurate estimates of speech parameters, yet it has a relatively fast computation speed.

The basic idea behind linear predictive analysis is that a speech sample can be approximated by a linear combination of past speech samples. If the sum of the squared differences between the actual speech samples and the linearly predicted ones is minimized over a finite interval of samples, then a unique set of predictor coefficients can be determined. These predictor coefficients are the weighting coefficients used in the linear combination.

To solve for the linear prediction coefficients, it will be necessary to consider, again, the prediction of speech from input samples as in Figure 3.4-1. The predicted speech is

$$\tilde{s}(n) = \sum_{k=1}^p a(k) \cdot s(n-k)$$

where $\{a(k)\}$ are the set of p weighting coefficients and $s(\tilde{n})$ are the past input speech samples. The open-loop prediction error is defined as

$$d(n) = s(n) - \tilde{s}(n)$$

The set of p predictor coefficients $\{a(k)\}$ can be determined by minimizing the total squared open-loop prediction error over a short segment N of the input speech. The total squared error is defined as

$$\alpha = \sum_m d^2(m)$$

The range of summation is temporarily left unspecified. Minimization of α is obtained by setting the partial derivative of α with respect to $\{a(k)\}$ to zero. The result is a set of p equations in p unknowns which can be solved for the unknown predictor coefficients. There exist two main methods of solving for the coefficients; the autocorrelation method (from which derive efficient lattice methods) and the covariance method. Details of the derivation of these solution methods are outlined extensively by Rabiner and Schafer [48]. The two prediction methods will be compared below on such issues as quality of prediction, complexity, computational requirements, and stability.

The quality of prediction for the two methods is about the same. However, for the autocorrelation method, the prediction error can be quite large at the beginning of the segment and so usually a window, which tapers the speech segment to zero, is used.

The complexity of the two solution methods is similar and consists of a number of correlations being performed on the input data followed by the solution for the predictor coefficients.

The computational requirements of the two solution methods involve two major issues. These issues are the amount of storage and the number of multiplications. The autocorrelation method is superior in these two categories, requiring somewhat less storage and fewer multiplications than the covariance method.

Another consideration in comparing the covariance and autocorrelation methods is the stability of the synthesis filter $H(z)$

$$H(z) = \frac{1}{1 - P(z)}$$

where $P(z)$ is the linear prediction filter. The synthesis filter is stable if all of its poles lie strictly inside the unit circle in the z -plane. The poles of the filter $H(z)$, are the zeros of the denominator polynomial $1 - P(z)$, where

$$1 - P(z) = 1 - \sum_{k=1}^P a(k) \cdot z^{-k}$$

For the autocorrelation method, all the roots of $P(z)$ lie inside the unit circle, and thus, $H(z)$ is stable. For the covariance method, the stability of the predictor polynomial cannot be guaranteed.

All of the above issues were weighed in deciding which prediction solution method to use. The final choice was to

use the autocorrelation method because of its ability to produce a stable filter for the system function.

As discussed earlier in this section, speech is predicted on a segmental basis, where a segment is made up of N contiguous samples. For adaptive predictive coding (APC) the predictor coefficients for the filter $P(z)$ are reset every N samples, thus making the predictor adaptive. It has been shown for the autocorrelation method, that N must be large enough to encompass several pitch periods to ensure reliable results [49]. The size of N , in terms of samples, will depend on the sampling rate used.

Another parameter which depends on the sampling rate is the number of predictor coefficients p . The speech spectrum is generally believed to have an average density of two poles (one complex pole) per kHz due to the vocal tract contribution. An additional three to four poles is required to adequately represent the source excitation spectrum and the radiation load [48].

The first task in implementing the adaptive predictive coder was to set up the predictor and the values for the segment length N , prediction order p , and the size of the windowed segment K . The overlapping window method was effected using a Hamming window to taper the speech samples towards zero at the beginning and end of the segment. The

autocorrelation method was effected using the software procedure AUTO written by Gray and Markel [41].

The prediction gain was used as a measure of the predictor performance while the parameters N, M and K were varied. The prediction gain G_p , expressed in decibels is

$$G_p(\text{dB}) = 10 \log_{10}(\sigma_s^2 / \sigma_d^2)$$

where σ_s^2 is the variance of the input speech $s(n)$ and σ_d^2 is the variance of the open-loop prediction error $d(n)$.

The block diagram of Figure 3.4-1 can be used to derive the prediction error $d(n)$. The speech used in the study was test sample number one. This is a nine second section of speech from a male speaker with an average pitch period of 5.9 msec, sampled at 6.666 KHz and low-pass filtered to 3.333 kHz. The segment size N was varied from 80 to 240. The prediction order p was varied from 8 to 16. The size of the windowed segment was varied from 100% of N (no windowing) to 200% of N, the segment size.

Using the set-up above, it was possible to obtain prediction gains on the order of 13.5 dB which compares favourably to known results. In a previous study, Noll was able to obtain prediction gains of almost 14 dB using similar speech data [50]. A relatively simple predictor was able to produce prediction gains on the order of 12 dB.

This simple set-up used no windowing ($k=100\%$), and a low prediction order ($p=8$). The segment length N for this case was 180 samples. Thus, quality was not sacrificed significantly in pursuit of a simplified system.

3.4.4 Additions to the APC Structure

There are a few additions to the basic APC structure that have been known to enhance the overall quality of the speech produced by the coder. The price of such embellishments is an accompanying increase in complexity or increase in overall bit rate. One must also consider how well these additions will work when applied to the mobile radio channel and its associated impairments.

A few of these additions will be considered in this section including: pre-emphasis of speech, noise spectral shaping, and pitch-prediction. The results of their addition to the basic APC system will be delayed until section 3.4.6 as the interaction with the quantization of the prediction residual and channel errors are taken into account.

Pre-emphasis of Speech

Pre-emphasizing the input speech can be effected using an approximate differencing filter of the form

$$R(z) = 1 - \mu z^{-1}$$

where μ lies between 0.0 and 1.0. The value of μ can be held constant or it can be varied on a segmental basis according to some strategy [49]. The location of the filter at the transmitter is shown in Figure 3.4-3. Also shown, is the location of the inverse, or de-emphasizing filter at the receiver. Pre-emphasizing the speech flattens its spectral dynamic range and thus reduces the power gain (sum of the squares of the predictor coefficients) of $P(z)$. By reducing the size of the predictor coefficients, limit cycle effects can almost be eliminated [51]. Limit cycle effects can occur in the APC coder because the predictor, although derived from the original speech signal, is used to predict on the basis of reconstructed speech samples which contain an appreciable amount of quantization noise. This unaccounted for quantization noise can be enough to send the output of the predictive filter into an unstable mode, or "limit cycle".

The output speech after de-emphasis may exhibit low-frequency roughness. This is the price to be paid for the benefits of pre-emphasis.

Figure 3.4-3 Block diagram of speech pre-emphasis.

Noise Spectral Shaping

The decoder output speech $\hat{s}(n)$ is given by equation 3.4-1. If the quantizer in Figure 3.4-2 is designed such that the quantization error $\delta(n)$ is mostly granular or round-off noise then the spectral envelope of the quantization noise will be flat. Thus, the perception of noise in the decoder output speech $\hat{s}(n)$ will occur in frequency regions where the energy of the input speech signal $s(n)$ is low (the interformant regions), while the noise will be masked in other frequency regions where the input speech signal energy is high (the formant regions).

Two variations of APC coding which can be used to shape the spectral envelope of the quantization noise are open-loop APC and APC with noise shaping (APC-NF). Open-loop APC was discussed earlier in section 3.4.2. This configuration shapes the quantization noise such that the envelope of its spectrum is identical to the input speech, producing a constant signal-to-quantization noise ratio across all frequency regions. The speech produced is perceptually less noisy.

The open-loop APC configuration would be a good choice if our ears were equally sensitive to quantizing distortion at all frequencies. However, there is some evidence that our perception of quantizing distortion is more acute at low

frequencies [12] and so it seems that something in between the open-loop and closed loop APC systems is desirable. Noise feedback coding (APC-NF), illustrated in Figure 3.4-4, attempts to re-shape the spectral envelope of the quantization noise such that the quantization noise is masked by the input speech signal in the upper frequency regions, as with the open-loop APC system, while at the same time provide a better signal-to-quantization noise ratio in the lower frequency regions. This frequency response is achieved by setting the filter coefficients of $F(z)$ to be a frequency-weighted version of the predictor coefficients of $P(z)$ [12].

Pitch Prediction

Voiced speech segments exhibit quasi-periodicity, meaning that over a limited length of time the speech appears to be periodic. Quasi-periodicity still exists, to a lesser extent, in the prediction error signal $d(n)$, formed by subtracting the prediction signal $\hat{s}(n)$ from the input speech sample $s(n)$. This occurs because the prediction method uses only a finite number of adjacent or neighboring samples next to the speech sample to be encoded. The near-sample-based prediction methods do not utilize the fact that samples one pitch period away from the sample to be encoded exhibit high correlation. The result of a second stage of

Figure 3.4-4 APC system with Noise Feedback Coding filter $F(z)$

distant-sample-based prediction can result in a prediction error which is less periodic, or redundant in appearance.

Prediction of the open-loop prediction error signal $d(n)$ normally utilizes from one to three samples centered one pitch period away from the sample to be predicted. Using z -transform notation we can express this prediction, known as "pitch-prediction", as

$$P_d(z) = b_1 z^{-M+1} + b_2 z^{-M} + b_3 z^{-M-1} \quad (3.4-2)$$

where b_1 , b_2 , and b_3 are the pitch-predictor coefficients, and M is the pitch period. A block diagram of a closed loop APC system incorporating near-sample-based predictor $P_s(z)$ and distant-sample-based (pitch) predictor $P_d(z)$, is shown in Figure 3.4-5. It is necessary to use a higher order pitch predictor as the sampling rate is lowered. This is because the sampling frequency is unrelated to the pitch period and so individual samples do not show a high period-to-period correlation. The third-order pitch predictor provides an interpolated value with a much higher correlation.

The pitch period M can be determined using any one of the many pitch-measurement algorithms that have been discussed in speech literature [52]. Unlike vocoder techniques, the success of the pitch predictor with the APC system does not depend critically on accurate pitch detection. For example, it does not matter if the pitch

Figure 3.4-5

Closed loop APC system showing the location of the near-sample based predictor $P_s(z)$ and the far-sample based (pitch) predictor $P_d(z)$.

detector calculates the pitch period that is a multiple of the actual pitch period. A better prediction might actually be provided by the samples surrounding $x(n-2M)$ rather than $x(n-M)$.

Two simply implemented approaches to pitch period measurement that are adequate for pitch prediction are; the autocorrelation-based method and the average magnitude difference function-based (AMDF) method.

The autocorrelation method of pitch period measurement performs a short-time autocorrelation on a short segment of the prediction error signal $e(n)$ that contains at least a couple of pitch periods, making allowance for very low pitch period speakers. It then assigns the pitch period to be the lag at which the autocorrelation function is a maximum (the peak at lag=0, and an arbitrary number of lags after zero are ignored in the calculation).

The AMDF method of pitch measurement is based upon the idea that for a truly periodic signal $x(n)$ of period P , the sequence

$$y(n) = x(n) - x(n-k)$$

would be zero for $k = 0, \pm P, \pm 2P, \dots$. For short segments of voiced speech it is reasonable to expect that $d(n)$ will be small at multiples of the period. The pitch period is the

lag at which the AMDF function is minimum.

The AMDF method of pitch period extraction has the advantage that no multiplications (except possibly scaling) are performed, thus saving computation time.

The coefficients b_1 , b_2 , and b_3 are determined by minimizing the ~~mean-squared~~ prediction error between the prediction error signal $e(n)$ and the pitch predicted signal. The minimization process produces a series of simultaneous equations in three unknowns.

The application of a pitch predictor to the APC system which already utilizes a near-sample-based predictor can typically provide an additional prediction gain of 3 dB on top of the 10 to 13 dB provided by the first predictor [43].

Pitch prediction performs better with voiced speech than unvoiced speech as would be expected. For voiced speech, the pitch predictor attempts to cancel the large residual values that are due to pitch pulses. For unvoiced speech, where the coefficients are close to zero, pitch prediction has little effect. There can, however, be problems if the analysis frame includes a voiced-unvoiced or unvoiced-voiced transition. With these speech transitions, the pitch predictor could allow large residual errors to occur, producing problems with the residual quantizer where clipping errors will begin to propagate. These clipping

errors give the output speech a reverberant quality which is unacceptable [53].

3.4.5 Encoding the Predictor Parameters

The adaptive predictive coder sends predictor information along the digital channel to be used by the receiver in reconstructing the original speech. If both types of prediction are to be used then there are three sets of parameters that must be encoded: the near-sample-based predictor coefficients, the pitch predictor coefficients, and the pitch period. This information, along with the rms value of the open-loop prediction residual, is updated every segment and sent along the digital channel to the receiver.

Direct quantization of the near-sample-based predictor coefficients $\{a(k)\}$ is not advisable because round-off and truncation errors may cause some of the poles of the system function $H(z)$ to lie outside the unit circle, resulting in stability problems [54]. In solving for the predictor coefficients $\{a(k)\}$, the autocorrelation method calculates intermediate coefficients called partial correlation (parcor) coefficients or reflection coefficients $\{r(k)\}$. The reflection coefficients have the property that for a stable system function $H(z)$, they will have the following bounds:

$$|r(k)| \leq 1 \quad \text{for all } k \quad (3.4-3)$$

If these coefficients are used in the quantization process then a stable system function $H(z)$ will be produced if condition 3.4-3 holds true after quantization. Efficient predictor solution methods have evolved from the autocorrelation method, called lattice methods, whereby the reflection coefficients are solved for directly. A lattice-type synthesis filter is used with the reflection coefficients to derive output speech.

It is preferable to transform the reflection coefficients first before quantization because of the non-uniform distribution of the coefficients probability distribution function. The inverse sine, the inverse hyperbolic tangent, and the log area ratio transformations are all examples of such transformations. The most commonly used transformation is the log area ratio defined as

$$c(k) = \log[1 + r(k)]/[1 - r(k)]$$

where $\{c(k)\}$ are the log area coefficients. The resulting log area coefficients are then uniformly quantized. An inverse transformation would be performed at the receiver to recover the reflection coefficients.

