INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps.

ProQuest Information and Learning
300 North Zeeb Road, Ann Arbor, MI 48106-1346 USA
800-521-0600

Université d'Ottawa · University of Ottawa

Receiver-Based Packet Loss Concealment for Pulse Code Modulation (PCM G.711) Coders

By

Maha Elsabrouty

B.A. Sc., University of Cairo, 2000

A thesis submitted to the
Faculty of Graduate and Postgraduate Studies
University of Ottawa

In partial fulfillment of the requirements of the degree of
Master of Applied Science
In Electrical Engineering

Ottawa Carleton Institute of Electrical and Computer Engineering
School of Information Technology and Engineering
Faculty of Engineering
University of Ottawa

March 2002

Abstract

Voice-over-IP (VoIP), the transmission of packetized voice over IP networks, is gaining much attention as a possible alternative to conventional Public Switched Telephone Networks (PSTN). However, impairments present on IP networks, namely jitter, delay and channel errors can lead to the loss of packets at the receiving end. This packet loss degrades the speech quality. Model-based coders, especially G.729-A and G.723.1 International Telecommunication Union (ITU-T) Standards, have been extensively used for speech coding over IP networks because of their inherent ability to recover from erasure. Their built-in packet loss concealment makes their quality drop slowly with increasing amount of packet loss. However, their memory makes the transition from the concealed state to the correct state require a few frames and they actually tend to corrupt a few good packets before recovery as a result of a phenomenon known as " State Error". On the other hand, Pulse Code Modulation (PCM), although having a higher score than G.729 and G.723 in the periods of normal operations, does not have the ability to conceal erasure and the quality of speech during loss periods drops dramatically. Yet it can recover from packet loss more rapidly than model-based coders since the first speech sample in the first good packet restores speech to its original quality. The goal of this work is to develop a Packet Loss Concealment (PLC) algorithm to provide the G.711 PCM coders with the required ability to conceal erasure and maintain a high score of user satisfaction. This algorithm uses a receiver-based prediction model to develop an estimate of the missing speech segments.

Table of Contents

Table of Figures

List Of Acronyms

ACC	Advanced Audio Coding
ADPCM	Adaptive Differential Pulse Code Modulation
ANSI	American National Standard Institute
ASIC	Application Specific Integrated Circuit
CAE	Composite Acceptability Estimate
CELP	Code Excited Linear Prediction
CS-ACELP	Conjugate-Structure Algebraic CELP
CNG	Comfort Noise Generator
DCR	Degraded Category Rating
DMOS	Degraded Mean Opinion Score
DoD	US. Department of Defence
DRT	Diagnostic Rhyme Test.
DSL	Digital Subscriber Line
DSP	Digital Signal Processing
ETSI	European Telecommunications Standardization Institute
FEC	Forward Error Correction
GEO	Geostationary Satellite
GSM	Global System for Mobile communication
HEO	High Earth Orbit
ITU	International Telecommunication Union.
LAN	Local Area Network
LEO	Low Earth Orbit
LP	Linear Prediction
LPC	Linear Prediction Coefficients
LSF	Line Spectral Frequencies
MAC	Multiply Accumulation
MIPS	Million Instructions Per Second
MOS	Mean Opinion Score

OLA	Overlap Add
PAM	Pulse Amplitude Modulation
PC	Personal Computer
PCM	Pulse Code Modulation
PESQ	Perceptual Evaluation of Speech Quality
PLC	Packet Loss Concealment
POTS	Plain Old Telephone Service
PSQM	Perceptual Speech Quality Measure
QoS	Quality of Service
RORPP	Reverse Order Replicated Pitch Period
RTP	Real Time Protocol
SONET	Synchronous Optical Network
SNR	Signal to Noise Ratio
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
VAD	Voice Activity Detection
VoDSL	Voice Over Digital Subscriber Line
VoIP	Voice over Internet Protocol
WAN	Wide Area Network
WDM	Wave Division Multiplexing

Dedication

" To mom who taught me how to be what I want to be"

Acknowledgements

I would like to express my deepest gratitude to my supervisors, Dr. Tyseer Aboulnasr and Dr. Martin Bouchard. Their vast knowledge and stimulating discussion made this research one of the most enjoyable experiences in my life. I owe them a lot for their technical feedback and their continuous support.

I would like also to thank Dr. Abbas Yongacoglu. His help and experience were very beneficial to my work on this thesis.