To ensure efficient quantization of the log area coefficients, the minimum and maximum limits of the quantizer are selected to include 99.6 per cent of the entire range of the coefficients' variations. This can

be approximated by assuming that the log area coefficients are Gaussian-distributed. To achieve this approximation, the mean and standard deviation for each of the log area coefficients is calculated and the maximum and minimum quantizer limits are set equal to plus and minus, respectively, three standard deviations away from the mean. Using this method, the quantizer limits were derived from test speech sample number one, described in section 3.2. Table 3.4-1 lists the minimum $[\min(k)]$ and maximum $[\max(k)]$ quantizer limits for the quantizer used to quantize the first eight log area coefficients. The limits of column one are derived from the speech in an unprocessed form. Column two lists limits derived from speech which has been pre-emphasized slightly ($\mu=0.4$). Table 3.4-2 lists the number of bits $b(k)$ that have been assigned to each quantized log area coefficient. These bit assignments are taken from results published by Atal [51]. He determined that the prediction error, after quantizing the log area coefficients, is less if more bits are assigned to the lower order coefficients than the higher order coefficients.

Informal listening tests showed that the previously described method for the quantization of the log area coefficients did not produce any significant additional distortion in the output speech.

Coefficient Number	Min	Max
1	3.0354	-0.4434
2	0.8096	-2.0260
3	0.9063	-0.9372
4	0.4167	-0.6400
5	0.4317	-0.4763
6	0.3461	-0.3896
7	0.3118	-0.4396
8	0.4208	-0.3468

Coefficient Number	Min	Max
1	3.0800	-0.7860
2	0.8871	-1.8340
3	0.8518	-0.9048
4	0.5130	-0.7859
5	0.4044	-0.4712
6	0.3618	-0.4721
7	0.5194	-0.3600
8	0.3825	-0.4090

Table 3.4-1

Maximum and minimum limits for the quantizer used to quantize the first eight log area coefficients for: (a) No pre-emphasis and (b) Fixed pre-emphasis of 0.4.

Coefficient Number	Bits	Levels
1	5	32
2	5	32
3	5	32
4	4	16
5	4	16
6	4	16
7	3	8
8	3	8

Table 3.4-2

Bit allocation and number of levels for encoding the log area coefficients

The delay parameter, or pitch period (M) of the distant-sample-based (pitch) predictor needs approximately seven bits of quantization accuracy. Table 3.4-3 lists the minimum and maximum values for the pitch period.

The pitch predictor coefficients $\{b(k)\}$ can be quantized either directly, or after using a suitable transformation [51]. However, the results are inconclusive as to the effectiveness of performing transformations, and thus, the decision was to quantize the coefficients directly. Table 3.4-3 indicates the minimum and maximum limits used during the quantization of the predictor coefficients. The method of derivation of these limits is the same as for the limits of the near-sample-based predictor.

The last parameter that is quantized as part of the side information is the rms value of the open-loop prediction residual (θ_{rms}). This value is derived from the open-loop prediction residual signal $u(n)$ as shown in Figure 3.4-6. The rms value of the open-loop prediction residual (θ_{rms}) is defined as:

$$\theta_{rms} = \sqrt{\frac{\sum_{n=1}^N u^2(n)}{N}} \quad (3.4-4)$$

where N is the segment size. This parameter is used by the

Parameter	Minimum	Maximum	Bits
M	15	142	7
b ₁	-0.39746	0.60885	5
b ₂	-0.27905	0.68531	4
b ₃	-0.50801	0.5824	4

Table 3.4-3

Bit allocation and range of quantization for the pitch predictor parameters.

Figure 3.4-6

Block diagram of the open-loop APC coder used to derive the open-loop prediction residual signal $u(n)$.

adaptive quantizer in both the coder and decoder. The quantity θ_{rms} needs approximately 6 bits of quantization accuracy.

The total bit rate for the side information can be calculated using the bit assignments defined for the log area coefficients, the pitch predictor coefficients, the pitch period, and the rms value of the open-loop prediction residual. The details of the bit rate calculations for an APC coder with and without pitch prediction are shown in Figure 3.4-7. The total bit rate of the side information for the coder without pitch prediction is 1.44 kbit/sec. For the coder using pitch prediction, the bit rate of the side information is 2.18 kbit/sec. If the total bit rate of the coder would be 16 kbit/sec, then the bit rate of the side information makes up to 14 percent of the coder's total bit rate. The remainder of the bit rate is made up of the encoded prediction residual waveform. Efficient quantization and encoding of this signal is of obvious importance. The next section deals with this subject.

3.4.6 Efficient Quantization of the Prediction Residual

In addition to the transmission of the side information, the digital channel also carries information concerning the quantized prediction residual $\hat{r}(n)$. In section 3.4.5 it was shown that the prediction residual can

2

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

RMS value of the open-loop

prediction residual :

6 bits

log-area

coefficients : $(5 + 5 + 5 + 4 + 4 + 4 + 3 + 3) = \underline{33 \text{ bits}}$

total side information without pitch prediction $\underline{39 \text{ bits}}$

pitch predictor coefficients : $(5 + 4 + 4) = 13 \text{ bits}$

/pitch period : $\underline{7 \text{ bits}}$

total side information with pitch prediction $\underline{59 \text{ bits}}$

Bit rate of the side information

segment size (N) = 180

sampling rate = 6.666 kHz

$$\text{bit rate} = \frac{\text{total side information bits} \times \text{sampling rate}}{\text{segment size}}$$

$$\text{without pitch prediction : bit rate} = \frac{39 \times 6.666 \times 10^3}{180}$$

$$= 1.44 \text{ kbit/sec}$$

$$\text{with pitch prediction : bit rate} = \frac{59 \times 6.666 \times 10^3}{180}$$

$$= 2.18 \text{ kbit/sec}$$

Figure 3.4-7

Calculation of the bit rate of the side information for the APC coder with and without pitch prediction.

make up, typically, 75% of the total bit rate of the coder. For example, using a simple one-bit quantizer for the prediction residual will result in a bit rate of 6.666 kbit/sec for the prediction residual and a total bit rate of 8.85 kbit/sec (assuming the previously defined side information bit rate of 2.18 kbit/sec). However, the output speech that is produced using this simple quantization method sounds quite rough. For a one-bit quantizer, a small quantizer step-size produces peak-clipping of the closed-loop prediction residual signal $r(n)$ which causes occasional pops and clicks in the output speech. If the quantizer step-size is too large then granular (random) noise is perceivable in the output speech. The production of better quality output speech can only be achieved if more complex quantization of the prediction residual is used.

Figure 3.4-8 shows how the redundancy of the input speech is reduced after both near-sample-based or spectral prediction and pitch prediction have been performed. However, large spikes or peaks still exist in the open-loop prediction residual signal $u(n)$. Thus, the closed-loop prediction residual signal $r(n)$ will also exhibit these spikes as well. It has been shown by Atal that better quality output speech can be produced if the high amplitude portions of the closed-loop prediction residual $r(n)$ are accurately quantized [58]. Moreover, he showed that very

Figure 3.4-8

APC coder waveform examples: (a) speech input; prediction error with (b) spectral prediction and (c) spectral prediction plus pitch prediction. Waveforms (b) and (c) have been magnified by $\sqrt{10}$ and 10, respectively. (After [51].)

little or no audible distortion is introduced if the closed-loop prediction signal is severely center-clipped before encoding. The block diagram of a quantizer which exhibits both severe center-clipping and multi-level quantization of the high-amplitude portions of its input is shown in Figure 3.4-9. The rms value of the prediction residual θ_{rms} , calculated previously in equation 3.4-4, is first used to scale down the closed-loop prediction residual signal before it is input to the quantizer. The value L_c is the threshold level for the center clipper. The value L_p is the peak-clipping limit of the quantizer. The number of quantizer levels is fixed arbitrarily for the duration of the segment (N samples) or varied on a sample by sample basis.

As a result of the center clipping, the output of the quantizer contains a large number of zeros. Thus, the entropy of the quantizer can be made quite low depending on how high the threshold value for the center clipper is set. Table 3.4-4 shows a typical frequency distribution of the levels used by an eleven-level center-clipping quantizer for test speech sample number one. The table shows that the center-clipping threshold level L_c is set at 95% of the rms value of the prediction residual θ_{rms} . The peak-clipping limit L_p is set at 600% of θ_{rms} . These values were kept constant for the entire segment. The resulting zero level output constitutes 61.8% of the quantizer output. The

Figure 3.4-9

Closed-loop prediction residual signal $r(n)$ quantizer showing how both center-clipping and multi-level quantization of the high-amplitude portions of $r(n)$ is achieved.

Center-clipping threshold:

$$L_c = 0.95 \times \text{RMS value of the prediction residual } (\Theta_{rms})$$

Peak-clipping limit:

$$L_p = 6.00 \times \Theta_{rms}$$

Entropy = 1.76, using Huffman coding

Level Number	Frequency
5	0.001
4	0.001
3	0.006
2	0.028
1	0.149
0	0.618
-1	0.155
-2	0.034
-3	0.006
-4	0.001
-5	0.001

Table 3.4-4

Frequency distribution of the quantizer levels employed for an eleven-level center-clipping quantizer.

entropy for this quantizer was calculated to be 1.76.

The unequal probabilities of the quantizer output levels suggests that some type of variable length coding (VLC) of these levels should be used. Variable length coding assigns codewords of variable lengths to the possible level outcomes such that highly probable outcomes are assigned shorter codewords, and vice versa. However, problems can arise in performing variable length coding. Buffer overflow problems can occur when a large number of long code words are output from the variable length coder. Also, some variable length codes are not self-synchronizable. For example, one transmission error affecting only one coded symbol could cause the receiver to decode this symbol and all subsequent symbols in error. A number of solutions exist for the aforementioned two problems inherent in variable length coding.

One possibility of providing codeword synchronization would be to use variable-length-to-block coding. These codes use codewords of fixed lengths to represent a variable number of input samples. Because the codewords are fixed blocks there are no codeword synchronization problems. This type of coding has been used previously in conjunction with the center-clipping quantizer introduced in this section [51]. However, channel errors could cause a codeword to be decoded into a sequence with a different number of samples

than what it actually contained. This difference could be quite large.

Another method for providing codeword synchronization is shown in Table 3.4-5 for a quantizer with eleven levels. This code is self-synchronizable in that each codeword is of differing length and consists of all ones except for the last bit which is zero. The receiver can re-synchronize every time it receives a zero. The entropy of the code is kept low by assigning the longer codewords to the least probable levels, and vice versa. The entropy of this code, assuming the probability distribution of Table 3.4-4, is 1.78. Recall that the optimal or Huffman Code for this distribution was calculated to be 1.76. An error affecting a transmitted codeword would cause the codeword to be decoded either as two codewords in error or else, combined with the next coded word, as a single codeword in error. Thus, an error in one transmitted codeword will affect at most two output samples. The performance of this type of coder with channel errors introduced is much better than the variable-length-to-block coder discussed in the previous paragraph.

Buffer overflow is another problem which is inherent with variable-length coding. The quantizer produces samples at a constant rate which are mapped into codewords of variable length. These codewords must be initially fed into

Level Number	Codeword	Length
5	1111111110	10
4	11111110	8
3	111110	6
2	1110	4
1	10	2
0	0	1
-1	110	3
-2	11110	5
-3	1111110	7
-4	111111110	9
-5	11111111110	11

Table 3.4-5

Self-synchronizing code for an eleven-level quantizer.
 The entropy of this code assuming the probability
 distribution of Table 3.4-4 is 1.78.

a buffer, to accommodate a communications channel that accepts symbols at a constant rate. Buffer overflow occurs when a succession of long codewords is generated such that no room is left to store the next generated codeword. At the receiver, decoding delays occur as the decoded prediction residual waveform becomes out of date with respect to the current side information, which is received at a constant rate. The receiver must be able to buffer this side information.

Several techniques exist that minimize or eliminate buffer overflow problems. One scheme uses buffer-feedback control of the quantizer. With this technique, the quantizer range is adaptively companded on a sample-by-sample basis according to the status of the buffer. When the buffer is filling up, and is in danger of overflowing, the quantizer range is expanded so that symbols of smaller length have greater probability of being output.

Another method of changing the quantizer's characteristics is to switch between two fixed quantizers. Typically, a quantizer with variable length encoding is used at first and then a fixed-symbol quantizer is switched into use when it is known that the buffer will overflow with certainty. More detail will be given of these two buffer control methods in the paragraphs that follow.

The closed-loop APC system of Figure 3.4-5 was utilized to investigate the quantization and encoding of the closed-loop prediction residual signal $r(n)$. Quantization and encoding of the side information was effected using the methods of section 3.4.5. Center-clipping/multi-level quantization, and the variable-length code of Table 3.4-5 were used to quantize and encode, respectively, the prediction residual $r(n)$. The coded prediction residual was allocated a total number of R bits per segment, thus keeping the bit rate of the APC coder fixed at a constant rate. The first method used to eliminate the buffer overflow problem was buffer-feedback control of the quantizer.

With buffer-feedback control, the center-clipping threshold L_c of the quantizer is expanded, on a sample-by-sample basis, according to the number of remaining bits left of the total number of R bits allocated to encode the prediction residual. However, this method does not ensure that all N prediction residual samples will be encoded into R bits, or less. It is still possible that a build-up of uncoded prediction residual samples could cause a buffer overflow.

Several problems were encountered during simulations of the APC coder using buffer-feedback control of the quantizer. Limit cycles and instabilities occurred during some voiced segments of the speech sample used. It was

apparent that the quantizer was having difficulty coping with large prediction residual samples, especially near the end of the segment. It is known that using pitch prediction can sometimes cause these stability problems, especially if coarse quantization and low sampling rates are used. The removal of pitch prediction from the APC structure seemed to alleviate the stability problems but they still persisted. It is probable that the stability problems arise because of the coarse quantization that can occur at the end of the segment during periods of heavy buffer use. The second method of buffer overflow control was investigated in the hope of eliminating the stability problems.

The second method of buffer-overflow control uses a switched quantizer. The center-clipping-multi-level quantizer is used jointly with variable-length coding while the buffer still has room. A simple quantizer with fixed-length encoding is switched into effect if it is known the buffer will overflow. The type of simple quantizer used for this purpose was a one-bit quantizer. The algorithm for switching into one-bit quantization works as follows. A total of R bits are allocated for encoding the prediction residual signal. A count is made of both the remaining number of samples during the segment (A), and the remaining number of bits left for encoding the prediction residual (B). If A becomes less than or equal to B then the coder

switches into one-bit coding of the prediction residual. The variables A and B are initially set to N and R , respectively, at the beginning of the segment. This type of switched quantizer has been used previously by Tanaka, et al. [28].

Simulations were carried out on the APC coder with and without pitch prediction incorporated in the coder's structure. The value for R , the total number of allocated bits for encoding the prediction residual, was set high enough so that the quantizer never switched into one-bit coding.

For both types of structures, limit cycles or stability problems were encountered. To cure the stability problems both pre-emphasis of the input speech and a larger peak-clipping limit L_p for the quantizer were used. It was possible to use a pre-emphasis constant μ of 0.4 and a mild increase in L_p to cure the stability problems for the APC coder without pitch prediction. However, much more pre-emphasis and a much larger L_p were needed for the APC coder using pitch prediction. For this coder the stability problems were eliminated, but the output speech was severely degraded to the point of having worse quality than the coder without pitch prediction. Thus, further implementations of the APC coder would not use pitch prediction because a lower bit rate and less complexity could be achieved.

The reconstruction level for the one-bit quantizer was the next issue to be investigated. Figure 3.4-10 shows a block diagram of the switched quantizer with both the center-clipping quantizer and one-bit quantizer incorporated. The value of R was set equal to N at the start of the segment. This forced the switched quantizer into one-bit coding from the first sample in the segment onwards. Table 3.4-6 illustrates results of using different values for the reconstruction level L_r . The value of L_r that produces the highest segmental SNR is 1.65.

Using the optimized one-bit quantizer, simulations were run for the APC coder with several values for the quantity R . The value R , the number of bits to encode the prediction residual, essentially controls the bit rate of the coder since the rate of the side information is fixed. R was varied from 180 to 360.

There are also various parameters that control the arrangement of the variable-length quantizer which can be varied to control the quality of speech that the coder produces. These parameters must be set so as to produce speech of optimal quality. The quantizer parameters are: the number of levels (N_L), the center-clipping threshold (L_Q), and the peak-clipping limit (L_P). Essentially, N_L controls the average switch-over point of the quantizer to one-bit coding. L_Q and L_P control the amount of granular

Figure 3.4-10

Switched quantizer, consisting of a center-clipping multi-level quantizer and a one-bit quantizer.

Reconstruction Level Value	Segmental SNR
0.50	5.40
0.70	7.32
0.90	9.13
1.10	10.35
1.30	11.04
1.50	11.66
1.55	11.64
1.60	11.62
1.65	11.73
1.70	11.66
1.90	11.47
2.10	11.06
2.30	10.73

Table 3.4-6

Segmental SNR performance for various reconstruction level values of the one-bit quantizer. The segment size (N) is 180. The prediction order (P) is 8.

noise, and peak-clipping distortion of the output speech, respectively. Section 3.4.8 will describe in greater detail the simulations that were carried out to arrive at the various values for R , N_L , L_c and L_p in order to produce several final configurations of the APC coder.

3.4.7 Contending with Channel Errors

As mentioned in section 2.1, there are basically two different types of error patterns that can occur on a mobile radio channel. Bursty error patterns are the most common. These patterns are due to multipath propagation, which produce severe signal fades. Random error patterns, which are less common, are due to external interference to the transceiver, such as automobile ignition noise, although ignition noise can also cause burst errors.

Section 4.1 describes a series of mobile radio field tests conducted by researchers at Carleton University to investigate different methods of transmitting digital data [56]. The bit rate of the data was 4.8 kbit/sec and the modulation frequency was 820 MHz (upper UHF). Refer to section 4.1 for a detailed description of the experiments conducted. Although average bit error rates of over 1% were encountered for a few test runs, only those test runs with average bit error rates of under 1% were considered for this study. It was felt that a transmission channel with average

bit error rates of over 1% would cause too great a degradation in the output speech produced by the speech coders under consideration. Greater transmitter power or diversity techniques should be used to keep the average bit error rate below 1%. The average burst error length for the test runs to be considered was around 9 bits.

The data block sent by the APC coder is made up of two types of information; the side information; and the encoded residual waveform. The side information consists of the encoded bits for the near-sample based predictor coefficients and the rms value of the open-loop prediction residual. The spectral distortion measure was used in a previous study to determine the degree of degradation introduced by errors affecting the side information [57]. The authors arbitrarily decided that if the information frame has four or more bits in error then the spectral information of that frame can be considered to be destroyed. Since the average burst error length of the mobile radio channel data for the study conducted at Carleton University was determined to be nine bits in length, one can conclude that a burst error affecting the side information will most likely destroy that information.

Two approaches can be made to protecting the side information that has been hit by a burst of errors. One approach is to use extensive error correction coding which

will protect the side information from the errors. Considering the typical average burst error length of the mobile radio channel and the resulting overhead in the parity bits (and thus, increase in the total bit rate) this approach was ruled out.

The second method of protection would be to concede that the side information will likely be destroyed if it is hit by a burst of errors and to regenerate alternative side information in its place. Generation of the alternative side information can be realized by utilizing the fact that the side information for neighbouring segments are highly correlated, especially for voiced segments. Thus, the side information for a segment corrupted by errors can be directly replaced by side information from the last uncorrupted segment. Detection of whether the side information has been hit by a burst error can be effected by using a Hamming error correction code. Hamming codes are capable of correcting data blocks with one bit in error.

They can also detect (but not correct) multiple errors in a data block. The code is constructed by adding an appropriate number of parity bits to the end of each transmitted codeword. The block length of the side information for the APC coder was determined to be 39 bits. A shortened (63,57) Hamming code can be used to provide one-bit error correction and multiple-bit error detection of.

this information. This code has 6 parity bits, thus the block length of the side information increases to 45.

The Hamming code method of protecting the side information is relatively simple. Also, the bit rate of the coder is not significantly increased. Thus, this method of protecting the side information was effected.

The performance of the Hamming code to burst-type errors and random-type errors was as expected. Two error patterns, one of each type, were subjected to the side information of the APC coder. The average bit error rate of the patterns were both 0.74%. The average burst error length for the burst-type error pattern was 6.8 bits. The coder processed 276 segments of speech. The results show that for the burst-type error pattern, 28 segments of side information were corrupted by errors. The Hamming code corrected only 3 of these segments. For the random-type error pattern, 94 segments of the side information were corrupted by errors. The Hamming code corrected 81 of these segments. Thus, for burst error patterns, the Hamming code is only useful for detection, and not correction.

The other set of information that can be corrupted by channel bit errors is the encoded prediction residual. The amount of this information is large in comparison to the side information. Up to 90% of the bit rate of the coder is

due to the encoded prediction residual data block. However, this data carries relatively less information about the input speech in comparison to the side information. In fact, if the prediction of the input speech is done well then the residual signal should be noise-like in nature. Errors corrupting the residual signal will not have that much effect on its statistics. However, the encoded residual data block is made up of a variable-length portion, and a single-bit portion. The coding boundary of these portions always occurs when the number of samples remaining during the segment becomes less than or equal to the number of bits left to encode the prediction residual. Any errors to the coded bit sequence will most likely cause problems for the receiver to decode this boundary. The output speech will become degraded because of this decoding problem. Error correction techniques could be used to diminish the degradation in the speech but the overhead involved would increase the bit rate of the coder substantially. Thus, the decision was to omit error protection of the prediction residual data block.

Simulations were run to see how much the operation of the decoder, and thus the output speech quality, would be affected by the introduction of bit errors in the encoded prediction residual data block. The results of these simulations show that these errors do affect the quality of

the output speech produced by the APC coder. The effect, however, is not enough to warrant the use of error correction coding.

Occasionally, the received signal power on the mobile radio channel can become so low that bit timing or synchronization of the receiver could be lost for a brief period of time. When the receiver regains bit synchronization it has no way of knowing which part of the data block the next decoded bit belongs to. The simplest way to ensure block synchronization is to assign a globally-known synchronization symbol to the beginning of the data block. The length of the symbol can be kept short, around eight bits for example. The finalized transmission data block in Figure 3.4-11 illustrates the location of the synchronization byte as well as the location of the other three information groups.

3.4.8 Final Configurations of the APC Coder

Figure 3.4-12 illustrates that the number of bits (R) devoted to encoding the prediction residual signal determines the bit rate of the coder. At the lowest extreme, 180 bits are used to encode 180 prediction residual samples. This effects a coder with a bit rate of 8.16 kbit/sec. To effect a coder with a bit rate of 16 kbit/sec, 380 bits should be used to encode the prediction residual.

Sync Byte	Side Information	Parity Bits	Encoded Prediction Residual
--------------	---------------------	----------------	--------------------------------

Figure 3.4-11

Transmission Data Block

synchronization block :	8 bits
side information :	39 bits
parity bits :	6 bits
encoded prediction residual :	<u>180 bits to 380 bits</u>
total data block :	233 bits to 433 bits

$$\text{bit rate} = \frac{233 \times 6.666 \times 10^3}{180} = 8.6 \text{ kbit/sec}$$

$$\text{bit rate} = \frac{433 \times 6.666 \times 10^3}{180} = 16.0 \text{ kbit/sec}$$

Bit rate is from 8.6 kbit/sec to 16.0 kbit/sec

Figure 3.4-12

Calculation of the bit rate of the final configuration of the APC coder.

Three configurations for the APC coder were derived. The values of R (the number of bits used to encode the prediction residual) and the resulting bit rates for the three coders are:

APC coder 1 - $R = 180$, 8.6 kbit/sec

APC coder 2 - $R = 260$, 11.6 kbit/sec

APC coder 3 - $R = 340$, 14.6 kbit/sec

The bit rates are calculated using the same method as in Figure 3.4-12.

Simulations were carried out for the three coders using test speech sample one. The parameters derived for the APC coder in previous sections were again used for the coders during these simulations. The only parameter values that differed among the three coder configurations were the three parameters for the variable-length quantizer: N_L , L_C , and L_P . These values had no bearing on APC coder 1 as it uses one-bit coding throughout the entire segment. The values of three parameters for the other two coders are:

	<u>Coder 2</u>	<u>Coder 3</u>
Number of quantizer levels (N_L)	9	11
Center-clipping threshold (L_C)	120%	95%
Peak-clipping limit (L_P)	650%	600%

Results of the simulations for all three coders are shown in Figure 3.4-13. As the number of bits (R) allocated to encode the prediction residual is increased, a number of trends appear. Firstly, the number of segments using some one-bit coding decreases. Secondly, the entropy of the variable-length coder increases. With the knowledge of these last two trends, it is not surprising that the signal-to-noise ratio, and thus the quality, of the three coders increases as well.

Informal listening tests of the three coders showed that APC coder 3 produced output speech of the most natural quality. There was very little discernable distortion in the speech. The speech for APC coder 2 contained some high frequency distortion and sounded more "garbled". APC coder 1 produced speech that was the least natural of the three coders. All three versions of the APC coder appeared to produce better speech quality than the delta modulation speech coder. The higher bit rate version of both the APC coder and RELP coder produce speech of similar quality. The lower bit rate versions of these two coders have quite different quality distortions and are thus, difficult to compare.

	<u>Coder 1</u>	<u>Coder 2</u>	<u>Coder 3</u>
<u>One-bit quantizer</u>			
number of segments that use, partially, one-bit coding (total of 180 segments) :	180	145	80
average switch-over point to one-bit coding :	1.0	113.9	143.2
<u>Variable-length quantizer</u>			
output probability of the zero level :	-	0.67	0.59
entropy :	-	1.59	1.83
<u>APC coder performance</u>			
Signal-to-noise ratio (dB) :	10.2	13.4	16.8
Segmental signal-to-noise ratio (dB) :	11.4	14.4	17.4

Figure 3.4-13

APC simulation results using test speech sample one.

3.5 Coder 3 : Residually Excited Linear Prediction Coder (RELP)

3.5.1 Introduction to Residually Excited Linear Prediction

In section 3.4, the APC method of speech coding was described. The APC coder is a hybrid coder which uses a suitable combination of waveform coding techniques and vocoder techniques. In particular, linear prediction methods are used to generate a predicted speech signal which is subtracted from the input speech to form a residual waveform. The bit rate of the APC coder can be kept relatively low by utilizing efficient quantization of this prediction residual waveform.

The residually excited linear prediction (RELP) speech coder, originally proposed by Un and Magill [58], is similarly, a hybrid speech coder that uses linear prediction techniques to form the prediction residual signal. The RELP coder uses a different method than APC to reduce the bit rate of the residual signal before it is quantized. The bit rate reduction is achieved by low-pass filtering the residual signal followed by a lowering of the sampling frequency (decimation) by a certain factor (2 to 8). At the receiver, the fullband residual signal is reconstructed, approximately, by applying high frequency regeneration techniques to the received decimated baseband signal. The full-band signal is then used as the excitation signal for

the all-pole synthesis filter.

The RELP speech coder is able to produce good-quality, natural-sounding speech at medium bit rates (8 to 16 kbit/sec). In subjective speech quality tests, the RELP coder has outperformed the APC coder at bit rates below 12 kbit/sec [11]. The performance of the RELP coder in the presence of channel errors should be slightly better than that of the APC coder. This belief is based on the less complex nature of the quantization and encoding process for the RELP coder compared to the APC coder. Thus, the RELP coder appears to be a viable speech coder for mobile radio.

In this section, the RELP speech coder will be examined. Sub-section 3.5.2 describes a RELP coder based on a system first introduced by Viswanathan, Higgins, and Russel [26]. The following sub-section will describe in greater detail the implementation of the different blocks of the RELP coder. Sub-section 3.5.4 will outline the methods used to contend with channel errors. Finally, the bit rates for three configurations of the RELP coder and preliminary results based on simulations will be presented.

3.5.2 Basic RELP System

Figure 3.5-1 shows a block diagram of a basic RELP speech coder. At the transmitter, a predicted speech signal

Figure 3.5-1

Block diagram of a basic RLP speech coder:

(a) transmitter and (b) receiver.

$\tilde{s}(n)$ is subtracted from the input speech $s(n)$, to form the open-loop prediction residual signal $d(n)$. The predicted speech samples $\tilde{s}(n)$ are formed using linear prediction techniques similar to those used for the APC coder. Refer to section 3.4.3 for more details on the prediction of speech. The spectral envelope of the prediction error signal $d(n)$ is rendered relatively flat because most of the redundancy of the input speech signal $s(n)$ is removed by the prediction process.