I would like to give special thanks to my mother, Hoda. Her love, encouragement and confidence in me gave me the motivation and the eager to seek the best in my research. I am obliged to her, my brother Khalid and my sister Reem for their continuous support, love and care.

1. Motivation and Thesis Discipline

1.1 Introduction

The transmission of Voice over the Internet Protocol (IP) network has gained a lot of attention in the last few years. A lot of the research work has focused on this active interesting domain and many solutions and proposals have been offered to help the promising multi task network to improve and adapt with the consumers' needs. The IP network used to transmit the voice can either be a public Internet or a private network, usually a Local Area Network (LAN) used by big enterprises. Many configurations are possible for the call setup between users [35]. Costumers can put the VoIP network to work in three ways. The simplest conceptually but least convenient is to use Multimedia PCs equipped with software that is able to perform the real time tasks required for the conversation. Another approach is available through the so-called "voice-enabled" cable modems or Digital Subscriber Lines (DSLs) that enable bypassing the PC and use the regular phones plugged to such boxes. The third approach that is the most convenient but also most expensive is to connect to the network through an Application Specific Integrated Circuit (ASIC) Digital Signal Processing (DSP) chip that is customized to perform the Internet telephony tasks.

Although it originated basically as a cheap alternative to make long distance calls instead of the PSTN telephone network, Voice over IP is much more than that today. A half-century of amazing development and breakthroughs in computer networks supported the IP technology and gave it the chance to extend, develop and turn into a giant connection that covers the globe. The benefits of using VoIP as an alternative or even possibly a replacement of the existing telephony systems are actually more than providing cheaper calls. The motivation of moving to the new domain are based on the promising merging between data and voice networks which enables the use of a unified network for all types of applications. This gives a great opportunity for enterprises to make full use of their network resources and offer individuals the facility of unified access of e-mail, fax and voice.

The benefits can even go beyond this to adding new features, such as video conferencing, to the VoIP service. The flexibility of IP networks makes the application of new speech compression techniques possible and thus CD-quality speech coders that are available today can one day be applied to voice conversation on the network. Although serious problems concerning the algorithmic delay and the complexity may face such applications, the fact that IP networks do not necessitate a certain type of coding standard cannot be ignored.

However, IP networks were originally designed for non-real-time data transfer. This fact made it challenging to adapt to the requirements of real time speech transfer. The operating assumptions of the IP packet network are based on the requirements for non real time data. Packet sizes, packet header overhead and the sizes of queuing and other buffers were chosen to give optimal efficiency of data traffic.

Another challenge comes from the configuration of the access links. These access links are totally dedicated to voice in circuit switched telephone networks. However, in the case of IP network the link can be shared with other voice and data packets. In some cases, this will severely limit the data rate. Internetworking with the traditional circuit switched PSTN network, PBXs and special networks such as wireless also pose quality management issues.

Finally, there are some specific challenges associated with many of the standard voice features, such as hands-free and conferencing features that many conventional telephony customers have got used to and prefer to have in their telephony system.

All of the above are inherent features and properties of IP networks; properties that could disturb real-time voice conversation and reduce the quality of speech. Engineers and system designers have summarized the main problems with configuring the IP network to real-time applications into three main impairments, namely delay, jitter and packet loss.

1.2 Motivation

We have mentioned before that there are two types of IP networks, the public Internet and the private networks. Most of the industrial applications have focused on the second type of networks as the convenient environment for real-time IP application. This has come from the fact that the previously mentioned impairments, namely delay, jitter and packet loss, are well controlled in private networks. Since the IP network is inherently a *best effort* network the transmitted data is not guaranteed to reach the destination and the packets may arrive out of order. However, the use of VoIP over the Internet is still attractive. If the existing impairments can be avoided, the quality of service will be enhanced and the scope of the Internet telephony application will be widened. Generally, the end–to-end quality depends principally on the following factors:

- The speech codec used
- End-to-end delay across the network and variation in the delay
- Packet loss across the channel
- Echo control

Many approaches on the network side have been investigated to adapt the IP configuration to real time applications. This is discussed in more details in Chapter 5. Speech compression, which is an important part of any speech communication system, has a direct impact on the effect of the existing impairments. Given that, packet loss occurs often in an IP network, an appropriate speech compression algorithm to be used in VoIP has to be able to provide good quality during erasure periods and to be able to recover well after the loss.