The ~~next~~ block of the RELP coder is the lowpass filter $h(n)$ which extracts the portion of the prediction residual signal $d(n)$ below a frequency of B Hz. The filtered signal $x(n)$ is usually termed the "baseband" of the prediction residual signal. The extracted baseband signal $x(n)$ is then decimated (downsampled) to its Nyquist rate of $2B$ Hz to produce the signal $y(n)$. The sampling rate reduction is achieved by saving only every L th sample of $x(n)$. The decimation factor L can be expressed as

$$L = W/B$$

where W is the bandwidth of the input speech signal $s(n)$ and B is the bandwidth of the extracted baseband signal $x(n)$. For the general decimation process, L must be expressed as a rational fraction. The decimation process can be further simplified if L is an integer. Figure 3.5-2 shows typical

Figure 3.5-

Typical spectra for decimation by an integer factor L . (After [6].)

spectra (magnitude of the discrete Fourier transform) of the signals $d(n)$, $h(n)$, $x(n)$ and $y(n)$, illustrated in Figure 3.5-1, for an integer decimation factor of L . It can be seen from Figure 3.5-2 that the lowpass filter $h(n)$ limits the amount of aliasing of the signal $x(n)$ during the decimation process in forming the signal $y(n)$. Thus, the stopband of the lowpass filter must be designed such that these aliased components of the signal $y(n)$ are not perceivable in the output speech.

The next block of the RELP coder quantizes the decimated baseband residual $y(n)$. Since the highpass information in the signal $x(n)$ is first removed and the sampling rate is subsequently lowered, the quantization process that follows does not have to be as involved as, say, the APC coder. It is sufficient to use a fixed block quantizer and encode the quantizer output with a moderate number of bits (2 to 4).

At the receiver, a high-frequency regeneration process is used to produce a signal with a full bandwidth of W Hz. This process is illustrated in Figure 3.5-1(b). The first step of the regeneration process is to interpolate the received decimated baseband signal $\hat{y}(n)$ by inserting $L-1$ zero-valued samples between each pair of samples of $\hat{y}(n)$. This effects an upsampling of L -to-1. Subsequent lowpass filtering produces the reconstructed baseband residual $\hat{x}(n)$.

This signal is then used to regenerate the high frequency component (above B Hz) of the prediction residual. The regenerated signal is then highpass filtered at B Hz and summed with $\hat{x}(n)$ to give the full-band prediction residual signal $\hat{d}(n)$. The resulting signal $\hat{d}(n)$ is then applied to the all-pole LPC synthesis filter to produce output speech. Figure 3.5-3 shows typical spectra for the signals $\hat{y}(n)$, $z(n)$ and $\hat{d}(n)$. The spectrum for the signal $z(n)$ resembles a "folded" version of the spectrum for $\hat{y}(n)$. This folded spectrum is due to the interpolation process. The resulting spectrum of the full-band prediction residual signal $\hat{d}(n)$ will look like the spectrum of $\hat{y}(n)$ below the frequency B Hz. The shape of the spectrum envelope of $\hat{d}(n)$ above B Hz will depend on the type of high-frequency regeneration (HFR) method used. Two simple but effective HFR methods are non-linear distortion and spectral folding.

High-frequency regeneration is a process where a high frequency signal is generated from the baseband residual signal such that the resulting full-band signal has a flat spectrum and proper harmonic structure for voiced sounds [26]. The spectral folding technique was described earlier in the context of the interpolation process. Recall that interpolating a baseband signal results in a signal with aliased versions of the baseband signal around multiples of the decimated sampling frequency. If the baseband signal

Figure 3.5-3

Typical spectra for the generation of the full-band prediction residual $\hat{d}(n)$. (After [6]).

has a flat spectral envelope up until B Hz then the resulting full-band signal will have a flat spectral envelope until W Hz. Thus, the hoarseness, which is typical in the output speech for RLP coders using non-linear distortion techniques, is not present. In its place, audible "tonal noises" are present at multiples of $2B$ Hz. The tonal noises are most apparent for voiced speech sounds where the resulting baseband residual signal does not have a flat spectrum due to pitch sounds which "leak through" the prediction process. These pitch sounds get aliased, inharmonically, during the interpolation process. Thus, the resulting tonal noises are more audible for large decimation factors and high-pitched speakers. One method of reducing tonal noises due to voiced sounds is to utilize another stage of prediction based on distant samples. There is some evidence that this "pitch" prediction process works better when applied to the decimated baseband residual $y(n)$ rather than the prediction residual $d(n)$ [59]. Recall, that the subject of pitch prediction was investigated for the APC coder in section 3.4.4.

The spectral folding method exhibits spectral regularity in the way it reflects the baseband into the higher frequency bands. Tonal noises arise from this regularity. Perturbed spectral folding is one method that has been found to break up this spectral regularity

[26]. This method essentially adds a signal-correlated noise to the spectrally folded residual signal. The additive noise has the effect of masking the tonal noises at the expense of causing slight roughness in the output speech.

3.5.3 Implementation of a RELP Coder with Pitch Prediction and Perturbed Spectral Folding

Based upon the separate results of Viswanathan [26] and Sluiter [59], it was decided that the RELP coder to be implemented should include pitch prediction and perturbed spectral folding. Both of these additions can be switched in or out of the basic RELP coder at any time during the simulations to observe their effects on the reconstructed output speech. The paragraphs to follow, will describe in greater detail how the blocks of the RELP coder were implemented.

Spectral Predictor

The spectral parameters of the RELP coder are the LPC coefficients which are solved on a segmental basis, using a particular solution method. In section 3.4.3, a spectral predictor was derived for the APC coder using the autocorrelation method of solving for the predictor coefficients $\{a(k)\}$. There are basically three parameters

that can be varied when solving for the coefficients and they are: the segment size N , the prediction order M , and the size of the windowed segment K . In section 3.4.3 it was found that no windowing, a prediction order of 8, and a segment size of 180 provided adequate prediction of a speech file with a 3.33 kHz band width (speech sample number one). The same method of solution (autocorrelation method) and prediction parameter values as those used for the APC coder were used for the RELP coder. The performance of the RELP coder was sufficient using these parameters. Higher prediction orders of 12 or 14 were found to give only a slight performance gain.

Baseband Extraction and Decimation.

The baseband signal is obtained by lowpass filtering the prediction residual signal $d(n)$. The cut-off frequency of the filter is B Hz where $B = W/L$. The bandwidth (W) of the speech signal used for testing the RELP coder was 3.333 kHz. Decimation factors of $L = 2, 3$, and 4 were used for the downsampling process. Thus, three lowpass filters with cut-off frequencies of 1.67 kHz, 1.11 kHz, and 833 Hz, respectively, were required. Linear phases FIR filters were used to implement the lowpass filtering operation. These filters have the property of producing a constant group delay at their output. This group delay can then later be

compensated for. The three filters were designed using the program, FILTER, written by McLellan et al. [61]. The program utilizes the Remez exchange algorithm to design filters with linear phase. The three filters were designed to have a filter order of 181 and a transition region of 110 Hz. This filter order produces a constant phase delay of 90 samples. Table 3.5-1 details important specifications for the three filters. Note that the resulting stop-band attenuation for all three filters is 55.2 dB. The stop-band attenuation must be this large so as to adequately extract the baseband and to avoid aliasing during the spectral folding process. Figure 3.5-4 shows the frequency response of the lowpass filter used for the decimation by 2 process.

Pitch Predictor

The pitch predictor is placed after the decimation process and incorporates an adaptive quantizer in a closed-loop configuration. Figure 3.5-5 shows a block diagram of the transmitter portion of the RELP coder indicating the placement of the pitch predictor. The pitch predictor for the RELP coder utilizes three predictor coefficients, or taps. This predictor is the same as the one proposed for the APC coder in section 3.4.4. The system function of the pitch predictor is given by equation 3.4-2. The AMDF method of pitch period estimation is used to calculate the pitch

Decimation Factor	Transition Region (Hz)
4	778 - 888
3	1056 - 1166
2	1611 - 1721

Order of the filters = 181
 Transition region = 110 Hz
 Sampling frequency = 6.666 kHz
 Resulting stop-band attenuation = 55.2 dB

Table 3.5-1

Specifications for three low-pass filters which perform baseband extraction.

Figure 3.5-4 Frequency response of a lowpass filter for decimation by 2.

Figure 3.5-5

Block diagram of a RLP transmitter incorporating pitch prediction in a closed-loop configuration.

period M from the decimated baseband residual signal $y(n)$. The algorithm considers pitch periods from 2.4 to 20.3 msec. The pitch predictor coefficients, b_1 , b_2 , and b_3 , are also calculated from the signal $y(n)$ using the computed pitch period.

Adaptive PCM Coder

Since the bit rate of the prediction residual has already been reduced by the decimation process, and the pitch predictor removes further redundancy in this signal, the quantization of the closed-loop second residual signal $w(n)$ need not be too critical. Thus, a simple quantizer, using fixed block coding, can be utilized. The quantizer employed was a pdf-optimized non-uniform unit-variance quantizer. The reconstruction levels of the quantizer were based upon the assumption that the quantizer input will have a Laplacian distribution. In previous work [26], a Laplacian quantizer outperformed Gaussian and gamma quantizers for speech data. Table 3.5-2 shows the optimum decision values [for input $x(n)$] and reconstruction values [for output $y(n)$] of the Laplacian quantizer [43]. Only codewords of lengths 3 and 4 bits long are presented in the table. To ensure that the input to the quantizer will be of unit variance, the input samples are divided by a gain G equal to the rms value of the open-loop pitch prediction

Figure 3.5-6

Derivation of the open-loop prediction residual $u(n)$.

Bit Length	3		4	
n	x(n)	y(n)	x(n)	y(n)
1	0.000	0.233	0.000	0.124
2	0.533	0.833	0.264	0.405
3	1.253	1.673	0.567	0.729
4	2.380	3.087	0.920	1.111
5	-	-	1.345	1.578
6	-	-	1.878	2.178
7	-	-	2.597	3.017
8	-	-	3.725	4.432

Table 3.5-2

Table of optimum decision values $x(n)$ and reconstruction values $y(n)$ for the Laplacian quantizer. Codewords of length 3 and 4 bits long are shown.

residual $\hat{u}(n)$. Figure 3.5-6 indicates how $u(n)$ is derived. The quantizer is made adaptive by resetting the value of G every segment (N samples). It is assumed that the rms value of the open-loop pitch prediction residual $u(n)$ will be close to the rms value of the closed-loop pitch prediction residual $w(n)$. The two rms values will be slightly different due to the effects of the quantization process on the closed-loop signal $w(n)$.

Encoding the Predictor Parameters

The RELP coder sends prediction information over the channel during every segment to be used by the decoder to reconstruct the original speech. This information constitutes the entire side information of the RELP coder. The prediction information to be sent are: the log-area coefficients $\{c(k)\}$, the pitch predictor coefficients $\{b(k)\}$, the pitch period (M), and the rms value of the open-loop pitch prediction residual (G). Section 3.4.5 outlines the methods used to quantize and encode all four sets of parameters for the APC coder. These same methods were used for the RELP coder.

In Figure 3.5-7, calculations for the bit rate of the side information are shown. As the figure indicates, the bit rate for the side information using decimation by a factor of 4 is 2.52 kbit/sec. The bit rate for the side

RMS value of the open-loop prediction residual :	6 bits
log-area coefficients : (5 + 5 + 5 + 4 + 4 + 4 + 3 + 3) =	33 bits
pitch predictor coefficients : (4 + 3 + 3) =	10 bits
pitch period : (decimation by 4)	5 bits
(decimation by 3 or 2)	6 bits
total side information (decimation by 4)	54 bits
(decimation by 3 or 2)	55 bits

Bit rate of the side information

segment size (N) = 180
sampling rate = 6.666 kHz

$$\text{bit rate} = \frac{\text{total side information bits} \times \text{sampling rate}}{\text{segment size}}$$

$$\begin{aligned} \text{decimation by 4 :} \quad \text{bit rate} &= \frac{54 \times 6.666 \times 10^3}{180} \\ &= 2.52 \text{ kbit/sec} \end{aligned}$$

$$\begin{aligned} \text{decimation by 3 or 2 :} \quad \text{bit rate} &= \frac{55 \times 6.666 \times 10^3}{180} \\ &= 2.56 \text{ kbit/sec} \end{aligned}$$

Figure 3.5-7

Calculation of the bit rate of the side information for the RELP coder.

information using decimation by a factor of either 3 or 2 is 2.56 kbit/sec.

Perturbed Spectral Folding

It was mentioned in section 3.5.2 that the spectral folding method produces tonal noises in the output speech due to the spectral regularity that results as the baseband is reflected into the higher frequencies. It was suggested that perturbed spectral folding could be used to break up this spectral regularity and thus, lessen the tonal noise. This method, due to Viswanathan, Higgins, and Russel [26] will now be examined.

Figure 3.5-8 shows a block diagram of the revised RELP receiver incorporating both pitch prediction and perturbed spectral folding. The random perturbation process is performed on non-zero samples of the folded baseband residual signal $\hat{z}(n)$. Perturbation is performed by simply interchanging the sample with an adjacent (zero) sample. No perturbation is performed if the magnitude of the non-zero sample is larger than an arbitrary threshold X ; this avoids perturbing samples corresponding to pitch pulses. Non-zero samples with magnitudes less than X are perturbed with a variable probability, with the smaller valued samples having a higher probability of being perturbed. The perturbed signal is then highpass filtered and added to the

Figure 3.5-8

Block diagram of the RLP decoder incorporating pitch prediction and perturbed spectral folding.

reconstructed baseband residual signal $\hat{x}(n)$.

A flowchart of the perturbation process is shown in Figure 3.5-9. The random numbers are generated using a zero-mean Gaussian distribution with variance $\sigma^2(x)$. The variance $\sigma^2(x)$ is set equal to 0 for $|x| \geq X$, and $D(1 - |x|/X)$ for $|x| < X$. The values of the parameters X , C , and D were chosen to give the best output speech quality. The optimal values are: $C = 0.5$, $D = 0.7$, and $X =$ the long-term rms value of the decimated baseband residual signal $\hat{y}(n)$. The rms value was found to be 18.5 for decimation by 4, 21.5 for decimation by 3, and 30.0 for decimation by 2.

In Figure 3.5-8, a lowpass filter is used to extract the baseband signal while a highpass filter is used to extract the randomly perturbed signal. Both of these filters have the same cut-off frequency of B Hz. The two filters are constructed to have the same characteristics as the baseband extracting filter used at the transmitter. Thus, both of these filters produce a constant delay of 90 samples.

Output Reconstruction

The excitation signal $p(n)$ is used along with the spectral filter $P_s(z)$ to reconstruct the output speech. Between the two occurrences where the spectral filter $P_s(z)$

Figure 3.5-9.

Flowchart of the process for randomly perturbing the non-zero samples of the spectrally folded baseband residual signal. The process is explained on the previous page. (After [26].)

is used at the transmitter and the receiver, the intermediate signal goes through two stages of filtering. Since both stages of filtering use linear phase FIR filters with a constant phase delay of 90 samples, the intermediate signal is effectively delayed a total of 180 samples, or one segment. This delay of one segment can be taken care of by buffering the spectral predictor coefficients $\{a(k)\}$ either at the transmitter or the receiver.

Implementation Results

The RELP coder was implemented using the different blocks described in this section. Most of these blocks can be implemented in a straightforward fashion, requiring only a small amount of parameter optimization, unlike the APC coder. The only portions of the RELP coder that required any extensive testing were the pitch predictor and perturbed spectral folding subsystems. These tests were carried out to determine if the performance of the two subsystems, in terms of speech output quality, warranted their inclusion in the final configuration of the coder.

Informal speech tests of the RELP coder showed that the inclusion of pitch prediction cut down the amount of tonal noise perceived in the output speech. This improvement of the speech quality was observed most noticeably for the lower decimation factors of 2 and 3.

7 Perturbed spectral folding was used in place of the normal spectral folding procedure to see if tonal noise could be reduced any further. Some reduction of the tonal noise was perceived for the highest decimation factor of 4. However, the reduction in tonal noise was partially offset by the appearance of a small amount of additive noise. This noise, however, was less objectionable than the tonal noise that it replaced.

Thus, both pitch prediction and perturbed spectral folding were left in the final configuration of the RLP coder because of their capabilities in combating the tonal noise problem.

The last parameter of the RLP coder to be finalized was the codeword size used to encode the closed-loop pitch prediction residual $w(n)$. The two codeword sizes considered were 3 and 4 bits. A codeword size of 3 bits corresponds to having 8 quantizer levels. A codeword size of 4 bits corresponds to having 16 quantizer levels. The output speech quality was found to be only slightly better using a codeword size of 4 rather than 3. The bit rate for the quantizer's encoded output is 33% higher for a codeword size of 4 compared to a codeword size of 3. The difference in output speech quality was not considered to be large enough to warrant having a higher bit rate and so, the codeword size was set to 3.

3.5.4 Contending with Channel Errors

The data block transmitted by the RELP coder is similar to that of the APC coder, being made up of two types of information; the side information and the residual waveform information. Thus, the methods used to contend with mobile radio channel errors should be quite similar.

The side information consists of the encoded bits for the prediction information and the gain constant (G). Considering the relative importance of this information, there should be some sort of error detection/correction associated with it. The same type of error detection/correction used for the APC coder was used for the RELP coder. Recall, that a Hamming correction code was used to provide one-bit error correction and multiple-bit error detection of the side information. A shortened (63,57) Hamming code, having 6 parity check bits, can be used to encode the side information. This increases the size of the side information data block from 54 (decimation by 4) to 60, and from 55 (decimation by 3 or 2) to 61.

Error protection of the residual waveform was not considered to be viable. The residual waveform carries less information about the input speech in comparison to the side information. Since the RELP coder encodes the residual waveform into fixed size blocks, it should be relatively

more robust to channel errors than the residual waveform information for the APC coder, which is encoded into variable-length blocks.

The subsequent simulations of the RELP coder showed that its performance, in the presence of channel errors, was similar to that of the APC coder. This result is not surprising considering that the same error detection/correction methods are used for the two coders.

3.5.5 Final Configurations of the RELP Coder

Three configurations of the RELP coder were derived. The only difference between the three different versions of the coder is the amount of decimation used. The three decimation factors used were: 4, 3, and 2. The calculations of the bit rates for the three RELP coders are shown in Figure 3.5-10. The results are summarized below.

RELP coder 1 - decimation by 4, 7.5 kbit/s

RELP coder 2 - decimation by 3, 9.2 kbit/s

RELP coder 3 - decimation by 2, 12.6 kbit/s

Results of the simulations of all three coders are shown in Figure 3.5-11. Test speech sample one was used. As expected, the objective quality measures of the RELP coder increase as the decimation factor decreases (or bit rate increases).

segment size (N) = 180
 sampling rate = 6.666 kHz

Side Information + Overhead :

synchronization block :	8 bits
side information : (decimation by 4)	54 bits
(decimation by 3 or 2)	55 bits
parity bits :	6 bits
total :	68 bits
(decimation by 3 or 2)	69 bits

Residual Waveform :

decimation by 4	-	$\frac{3}{4} \times 180 = 135$ bits
by 3	-	$\frac{3}{3} \times 180 = 180$ bits
by 2	-	$\frac{3}{2} \times 180 = 270$ bits

Bit Rate :

decimation by 4	-	$\frac{(68 + 135)}{180} \times 6.666 \times 10^3 = 7.5$ kbit/sec
decimation by 3	-	$\frac{(69 + 180)}{180} \times 6.666 \times 10^3 = 9.2$ kbit/sec
decimation by 2	-	$\frac{(69 + 270)}{180} \times 6.666 \times 10^3 = 12.6$ kbit/sec

Figure 3.5-10

Calculation of the bit rate of the final configuration of the RELP coder.

	Coder 1	Coder 2	Coder 3
Signal-to-noise ratio (dB)	10.41	13.83	16.48
Segmental signal-to-noise ratio (dB)	10.98	12.85	14.67

Figure 3.5-11

Simulation performance of the three RELP coder configurations using test speech sample one.

Informal listening tests of all three coders showed that RELP coder 3 produced output speech with the most natural quality. There was very little discernable distortion. The speech for RELP coder 2 contained some tonal noise, sounding a bit "synthetic". RELP coder 1 produced speech that was the least natural sounding of the three coders. The tonal noise in the output speech of this coder was most apparent, producing a "rain-barrel" effect. All three RELP coders produced output speech of higher quality than either of the delta modulation coders. It was difficult to compare the RELP coders' output speech with that of the APC coders' because the distortion in the two coders sound quite different.

CHAPTER FOUR

OBJECTIVE AND SUBJECTIVE EVALUATIONS OF THE SPEECH CODERS

4.1 Acquisition of the Mobile Radio Channel Data

In the spring of 1983, a series of field measurements were carried out at Carleton University for the Department of Communications to study the characteristics of the digital land mobile radio channel under various conditions [56]. A brief description of the experiments that were carried out will now be given, with emphasis on those experiments that have relevance to this study.

The base station transmitter was a commercial (General Electric) FM unit rated at 30 watts output power. The 820 MHz base station antenna was a 2.5 m mast of vertically stacked quarter-wave coaxial elements with a gain of 8 dB. The antenna, located on the top of the Carleton University Mackenzie Engineering Building, was about 30 meters above ground level. The clutter objects in the immediate surrounding of the base station consisted mostly of buildings two to five stories in height, with the exception of the 24-storey Arts Tower to the west.

A data generator (Gandalf TTS400C) was used to generate a repetitive 2047 bit pseudo random binary sequence (PRBS) at 4800 bits/sec. A commercial radio modem (Mobile Data International) generated the digitally modulated baseband signal for transmission. A timing circuit was used to produce a 20 second ON/20 second OFF signal to control the data flow from the data generator to the FM transmitter. The data generator clock and ON/OFF timing signals were generated simultaneously at the transmitter and the receiver using two synchronized highly stable frequency standards.

The receiver was a commercial (General Electric) FM transceiver having a noise bandwidth of 11 kHz and a sensitivity of 0.35 μ V for 12 dB SINAD. The transceiver equipment was installed in a mobile van. A quarter-wave whip antenna was mounted on the roof of the van, about 3 meters above road level.

The output of the modem carries the binary received data and was recorded on one track of a 7-track FM recorder. On the other tracks of the FM recorder, other data such as the received signal envelope, data generator clock, and ON/OFF timing signal, were recorded.

Field data at 820 MHz was collected in four regions of the greater Ottawa area representing typical urban, suburban residential and highway terrains. A looping route was

chosen for each area, except for the Queensway. The aim was to maintain a relatively constant terrain type and distance to the transmitter. This was necessary to obtain data whose statistical variations would be primarily due to multipath fading and shadowing. About six minutes of recordings were made for each area.

4.2 Extraction of the Bit Error Patterns

The first step in processing the acquired field data involved sampling and storing the data recorded on the audio tape. The tapes are played back in real-time (30 IPS) and the recorded data generator clock provides timing for sampling the data at 4800 bits/sec. Data acquisition is synchronized to the 20-second intervals of data transmission. Sampling begins when a "start" pulse is detected from the ON/OFF timing signal recorded on one of the tape tracks, indicating the start of a data interval. The channel data, recorded on one of the FM tracks, is FM demodulated and stored in computer memory. A total of 90,000 samples, or 18.75 seconds of the interval is acquired. The data for one interval is then stored in a file on the Winchester hard disk during the 20-second unmodulated interval. A total of 36 data files were created from the tapes of the acquired field data.

The next processing step involved comparing the received data sequence with the known repetitive PRBS sequence. Any differences between the two sequences would then indicate where the bit errors occurred in the sequence. Before this comparison can be made the received data sequence must be synchronized with the known sequence. This is done by convolving the two sequences until a minimum convolution sum is found. In most of the files, there were few error bursts and a unique synchronization offset could be found. However, in a few files, portions of the data could not be synchronized properly and thus these portions were thrown out and not used in any further applications. The final error pattern was obtained by comparing the received data sequence with the known synchronized sequence and marking the differences as bits in error. Table 4.2-1 presents various statistics for the error patterns generated from the field data.

As mentioned earlier in this section, the bit rate of the pattern generator was 4800 bits/sec. The bit rates of the speech coders under simulation were quite variable, ranging from a low of 7.5 kbit/sec to a high of 16.1 kbit/sec. To use the bit error patterns in the speech coding simulations, a method was employed to approximately translate the bit error patterns to the higher sampling rate. Figure 4.2-1 illustrates a typical bit error pattern.

(d) Downtown Ottawa

File Name [NDT01.XX]	Probability of error [P(e)]	Average burst error length [b] (bits)	Standard deviation of b [σ_b] (bits)	Percentage of files recovered
.01	2.38×10^{-3}	5.5	3.6	100
.02	3.90×10^{-3}	9.9	14.2	100
.03	5.61×10^{-3}	8.9	9.1	100
.04	3.82×10^{-3}	8.3	9.7	100
.05	1.36×10^{-4}	9.1	8.8	100
.06	7.51×10^{-4}	9.6	8.7	100
.07	4.38×10^{-3}	8.6	7.4	100
.08	1.48×10^{-3}	8.3	5.4	100
.09	1.26×10^{-3}	8.4	9.8	100

(b) Downtown Hull

File Name [NDT09.XX]	Probability of error [P(e)]	Average burst error length [b] (bits)	Standard deviation of b [σ_b] (bits)	Percentage of files recovered
.01	8.65×10^{-3}	10.8	15.4	100
.02	7.51×10^{-2}	10.6	11.8	100
.03	1.16×10^{-3}	12.1	19.3	100
.04	7.36×10^{-3}	9.4	12.6	100
.05	5.78×10^{-3}	9.6	11.7	100
.06	4.80×10^{-3}	11.2	15.3	100
.07	5.05×10^{-3}	10.8	12.4	100
.08	6.29×10^{-3}	10.9	15.1	100
.09	7.50×10^{-3}	8.4	10.0	100

Table 4.2-1

Statistics of the error patterns generated from the mobile radio experimental field data.

(c) South Ottawa (residential area)

File Name [MOI13.XX]	Probability of error [P(e)]	Average burst error length [b] (bits)	Standard deviation of b [σ_b] (bits)	Percentage of files recovered
.01	1.07×10^{-1}	12.4	16.3	100
.02	2.18×10^{-2}	7.5	8.5	100
.03	9.88×10^{-2}	15.0	21.4	100
.04	1.54×10^{-1}	15.2	21.1	67
.05	1.48×10^{-1}	15.6	20.6	83
.06	2.09×10^{-2}	15.4	20.2	92
.07	8.27×10^{-1}	14.4	20.7	92
.08	3.11×10^{-1}	24.5	30.8	25
.09	1.81×10^{-1}	14.2	19.0	100

(d) Queensway

File Name [MOI12.XX]	Probability of error [P(e)]	Average burst error length [b] (bits)	Standard deviation of b [σ_b] (bits)	Percentage of files recovered
.01	4.78×10^{-2}	7.5	6.8	100
.02	5.66×10^{-3}	9.1	9.0	100
.03	2.12×10^{-3}	7.5	10.3	100
.04	7.27×10^{-2}	7.7	20.3	100
.05	2.35×10^{-2}	7.5	8.4	100
.06	4.79×10^{-2}	8.1	7.9	100
.07	3.87×10^{-2}	8.1	8.0	100
.08	1.93×10^{-2}	6.9	7.4	100
.09	5.47×10^{-2}	9.9	10.9	100

Table 4.2-1

Statistics of the error patterns generated from the mobile radio experimental field data.

(a) at 4800 bits/sec with two error bursts, starting at bit positions 5 and 11. If these error bursts occur at the higher sampling rate, shown in (b), bits 8 through 12 and 17 through 18 are entirely corrupted, while bits 7, 13, 16, and 19 are only partially corrupted. The entirely corrupted error bits are marked as errors. The partially corrupted bits are marked in error with a variable probability. The probability of being marked as an error is equal to the degree of corruption of the bit.

Random numbers between 0.0 and 100.0 are generated to make the decision. If a bit is 70% corrupted, then a generated random number between 0 and 70.0 will cause that bit to be marked as an error.

4.3 Results of the Objective Testing

This section describes the computer simulations that were run to objectively evaluate the eight final speech coder configurations. The objective measure used was the segmental SNR (see Appendix A). Speech data from two different speakers was used as input to the speech coder simulators. Five different channel conditions were also simulated. In total, eighty different simulations were performed. The speech output from the simulations was recorded on analog tape. It must be emphasized here that objective measures were not the only means taken to evaluate

Figure 4.2-1

Method to approximately translate a 4.8 kbit/sec error pattern (a) to a higher bit rate (b).

the performance of the speech coders. The definitive mechanism for noise and distortion measurement of the human voice is the human perception mechanism; therefore, more importance has been placed on the perceptual and subjective evaluations of performance for the speech coders. These subjective evaluations are detailed in sections 4.4 and 4.5.

Table 4.3-1 lists the eight final speech coder configurations and their respective bit rates. The background information on the three speech coders used to make up these eight configurations is provided in sections 3.3, 3.4, and 3.5. The last sub-section of these sections describes the final configurations of the coders in greater detail, providing the relevant set-up data derived during the optimization of the configurations.