After thorough experiments on the quality of the voice on different media and considering the data presented on the quality of delivering speech over hybrid systems at tandems, it was clear that the G.711 codec exhibits the best performance among all codec standards. Typical MOS scores show that the G.711 achieves around 4.3 on the 5- point scale while the G.729 scores a number varying between 3.4 - 3.7. These scores are the values in the case of no packet-loss. However, in the erasure periods, much lower scores

are expected. The quality is more severely affected in the case of the PCM coded G.711 speech packets as all the CELP based coders have inherent built-in concealment procedures [46], [55] since the speech is modeled during encoding.

While having these concealment parameters available during the coding / decoding process, the model-based coders cannot recover rapidly from packet loss. They need a few good packets in order to start catching- up again. This is because the memory of the coders still holds their concealed-states even after the interval of lost packet has terminated. This is known as "*The State Error* " [2]. The G.711, on the contrary, can easily recover from the lost packets since the first correct speech samples after erasure restore the speech to the original quality. However, a proper concealment of the packet loss requires more extra computations than in the case of the CELP coders.

Communication systems using model-based coders in speech communication made the decision mainly based on their advantage of less bandwidth utilization that they provide. With the convergence of data and voice and the new technologies that enabled broader bandwidth, user satisfaction and immunity to communication impairments that candidate-coding standards exhibit are more critical factors in selecting the right system.

If a new concealment technique effectively solves the problems of packet loss involved with the transmission of the G.711 it will be a welcome alternative to the existing model-based coders, which are now considered the typical coders for VoIP. The lower complexity of the G.711, compared to other standards, allows for some reasonable added complexity to implement the concealment algorithm. In addition, the absence of the state error problem gives the PCM coders the advantage over the memory dependent model-based codecs. The high quality of the G.711 codec in case of no Packet loss and the fact that its extensive use as a general traditional coder, make it a suitable candidate for all transmission media assuming a powerful, well-tested quality concealment algorithm exists.

1.3 Objective

The objectives of this thesis are the following:

- Providing a review of the different coding categories. This is necessary to properly understand the different properties of each coding technique and thus be able to judge the suitability of each to the real time application intended, namely VoIP.
- Providing a complete analysis of the Linear Prediction (LP) with relevant mathematics and complexity estimation of its implementation. This will help in developing the concealment technique that we propose in this thesis and in evaluating its feasibility.
- Investigating the impairments present in the IP network and presenting the available and proposed solutions to resolve such problems.
- Investigating the performance of different concealment algorithms for the PCM coding technique. A comparative study of their performance with the proposed new concealment algorithm is targeted as a verification of the performance of the new algorithm.

The main contribution of the thesis is presenting a new concealment algorithm to be added to the PCM decoder (receiver-based). The algorithm will be shown to provide excellent performance in the periods of loss with the limited complexity.

1.4 Thesis Layout

The seven remaining chapters will be divided as follows:

Chapter 2: Understanding Voice

In this chapter we will review the nature of speech and how it is produced. This will be followed with the presentation of different categories of speech and defining the most

important speech parameters, such as the formants and the pitch. We then describe the different coding categories with examples of the most popular standards. A comparison between different coding techniques is performed to reveal suitability of each coding technique to the desired application, namely VoIP.

Chapter 3: Linear Prediction of Speech

In this chapter, we consider the Linear Prediction process, the LP coefficients and different methods used to extract them. Mathematical explanations and complexity estimation are then performed for the Levinson-Durbin algorithm, used to extract these parameters. The autocorrelation calculation is then presented. A complete explanation of the time- domain and frequency-domain approaches used in the autocorrelation function estimation is presented with complexity estimation and optimized complexity estimation for the case of real speech sequences. At the end, pitch detection is presented with different approaches used in its calculation. This chapter actually contains the main mathematical background we relied on in the rest of the thesis.

Chapter 4: Voice Perception and Assessment Tools

In this chapter we present a review of the physiological and psychological aspects concerned with the process of voice production and perception. Several subjective and objective assessment tools are presented with comparative analysis of their principles of work. A new objective assessment category simulating the subjective human assessment quality is reviewed. A new tool in this category, the ITU-T P.862 standard, is investigated in more detail later in this chapter. This is the tool we used in the thesis to assess the quality of the concealed packets for the different concealment techniques investigated.

Chapter 5: Transmission Impairments and Concealment Algorithm

In this chapter we present the different impairments in the IP network, namely, delay, jitter and packet loss. A study of their origins, the typical allowed values and the effect of these impairments on the quality of speech are performed. A review of some currently proposed solutions aiming at reducing or eliminating their effects is presented. We focus

on packet loss concealment algorithms, especially the two standards of packet loss concealment for the PCM G.711 standard, namely, the ITU-T (G711 Annex A) and the ANSI (T1-521-2000). These two standards are presented in more detail later in this chapter.