The optimization of the eight speech coder configurations was performed using the test speech data described in section 3.1 (test speech sample one). For the objective evaluation of the speech coders, a different set of speech data was used. Two different speakers were chosen to see how the performances of the speech coders would be affected if speech data other than the training speech data were used. Six speech data files (three per speaker) were generated from a male and female speaker. Their respective average pitch periods were 7.1 msec and 4.0 msec. Recall, that the average pitch period of the training speech data

CVSDM

- 1) 12.0 kbit/sec
- 2) 16.1 kbit/sec

A3C

- 1) 8.6 kbit/sec - Total Residual Bits = 180
- 2) 11.6 kbit/sec - Total Residual Bits = 260
- 3) 14.6 kbit/sec - Total Residual Bits = 340

RELP

- 1) 7.5 kbit/sec - Decimation by 4
- 2) 9.2 kbit/sec - Decimation by 3
- 3) 12.6 kbit/sec - Decimation by 2

Table 4.3-1

Respective bit rates of the final configurations of the speech coders.

ERR1.DAT	:	Error-free channel
ERR2.DAT	:	Mobile radio channel, ABER = 7.3×10^{-4}
ERR3.DAT	:	Mobile radio channel, ABER = 2.1×10^{-3}
ERR4.DAT	:	Mobile radio channel, ABER = 7.4×10^{-3}
ERR5.DAT	:	Random error channel, ABER = 7.4×10^{-3}

Table 4.3-2

Five error files used to simulate five different error conditions. Average Bit Error Rate (ABER) is shown.

was 5.9 msec.

The method of generating the files and the sentences uttered were the same as those used to generate the test speech data. The unprocessed speech files and their respective sampling rates are:

MALE1.SPC	-	12.048 kHz
MALE2.SPC	-	16.128 kHz
MALE3.SPC	-	6.666 kHz
FEMALE1.SPC	-	12.048 kHz
FEMALE2.SPC	-	16.128 kHz
FEMALE3.SPC	-	6.666 kHz

Each file consists of approximately 7.4 seconds of speech.

A total of five different channel conditions were simulated: one error-free channel, three mobile radio channels, and one random error channel. The three mobile radio channel simulations were of different average error rates, all being below one percent. The average error rate of the random error channel simulation was the same as that of the mobile radio channel simulation with the highest average error rate. Thus, the robustness of the speech coders to various mobile radio channel conditions can be evaluated. As well, the performance of the speech coder to random-type errors and burst-type errors can be compared.

Three mobile radio channel error files were derived from the thirty-six bit error pattern files listed in Table

4.2-1. The first 7.4 seconds of these files were extracted. The average bit error rates and average burst error lengths were recalculated. The three files chosen were MDI01.07, MDI01.04, and MDI12.13. The average bit error rates $P(e)$ and average burst error lengths $\bar{b}$ of the first 7.4 seconds of these files are listed below.

MDI01.07 - $P(e) = 7.3 \times 10^{-4}$, $\bar{b} = 10.7$ bits

MDI01.04 - $P(e) = 2.1 \times 10^{-3}$, $\bar{b} = 9.8$ bits

MDI12.13 - $P(e) = 7.4 \times 10^{-3}$, $\bar{b} = 11.2$ bits

These three bit error patterns were chosen because their average bit error rates are relatively high, yet do not exceed one percent. A summary of the five channel conditions simulated and their respective average bit error rates is shown in Table 4.3-2.

The results of the simulations are provided in Table 4.3-3. The objective measure of performance shown in the table is the segmental SNR. Two types of graphs have been plotted from these results. Figures 4.3-1 and 4.3-2 contain plots of segmental SNR vs. average bit error rate and segmental SNR vs. bit rate, respectively, for all eight coder configurations. The graphs have been simplified by averaging the results for both the male and female speaker.

This concludes the presentation of the results of the objective testing. For a discussion of the results of the

(a) DM

Bitrate (kbit/sec)	Speaker	12.0		16.1	
		Male	Female	Male	Female
Error free		4.93	5.72	6.34	7.00
7.3×10^{-4}	M. R.	4.91	5.71	6.33	6.98
2.1×10^{-3}	M. R.	4.85	5.70	6.32	6.98
7.4×10^{-3}	M. R.	4.85	5.63	6.31	6.93
7.4×10^{-3}	Random	4.68	5.62	6.13	6.68
Average		4.84	5.62	6.29	6.91

(b) APC

Bitrate (kbit/sec)	Speaker	8.6		11.6		14.6	
		Male	Female	Male	Female	Male	Female
Error free		8.50	7.82	11.86	11.05	14.56	13.62
7.3×10^{-4}	M. R.	8.46	7.77	11.67	10.99	14.39	13.44
2.1×10^{-3}	M. R.	7.71	7.02	10.03	9.64	12.78	12.24
7.4×10^{-3}	M. R.	7.12	6.55	8.82	8.33	10.88	10.67
7.4×10^{-3}	Random	6.36	6.04	5.34	5.01	5.72	5.53
Average		7.63	7.04	9.54	9.00	11.67	11.10

(c) RELP

Bitrate (kbit/sec)	Speaker	7.5		9.2		12.6	
		Male	Female	Male	Female	Male	Female
Error free		8.25	8.01	9.88	9.11	11.39	10.84
7.3×10^{-4}	M. R.	7.84	7.26	9.63	8.38	10.93	10.21
2.1×10^{-3}	M. R.	6.74	6.44	8.10	7.25	9.37	8.75
7.4×10^{-3}	M. R.	4.99	3.04	6.55	4.42	7.60	5.86
7.4×10^{-3}	Random	4.75	3.54	5.04	3.63	6.12	4.82
Average		6.51	5.66	7.84	6.56	9.08	8.10

Table 4.3-3

Objective test results using the segmental SNR measure (in dB's).

Figure 4.3-1

Plot of Segmental SNR vs. Average Bit Error Rate for various mobile radio channel conditions.

Figure 4.3-1

Plot of Segmental SNR vs. Average Bit Error Rate for various mobile radio channel conditions.

- - DM
 ■ - APC
 ▲ - RELP

Figure 4.3-2

Plot of Segmental SNR vs. Bit Rate for various channel error conditions.

- - DM
- - APC
- ▲ - RELP

Figure 4.3-2

Plot of Segmental SNR vs. Bit Rate for various channel error conditions.

Figure 4.3-2

Plot of Segmental SNR vs. Bit Rate for various channel error conditions.

objective and subjective testing, refer to section 4.5

4.4 Results of the Subjective Testing

In the previous section (4.3), it was mentioned that subjective measures of speech quality are more relevant than objective measures such as the segmental SNR; the human perception mechanism is the definitive mechanism for measuring speech quality. This section describes the subjective quality tests that were performed using the output speech produced by the computer simulations. The reader should refer to section 4.3 for a detailed description of the test data used for these simulations.

In total, eighty different output speech segments, each consisting of 7.4 seconds of speech, were produced by the computer simulations and recorded on the reel-to-reel tape recorder (REVOX A77). To protect against possible effects of presentation order, it is necessary that the totality of the above stimuli be presented to the subjects in a random order [43]. Due to the limitations of the available apparatus for stimulus presentation, it was not possible to present a new random stimulus order for each subject. However, a compromise was made by recording four sets of the eighty speech segments presented in a different random order. Each subject was allowed to listen to one of the four recordings in a sitting session. Each recording was

used in a rotational order for each new subject so that all four recordings were used the same number of times.

Figure 4.4-1 shows a block diagram of the set-up used for the subjective testing of the speech coders. The four recordings were made on cassette tape (TDK-SA90) using a commercial cassette tape deck (Harmon Kardon CD91) for ease of use and portability reasons. A Northern Telecom Model 500 telephone handset was used to listen to the speech recordings. An attenuator was placed between the headphone output of the cassette deck and the telephone handset, allowing the subject to adjust the sound output to a comfortable level.

Each output speech segment was presented, followed by about 1.6 seconds of silence, during which time the subject was expected to classify the output speech quality using a five-point quality scale and write down this result on the score sheet provided. The five-point quality scale used is shown in Figure 4.4-2 [43]. An alternative five-point impairment scale could also be used by the subjects to rate the speech segments.

The subject was first asked to listen to a small subset of the speech output segments that were considered to be representative of the entire set of eighty speech output samples. During this time, the subject was asked not to

Figure 4.4-1

Subjective testing set-up.

Score	Quality Scale	Impairment Scale
5	Excellent	Imperceptible
4	Good	Just perceptible but not annoying
3	Fair	Perceptible and slightly annoying
2	Poor	Annoying but not objectionable
1	Unsatisfactory (bad)	Very annoying

Figure 4.4-2

Five-point quality scale or alternative five-point impairment scale.

record any scores. Finally, the actual testing was started, using one of the four recordings. Each recording lasted 12 minutes.

A total of twelve different subjects performed the subjective tests; four being female, and eight being male. Four of the subjects were engineers, and had some knowledge of speech coding, while the other eight were non-engineers, having no knowledge of the work being done. Thus, a good cross-section of subjects was used.

The mean opinion score (MOS) was calculated for each speech output segment by taking the arithmetic mean of the scores gathered. The results of the MOS calculations are shown in Table 4.4-1. Two types of graphs have been plotted from these results. Figures 4.4-3 and 4.4-4 contain plots of mean opinion score vs. average bit error rate and mean opinion score vs. bit rate, respectively, for all eight coder configurations. Once again, the graphs have been simplified by averaging the results for the male and female speakers.

This concludes the presentation of the results of the subjective testing. The next section discusses the results of both the objective and subjective testing.

(a) DM

Bitrate (kbit/sec)	Speaker	12.0		16.1	
		Male	Female	Male	Female
Error free		1.9	1.8	2.2	2.2
7.3×10^{-4} M. R.		2.2	2.0	2.4	2.1
2.1×10^{-3} M. R.		2.1	1.9	2.2	2.2
7.4×10^{-3} M. R.		1.9	1.8	2.1	2.2
7.4×10^{-3} Random		1.9	1.7	2.0	1.8
Average		2.0	1.8	2.2	2.1

(b) APC

Bitrate (kbit/sec)	Speaker	8.6		11.6		14.6	
		Male	Female	Male	Female	Male	Female
Error free		2.6	2.7	3.0	3.0	3.2	3.5
7.3×10^{-4} M. R.		2.8	2.4	3.1	2.6	3.1	3.5
2.1×10^{-3} M. R.		2.6	2.7	2.9	2.9	3.1	3.2
7.4×10^{-3} M. R.		1.8	1.4	2.5	2.3	3.0	2.6
7.4×10^{-3} Random		2.1	2.1	2.6	2.2	2.9	2.5
Average		2.4	2.3	2.8	2.6	3.1	3.1

(c) RELP

Bitrate (kbit/sec)	Speaker	7.5		9.2		12.6	
		Male	Female	Male	Female	Male	Female
Error free		2.1	1.7	2.6	1.9	3.6	3.6
7.3×10^{-4} M. R.		2.0	1.9	2.7	2.1	3.3	3.3
2.1×10^{-3} M. R.		2.2	1.9	2.8	1.7	3.3	3.4
7.4×10^{-3} M. R.		1.8	2.0	2.2	2.0	2.9	2.6
7.4×10^{-3} Random		1.9	1.8	2.1	1.6	2.5	2.2
Average		2.0	1.9	2.5	1.9	3.1	3.0

Table 4.4-1 (a)

Mean opinion score (MOS) subjective test results.

(a) DM

Bitrate (kbit/sec)	Speaker	12.0		16.1	
		Male	Female	Male	Female
Error free		0.66	0.55	0.58	0.76
7.3×10^{-4}	M. R.	0.76	0.53	0.52	0.52
2.1×10^{-3}	M. R.	0.43	0.50	0.37	0.49
7.4×10^{-3}	M. R.	0.32	0.37	0.52	0.71
7.4×10^{-3}	Random	0.66	0.61	0.88	0.55

(b) APC

Bitrate (kbit/sec)	Speaker	8.6		11.6		14.6	
		Male	Female	Male	Female	Male	Female
Error free		0.53	0.70	0.73	0.60	0.55	0.49
7.3×10^{-4}	M. R.	0.70	0.66	0.64	0.75	0.72	0.65
2.1×10^{-3}	M. R.	0.45	0.51	0.52	0.69	0.59	0.63
7.4×10^{-3}	M. R.	0.41	0.88	0.64	0.82	0.73	0.75
7.4×10^{-3}	Random	0.52	0.66	0.61	0.80	0.52	0.96

(c) RELP

Bitrate (kbit/sec)	Speaker	7.5		9.2		12.6	
		Male	Female	Male	Female	Male	Female
Error free		0.66	0.76	0.48	0.60	0.49	0.63
7.3×10^{-4}	M. R.	0.73	0.69	0.57	0.32	0.59	0.67
2.1×10^{-3}	M. R.	1.01	0.65	0.70	0.42	0.78	0.64
7.4×10^{-3}	M. R.	0.37	0.60	0.49	0.60	0.64	0.77
7.4×10^{-3}	Random	0.60	0.55	0.60	0.63	0.67	0.58

Table 4.4-1 (b)

Standard deviations of the mean opinion score
subjective test results.

Figure 4.4-3

Plot of Mean Opinion Score vs. Average Bit Error Rate for various mobile radio channel conditions.

Figure 4.4-

Plot of Mean Opinion Score vs. Average Bit Error Rate
for various mobile radio channel conditions.

- - DM
- - APC
- ▲ - RELP

Figure 4.4-4

Plot Mean Opinion Score vs. Bit Rate for various channel error conditions.

- - DM
- - APC
- ▲ - RELP

Figure 4.4-4

Plot Mean Opinion Score vs. Bit Rate for various channel error conditions.

Figure 4.4-4

Plot Mean Opinion Score vs. Bit Rate for various channel error conditions.

4.3.5 Discussion of the Results

Subjective quality testing can be very time consuming, while objective quality testing takes much less time and effort to perform. The ideal objective quality test would be one that could take the place of subjective tests, being valid across many different types of coders. The objective measure for such a test does not, as yet, exist. The segmental SNR measure works relatively well with waveform-type speech coders but, not with other speech coders such as vocoders. Thus, the segmental SNR measure would not be a meaningful quality measure for the RLP coder since it does incorporate some vocoding techniques. Also, the functional differences between all three speech coders under study, are large enough to expect corresponding differences in performance with respect to the segmental SNR measure. For example, an equal drop in voice quality for two different speech coders might not reflect an equal drop in segmental SNR.

Keeping in mind the inadequacies of the SNR measure as a subjectively meaningful measure of speech quality, the results of the objective and subjective testing will now be discussed. In these discussions, the results of the subjective testing will be given more importance, while the objective testing results will be used, whenever possible; to back up these results. The aim of this discussion is to

explore the capabilities of the different speech coding algorithms with regards to the requirements for the mobile radio channel.

A number of intuitive results can be seen from the graphs of the objective and subjective results. From the graphs of Figure 4.4-4, showing plots of Mean Opinion Score vs. Bit Rate, we can see that coder performance generally increases as the bit rate of the coder is increased. Secondly, from Figure 4.4-3, coder performance generally decreases as the average bit error rate is increased. These last two results are not surprising since this type of behaviour is observed, generally, in all speech coders. Of all the three speech coders, the DM coder gave the best performance in the presence of channel errors, being more robust than either the APC or RELP coders. This result is not unexpected as there is inherently more redundancy built into the data sent on the channel by the DM coder as compared to the other two coders. This redundancy is what makes the DM coder more robust. However, the objective and subjective qualities for both versions of the DM coder are less than that of the great majority of the other speech coder implementations tested.

For the error-free channel condition, only the 8.6 kbit/sec. APC coder has a lower subjective quality than the highest quality DM coder, whose bit rate is 16.1 kbit/sec.

As mobile radio channel errors are introduced, the subjective quality of the APC and RELP coders start to drop to the same level as the DM coder. At the highest average bit error rate of 7.4×10^{-3} , only the two highest bit rate RELP coders and the highest bit rate APC coder have a higher MOS score than both DM coders.

From Table 4.4-1, one can see that the MOS scores for all three coders are generally slightly higher for the male speaker compared to the female speaker. For the objective tests, only the DM coder achieved better results for the female speaker. However, the disparity of these results is not enough to suggest any coder performance bias towards either gender. A review of previous results of all three coders suggests that a bias can exist for the RELP coder, where the performance was found to be better for lower-pitched male voices [59].

The performance of all three speech coders in the presence of random-type error patterns is generally worse than in the presence of the burst-type mobile radio error patterns. This result holds true for both the objective and subjective quality tests. For the DM coder, which does not use any error protection, this result is less apparent. However, for the DM coder, one would expect that burst-type errors would cause distortions in the output speech similar to slope overload distortion, while random-type errors would

cause distortions similar to granular noise. Since slope overload distortion is often less perceptible than granular noise in DM coders, it follows that burst-type errors would be less perceptible than random-type errors of the same average error rate.

Recall, that both the APC and RELP coders use a side information replacement strategy to cope with channel errors. It was reported in sub-section 3.4.7 that the replacement strategy worked better for burst-type errors as compared to random-type errors of the same average error rate. The results from the objective and subjective testing agree, showing that the quality of both the RELP and APC coders is better in the presence of mobile radio channel errors.

From Table 4.4-2, we see that the speech coder with the highest mean opinion score was the 14.6 kbit/sec APC coder with a score of 3.6. By comparison, speech with a MOS score between 4.0 and 4.5 is generally believed to be of telephone-toll quality [43]. Thus, none of the coders tested were able to produce speech of telephone-toll quality.

An interesting development that can be seen from Table 4.4-2 is that, for most of the error conditions tested, the next three speech coders ranking below the 14.6 kbit/sec APC

Error Pattern	Coder	Bit Rate (kbit/sec)	MOS	Error Pattern	Coder	Bit Rate (kbit/sec)	MOS
Error Free Channel	APC	14.6	3.6	7.4×10^{-3} Mobile Radio	APC	14.6	2.8
	RELP	12.6	3.4		RELP	12.6	2.8
	RELP	9.2	3.0		RELP	9.2	2.4
	RELP	7.5	2.7		APC	11.6	2.1
	APC	11.6	2.3		DM	16.1	2.1
	DM	16.1	2.2		APC	8.6	1.9
	APC	8.6	1.9		DM	12.0	1.8
	DM	12.0	1.8		RELP	7.5	1.6
7.3×10^{-4} Mobile Radio	APC	14.6	3.3	7.4×10^{-3} Random Errors	RELP	12.6	2.7
	RELP	12.6	3.3		RELP	9.2	2.4
	RELP	9.2	2.9		APC	14.6	2.3
	RELP	7.5	2.6		RELP	7.5	2.1
	APC	11.6	2.4		APC	11.6	1.9
	DM	16.1	2.3		DM	16.1	1.9
	DM	12.0	2.1		APC	8.6	1.8
	APC	8.6	2.0		DM	12.0	1.8
2.1×10^{-3} Mobile Radio	APC	14.6	3.4				
	RELP	12.6	3.2				
	RELP	9.2	2.9				
	RELP	7.5	2.6				
	APC	11.6	2.2				
	DM	16.1	2.2				
	APC	8.6	2.0				
	DM	12.0	2.0				

Table 4.4-2

Mean Opinion Score rankings for various channel error conditions.

coder in quality, are the three versions of the RELP coder. In fact, the 7.5 kbit/sec RELP coder has a better MOS score than the 11.6 kbit/sec APC coder for three out of the five trials. This is true in spite of the fact that the bit rate of the RELP coder is more than 35 percent below that of the APC coder. At the two highest average error rates tested, the 12.6 RELP coder came out on top and tied for top spot, respectively, showing that it exhibits a comparatively good degree of robustness.

The overall performance of the APC speech coder was disappointing. Although the highest quality speech coder, as determined from the subjective quality tests, was the 14.6 kbit/sec APC coder, the other two versions of the APC coder languished in the bottom half the MOS score rankings. In fact, the 8.6 kbit/sec APC coder did not out score the 16.1 kbit/sec DM coder at any of the error conditions tested. The robustness of the APC coder to channel errors is similar to the RELP coder. At the highest average bit error rate applied, the APC coder's performance degrades to the same level as the DM coder.

The relatively poor performance of the DM coder was not unexpected. The MOS score ratings for the DM coder were consistently the lowest scores received for the three coders. Although the DM coder was indeed robust to channel errors, even the speech produced by the DM coder on

the error-free channel was found to be unacceptable.

The performance of the RELP speech coder was the best overall of the three coders. The three versions of the RELP coder placed high in the MOS rankings for all of the error conditions tested. Although the RELP coder was found to be less robust to channel errors than the DM coder, its MOS score remained relatively high even at the highest average error rate tested (7.4×10^{-3}).

The results from the subjective testing performed in this study will now be compared with results found in other studies. Since the testing methods and conditions could never be expected to be exactly the same between two separate studies, more emphasis will be placed on comparing global results rather than comparing specific quantitative results..

In one study, Svean subjectively evaluated four speech coders [11]. Two of the four speech coders he tested, the APC and RELP coders, were also tested in this study. Two of his RELP coders, had transmission rates close to two of the RELP coders of this study. His 9.9 kbit/sec and 7.2 kbit/sec RELP coders achieved MOS scores of 3.0 and 2.3, respectively. The three APC coders that he tested had bit rates of 16.0, 12.0 and 9.6 kbit/sec, scoring 3.2, 2.2 and 2.1, respectively. Thus, the MOS scores of the RELP and APC

coders of this study and Svean's study are quite close. Svean also found that the performance of the RELP coder came close to that of the APC coder, despite the fact that the bit rate of the RELP coder is almost 6 kbit/sec lower than that of the APC coder.

In another study, Daumer subjectively evaluated six different coders, including the APC coder [37]. The 12 kbit/sec and 16 kbit/sec versions of APC received MOS scores of 3.4 and 3.9, respectively, with no added channel impairments. In this study, the 14.6 kbit/sec. version of APC received a MOS score of 3.6, falling within the range of Daumer's two APC coders.

Before finishing this section, a few comments should be made on the MOS score standard deviation values listed in Table 4.4-1(b). One can see that these values range from a low of 0.32 to a high of 1.01. The large magnitude of these values is due to a number of reasons, such as coder presentation interactivity and coarseness of the five-point quality scale. In his test, Daumer's standard deviation range was between 0.6 and 0.9. Thus, the values obtained in this study don't seem out of line.

4-6 Conclusions and Recommendations of a Digital Speech Coding Algorithm for the Mobile Radio Channel

In the previous section, the results of the objective and subjective testing were discussed. In this section, a number of conclusions will be drawn on the work done in this study. At the end of this section, a recommendation will be made as to which digital speech coding algorithm would be best suited for the mobile radio channel.

The results of the subjective testing confirmed what was found for the objective tests: the continuously variable slope delta modulation speech coder did not produce acceptable quality reconstructed output speech at transmission rates equal to or below 16 kbit/sec. The mean opinion scores for the two versions of the DM coder without added channel impairments averaged around 2.0, indicating that the speech was of poor quality. Although the CVSDM coder is relatively simple, compared with the RELP and APC coders, and is quite robust to channel errors of up to one percent, its speech output quality is clearly inadequate for use in a mobile radio application.

Only the highest bit rate version of the adaptive predictive coder, transmitting at 14.6 kbit/sec, produced output speech approaching acceptable quality. The 14.6 kbit/sec APC coder achieved a MOS score of 3.6 with no channel impairments, indicating that the speech it produces

is close to good quality. Indeed, informal listening tests indicated that only a very small amount of background noise was perceptible.

The other two versions of the APC coder, transmitting at 8.6 and 11.6 kbit/sec, produced output speech of unacceptable quality. The MOS scores for these coders were all below 3.0, indicating the speech was of fair to poor quality. The background noise in these two coders was quite perceptible, to the point of being annoying.

The robustness of the APC coder to channel impairments was found to be the worst of all three coders, being slightly below that of the RELP coder. At high average error rates near one percent, the speech quality of the APC coder degrades to the same level as the DM coder, and is therefore, unacceptable.

The complexity of the APC coder is probably the greatest of all three coders. The greatest difference between the implementation of the APC and RELP coders is in the way that the residual waveform is quantized and encoded. The APC coder uses variable length coding to quantize and encode the residual, while the RELP coder uses a decimation process which consists of a simple lowpass filtering operation. Therefore, the APC coder is slightly more complex.

The two highest bit rate versions of the residually excited linear prediction coder, transmitting at 9.2 and 12.6 kbit/sec, produce output speech approaching acceptable quality with no channel impairments. The MOS scores for these two coders were between 3.0 and 4.0, indicating that the output speech was of fair to good quality. Informal listening tests indicated that of these two coders, only the output speech of the 9.2 kbit/sec RELP coder had noticeable voice-dependent distortions. This kind of distortion has been previously named "tonal noise" and gives the speech a slightly synthetic quality [59].

Unlike the APC coder, the RELP coder does not degrade to the same level of performance as the DM coder when channel errors of up to one percent are added. Not only is the 12.6 kbit/sec RELP coder slightly more robust than the 14.6 kbit/sec APC coder, but it also transmits at 2 kbit/sec less bit rate and is slightly less complex to implement.

In conclusion, based on the evidence presented in this section, the overall winner is the 12.6 kbit/sec version of the RELP coder. This coder produces output speech of fair to good quality with no channel impairments. The RELP coder also exhibits a slight degree of robustness to channel errors of up to one percent, maintaining output speech of fair quality. The RELP coder is also slightly less complex to implement than an APC coder. Thus, the RELP coder meets

some of the basic requirements for application with the mobile radio channel. There is still an extra 3.4 kbit/sec of bit rate available for the RELP coder if a transmission rate of 16 kbit/sec is allowable. Section 5.2 suggests several methods of using this available bandwidth for increasing the quality of the output speech and robustness of the RELP coder.

CHAPTER FIVE

SUMMARY AND SUGGESTIONS FOR FURTHER RESEARCH

5.1 Summary

Three digital speech coding algorithms were subjectively evaluated under a number of different channel conditions with the objective of determining which algorithm would be best suited for application to the mobile radio channel. The steps carried out to arrive at this final goal are presented below:

- 1) A study of both the mobile radio channel and available speech coding algorithms uncovered the following requirements: because of bandwidth limitations, the transmission rate of the coder should be less than 16 kbit/sec; for portability reasons, and the need to implement in VLSI, the coding algorithm should not be relatively complex; the coder should be relatively robust to mobile radio channel conditions which are characterized by burst-type errors and high average error rates of up to one percent; the decoder output speech quality should be relatively good even in the presence of these conditions. Based upon these requirements, the deltamodulation (DM) coder, adaptive predictive coder (APC), and residually

excited linear predictive (REL P) coder were chosen for study.

2) The operation of eight versions of the three different speech coding algorithms, along with five different channel conditions were simulated on a microcomputer. Three of the five channel conditions were derived from actual land mobile radio experimental data. Both male and female speakers were used as speech input to the coders.

3) Subjective tests were carried out using the "mean opinion score" method. Twelve subjects were asked to evaluate each of the eighty randomly presented speech samples on a quality scale from one to five using a telephone handset as the listening device.

4) An evaluation of the objective and subjective results was carried out to determine which one of the three speech coding algorithms would be most suitable for a mobile radio application.

The results show that although the DM coder is a relatively simple coder that is robust to mobile radio type errors, the output speech quality that it produces is unacceptable at transmission rates of less than 16 kbit/sec.

Only the highest bit rate version of the adaptive predictive coder, transmitting at 14.6 kbit/sec, produced output speech approaching acceptable quality. The speech

3 OF 3 DE 3

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

quality was close to being good with no channel impairments. The robustness of the APC coder to channel errors was found to be the worst of all three coders, being slightly below that of the RELP coder. The APC coder is also the most complex coder, being slightly more so than the RELP coder.

The two highest bit rate versions of the RELP coders transmitting at 9.2 and 12.6 kbit/sec. produce output speech approaching acceptable quality with no channel impairments. The performance of the RELP coder in the presence of channel errors is slightly better than that of the APC coder. The 12.6 kbit/sec RELP coder was able to produce output speech of slightly less than fair quality with mobile radio channel errors of up to 7.4×10^{-3} .

In conclusion, the digital speech coding method found most suitable for use on the mobile radio channel is the RELP speech coder. The RELP coder meets some of the basic requirements of speech quality, complexity, transmission rate, and robustness to burst-type errors of average error rates up to one percent.

5.2 Suggestions for Further Research

The search for a suitable speech coder for mobile radio is by no means a simple task since the field is constantly changing. Innovations in speech coding algorithms and digital signal processing hardware, as well as new knowledge about the characteristics of the mobile radio channel are on the verge of radically transforming the mobile radio field. Thus, although it is felt that the studies carried out for this thesis were extensive, they were by no means exhaustive. Some suggestions will now be made as to how the work done in this thesis can be improved and extended.

The amount of subjective testing performed in this thesis could be expanded and improved upon. For this thesis, a total of twelve subjects and two source speech samples were used, giving a total of 24 votes cast on each test condition. By comparison, Daumer was able to obtain 294 votes cast on each test condition for his subjective testing [37]. Also, reference conditions, such as random analog noise and speech-dependent distortion, could be incorporated into the subjective testing so that the results can be compared more reliably with other subjective test results using the same conditions. More extensive results could be obtained concerning the effects of mobile radio channel errors on the operation of the speech coders if

additional error data were incorporated into the subjective testing.

A number of suggested improvements to the RELP speech coder could be investigated. One such improvement would involve increasing the bit rate of the coder to 16 kbit/sec from 12.6 kbit/sec. The extra 3.4 kbit/sec would allow one to try one or more of the following improvements: increase the prediction order to make the residual spectrum more white; use a non-integer, and thus, lower decimation factor; use a non-integer, and ultimately larger, number of bits per sample for baseband residual quantization. The increase in bit rate would effect an increase in the speech output quality.

The performance of the RELP coder, with respect to channel errors, could be improved by employing more complex error protection of the side information data block. In this study, a simple error detection code was used to replace corrupted side information data blocks with the last uncorrupted side information data block. This algorithm was found to work well, except for unvoiced/voiced transitions.

An improved replacement algorithm exists, whereby parameters contained within the corrupted data block (n), are replaced by a set of parameters that are linearly interpolated between data blocks (n-1) and (n+1). This

algorithm has been shown to produce an improvement in vocoder speech corrupted by burst-type channel errors [57]. However, a full frame of delay must be introduced. This algorithm also assumes that two frames in a row cannot be corrupted by errors.

A fairly ambitious follow-up to the work done in this thesis might be the implementation of the RELP speech coder in VLSI, with the aim of incorporating the resulting circuit into a portable transceiver for a mobile radio application. The first step towards such a goal might be the implementation of the RELP coder using commercial DSP devices.

In the past, an APC coder has been realized using three NEC 7720 DSP devices on a printed circuit board, occupying more than 19 square inches [28]. By comparison, a RELP speech coder has been implemented using two TI TMS32010 DSP devices on a pc board, occupying less than 35 square inches [29].

APPENDIX A

OBJECTIVE MEASUREMENTS OF CODER PERFORMANCE

Several objective (quantitative) measures exist which can be used to evaluate the performance of speech coders. Most of these measures have the general form of a signal-to-noise ratio (SNR). The two methods that have been used in this study are: long-term method (SNR) and segmental or short-term method (SEGSNR).

The signal-to-noise ratio (SNR) is defined as

$$\text{SNR} = (\sigma_x^2 / \sigma_r^2)$$

This is the ratio of input signal variance to reconstruction error variance. The reconstruction error signal is defined as

$$r(n) = x(n) - y(n)$$

where $x(n)$ is the input waveform and $y(n)$ is the output waveform. The SNR is usually expressed in decibels (dB) as

$$\text{SNR(dB)} = 10 \log_{10} (\sigma_x^2 / \sigma_r^2)$$

The segmental SNR (SEGSNR) compensates for the underemphasis of weak-signal (low σ_x^2) performance of the SNR calculation. The SEGSNR is based on log-weighting that

converts component (ie. segmental) SNR values to dB values prior to averaging. The SEGSNR is defined as

$$\text{SEGSNR(dB)} = E [\text{SNR}_m(\text{dB})]$$

where SNR_m is the SNR for segment m . The expectation E is in practice a time average over all relevant segments of the input signal.

The segmental SNR has been found to correlate better with subjective perceptual judgments than the SNR method [43]. For this reason, the segmental SNR method was used whenever intra-coder (ie. between different configurations of the same coder) testing was to be done. Thus, the SEGSNR was used as a measure during the long period of the coders' optimization. During this period decisions were made for each of the three coders as to which coding features were to be added and what values parameters were to take on. Thus, the final configuration of the speech coder was assured of having the highest SEGSNR value.

The proceeding method for intra-coder testing is less time-consuming than using subjective testing methods. However, subjective testing methods were used for the final inter-coder (ie. between different coders) comparisons as there is not a good correlation between SNR or SEGSNR values and subjective quality.

REFERENCES

- [1] Department of Communications, "Spectrum Allocation Policy in the 406-960 MHz Frequency Band". February, 1979.
- [2] J. Wengler, "Broad Use Seen for Narrowband", Electronics Week, pp. 28-30, April 15, 1985.
- [3] W.G. Jakes, Jr., Microwave Mobile Communications, John Wiley and Sons, New York, 1974.
- [4] C.S. White, "Mobile Communications - Stalled", Telecommunications, March, 1984.
- [5] R. Wells, "SSB for VHF Mobile Radio at 5 kHz Channel Spacing", IERE Conf. Proc. Radio Receivers and Associated Syst., Southampton, pp. 29-36, July, 1978.
- [6] I.M. Bohm, "Mobile Telephone for Everyone Through Digital Technology", Telecommunications, pp. 68-72, October, 1985.
- [7] F. de Jager, C.B. Dekker, "Tamed Frequency Modulation - A Novel Approach to Achieve Spectral Economy in Digital Transmission", IEEE Trans. Commun., vol. 26, pp. 534-542, May, 1978.
- [8] K. Murota and K. Hirade, "GMSK Modulation for Digital Mobile Radio Telephony", IEEE Trans. Commun., vol. 29, pp. 1044-1050, July, 1981.
- [9] H. Suzuki, K. Momma, Y. Yamao, "Digital Portable Transceiver Using GMSK Modem and ADM Codec", IEEE Trans. Vehic. Tech., vol. 33, no. 3, pp. 220-226, August, 1984.
- [10] M. McLaughlin, D. Linder, and S. Carney, "Design and Test of a Spectrally Efficient Land Mobile Communications System Using LPC Speech", IEEE Trans. Vehic. Tech., vol. 33, no. 3, pp. 227-236, August, 1984.
- [11] J. Svean, "Quality Measurements on Speech Coders for Mobile Radio", ICASSP 1982.
- [12] B.S. Atal, and M.R. Schroeder, "Predictive Coding of Speech Signals and Subjective Error Criteria", IEEE Trans. ASSP, vol. 27, no. 3, June, 1979.

- [13] O.C. Yue, "Spread Spectrum Mobile Radio, 1977-1982", IEEE Trans. Vehic. Tech., vol. 32, no. 1, pp. 98-105, February, 1983.
- [14] R.K. Chen, "ISDN - The Network of the Future", Telecommunications, pp. 45-68, December, 1983.
- [15] R.C. French, "Radio Propagation in London at 462 MHz", Radio and Electronics Engineer, vol. 46, pp. 333-336, July, 1976.
- [16] M. Schwartz, et al., Communication Systems and Techniques, McGraw Hill, New York, 1966.
- [17] J.L. Flanagan, et al., "Speech Coding", IEEE Trans. Commun., vol. 27, no. 4, pp. 710-733.
- [18] CCITT Study Group XVIII, "32-kbps Adaptive Differential Pulse Code Modulation (ADPCM)", Draft Recommendation G. 722, 31pp., November, 1983.
- [19] R.E. Crochiere, D.J. Goodman, L.R. Rabiner, M.R. Sambur, "Tandem Connections of Wideband and Narrowband Speech Communications Systems : Part 1 - Narrowband-to-Wideband Link", The Bell System Tech. J., November, 1977.
- [20] A. Gersho, and V. Cuperman, "Vector Quantization : A Pattern Matching Technique for Speech Coding", IEEE Communications Magazine, August, 1984.
- [21] J.L. Flanagan, K. Ishizaka, and J.L. Shipley, "Synthesis of Speech from a Dynamic Model of the Vocal Cords and Vocal Tract", Bell Syst. Tech. J., vol. 54, no. 3, pp. 485-506, March, 1975.
- [22] B.S. Atal and M.R. Schroeder, "Adaptive Predictive Coding of Speech Signals", Bell Syst. Tech. J., vol. 49, pp. 1973-1986, October, 1970.
- [23] R.E. Crochiere and J.L. Flanagan, "Sub-band Encoding of Speech", IEEE Int. Conf. Commun., June 1977.
- [24] R. Zelinski and P. Noll, "Adaptive Transform Coding of Speech Signals", IEEE Trans. ASSP, vol. 25, pp. 299-309, August, 1977.
- [25] B.S. Atal and J.R. Remde, "A New Model of LPC Excitation for Producing Natural-sounding Speech at Low Bit Rates", Proc. Int. Conf. ASSP, pp. 614-617, 1982.

- [26] V.R. Viswanathan, A.L. Higgins, and W.H. Russel, "Design of a Robust Baseband LPC Coder for Speech Transmission Over 9.6 kbit/s Noisy Channels", IEEE Trans. Commun., vol. 30, no. 4, April 1982.
- [27] B.C. Cole, "The Big News at FSSCC : Digital Signal Processors", Electronics, pp. 50-52, Dec. 23, 1985.
- [28] S. Tanaka, et al, "A 16 KBPS APC Codec Using Digital Signal Processors", ICASSP 1984.
- [29] M. Dankberg, R. Iliris, D. Saxton, and P. Wilson, "Implementation of the RELP Vocoder", ICASSP 1984.
- [30] R.B. Hanes, F.A. Westall, and C. Goody, "A 16 kbit/s Speech Coder Using a Single DSP Device", ICASSP 1984.
- [31] Y. Wake, et al, "A Multi-Pulse LPC Speech Codec Using Digital Signal Processors", ICASSP 1985.
- [32] R.D. Hangartner, V.K. Jain, "32 kbit/sec ADPCM/PCM Transcoder Using TI-320 Microprocessor", ICASSP 1985.
- [33] Z.A. Putnins et al, "A Multi-Pulse LPC Synthesizer for Telecommunications Use", ICASSP 1985.
- [34] T. Nishitani et al, "A CCITT Standard 32 KBPS ADPCM LSI CODEC", ICASSP 1985.
- [35] G. Davidson et al, "Real-Time Speech Compression With a VLSI Vector Quantization Processor", ICASSP 1985.
- [36] B. Fette, C. Rimpo, J. Kish, "A Manpack Portable LPC 10 Vocoder", ICASSP 1984.
- [37] W.R. Daumer, "Subjective Evaluation of Several Efficient Speech Coders", IEEE Trans. Commun., vol. 30, no. 3, April, 1982.
- [38] P. Cummsky, N.S. Jayant and J.L. Flanagan, "Adaptive Quantization in Differential PCM Coding of Speech", B.S.T.J., vol. 52, pp. 1105-1113, September 1973.
- [39] R.V. Cox and R.E. Crochiere, "Real-Time Simulations of Adaptive Transform Coding", IEEE Trans. ASSP, vol. 29, pp. 147-154, April, 1981.
- [40] R.E. Crochiere, R.V. Cox, and J.D. Johnston, "Real-Time Speech Coding", IEEE Trans. Commun., vol. 30, no. 4, pp. 621-634, April 1982.

- [41] A.H. Gray Jr. and J.D. Markel. "Linear Prediction Analysis Programs (AUTO-COVAR)", Programs for Digital Signal Processing, IEEE Press, John Wiley & Sons, Inc., New York, 1979.
- [42] E.M. De Loraine, S. van Miero, and B. Derjarvitch, French Patent 932, 140, 1946.
- [43] N.S. Jayant, P. Noll, Digital Coding of Waveforms : Principles and Applications to Speech and Video, Prentice-Hall, Inc., New Jersey, 1984.
- [44] G.L. Baldwin and S.K. Tewksbury, "Linear Delta Modulator Integrated Circuit with 17 Mb/s Sampling Rate", IEEE Trans. Commun., pp. 977-985, July 1974.
- [45] N.S. Jayant and A.E. Rosenberg, "The Preference of Slope Overload to Granularity in the Deltamodulation of Speech", Bell Syst. Tech. J., pp. 3117-3125, December, 1971.
- [46] A. Tomozawa and H. Kaneko, "Companded Delta Modulator for Telephone Transmission", IEEE Trans. Commun. Tech., pp. 149-157, February, 1968.
- [47] R. Steele, Delta Modulation Systems, Halsted Press, New York, 1975.
- [48] L.R. Rabiner and R.W. Schafer, Digital Processing of Speech Signals, Prentice-Hall, Inc., New Jersey, 1978.
- [49] J.D. Markel and A.H. Gray, Jr., Linear Prediction of Speech, Springer-Verlag, New York, 1976.
- [50] P. Noll, "A Comparative Study of Various Schemes for Speech Encoding", Bell Syst. Tech. J., vol. 54, no. 9, pp. 1597-1614, November, 1975.
- [51] B.S. Atal, "Predictive Coding of Speech at Low Bit Rates", IEEE Trans. Commun., vol. 30, no. 4, April, 1982.
- [52] L.R. Rabiner, M.J. Cheng, A.E. Rosenberg and C.A. McGonegal, "A Comparative Performance Study of Several Pitch Detection Algorithms", IEEE Trans. ASSP, pp. 399-418, October, 1976.
- [53] J. Makhoul and M. Berouti, "Adaptive Noise Spectral Shaping and Entropy Coding in Predictive Coding of Speech", IEEE Trans. ASSP, vol. 27, no. 1, February 1979.

- [54] A.H. Gray, Jr. and J.D. Markel. "Quantization and Bit Allocation in Speech Processing". IEEE Trans. ASSP. vol. 24. pp. 459-473. December 1976.
- [55] B.S. Atal and M.R. Schroeder. "Improved Quantizer for Adaptive Predictive Coding of Speech Signals at Low Bit Rates". Proc. Int. Conf. ASSP, April 1980.
- [56] H.M. Hafez, A.U.H. Sheikh, W. Burchill, W. Sullivan, and S. Boucouris. "Experimental Study of Digital Land Mobile Systems". Department of Systems and Computer Engineering, Carleton University, September 1983.
- [57] N. Dal Degan, F. Perosino, F. Rusina. "Communication by Vocoder on a Mobile Satellite Fading Channel". ICASSP 1985.
- [58] C.K. Un and D.T. Magill. "The Residually-Excited Linear Prediction Vocoder with Transmission Below 9.6 KBPS". IEEE Trans. Commun., vol. 23. pp. 1466-1473. 1975.
- [59] R.J. Sluyter, G.J. Bosscha, H. Schmitz. "A 9.6 KBIT/S Speech Coder for Mobile Radio Applications". ICC 1984.
- [60] R.E. Crochiere, L.R. Rabiner. Multirate Digital Signal Processing. Prentice-Hall, Inc., New Jersey. 1983.
- [61] J.H. McClellan, T.W. Parks, L.R. Rabiner. "FIR Linear Phase Filter Design Program". Programs for Digital Signal Processing. IEEE Press, John Wiley & Sons, Inc., New York. 1979.

END

FIN

6 04 87