Chapter 6: Design History

We chose to title this chapter "Design history" to emphasize that all the design phases presented in it are not yet the final form of our proposed algorithm. It provides a review of most of the ideas we implemented for troubleshooting the difficulties and problems we faced. The chapter starts with explaining the basic principle of work we rely on in the design of the Packet Loss Concealment (PLC) algorithm then followed with an explanation of the different phases we have passed until the development of its final form. This is presented with the corresponding results for each of these phases. Results are presented as appendix A and are compared to typical values obtained from other concealment algorithms under the same loss patterns.

Chapter 7: New Packet Loss Concealment Algorithm

In this chapter, we propose the final form of the concealment algorithm as the step following the last phase presented in Chapter6. A description of how the new algorithm operates in periods of normal conditions (no-loss) and in erasure periods is presented along with a discussion of the delay consideration and complexity. We then present the test setup: the test files, test tool and the different test patterns used. The performance of the proposed concealment algorithm is compared to the performance of the ITU-T G.711 Annex A PCM concealment standard. Modified loss patterns are then compared to both the ITU-T standard and the ANSI Annex B standard. The new concealment algorithm has proved to be superior to both standards in almost all cases.

Chapter 8: Conclusion and Future Work

This chapter concludes the thesis by discussing the main results and contributions as well as possible future work that may enhance the results presented here.

2 Understanding Voice

2.1 Overview

In order to select the appropriate technique to conceal errors in a given speech transmission system over IP networks, the main characteristics associated with both the nature of speech and its production process should be first understood. The concealment algorithm will also clearly depend on the specific coding/transmission system. We thus review three distinct coding categories commonly used highlighting their advantages and disadvantages.

2.2 Basic characteristics of human voice

Human speech can be identified by either looking at it from its physical nature or by considering the physiological process responsible for its production.

If considering it as a physical phenomenon, speech is principally defined as "a sound produced by humans in order to communicate" [2]. As any sound wave it is a transverse wave that causes perturbation in the atmospheric pressure and can propagate through different media, usually through air, and is perceptible when received by the ear.

On the other hand, when investigating its production process it is known that the acoustic speech signal originates in the vocal tract. The vocal tract is the combination of human organs that contribute to the speech production. Speech is then identified as a pressure acoustic signal that is articulated in the vocal tract. This articulation depends on the nature of the organ and its location.

2.2.1 Speech Production Process

Voice is produced when air is forced from lungs through our vocal cords and along the vocal tract. The vocal cords are two pairs of mucous membrane that project into the cavity of the larynx, while vocal tract extends from the openings of the vocal cords, referred to as glottis, to the mouth.

This air- flow is referred to as the *excitation signal* of the production of speech [5]. This excitation signal causes the vocal cords to vibrate and propagate the energy to excite the oral and nasal openings, which play a major role in shaping the spectrum of the sound produced. Figure 2.1 explains the main components in the vocal tract path. As seen in the figure, the human vocal tract consists mainly of two parts: oral tract (from lips to vocal cords) and nasal tract (from the volume of nostrils). These two parts play the main role in shaping the voice.

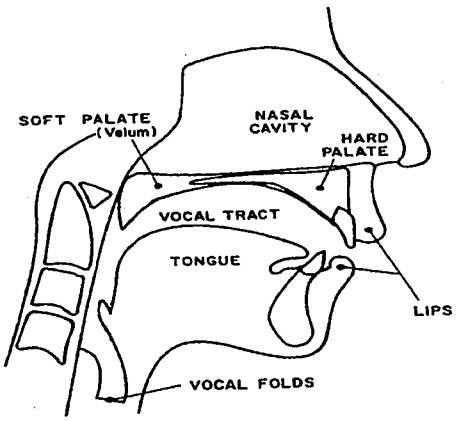


Figure 2.1: The human vocal tract [32]

2.2.2 Acoustic Characteristics of Speech

2.2.2.1 Formants

Formants are best described as high-energy peaks in the frequency spectrum of speech sound. Formants can be noticed in the spectrogram of speech segments exhibiting semi-periodic nature (referred to as "*voiced*" speech, which will follow in the later explanation). Three main formants can be easily recognized in the frequency analysis of any human speech. Their location and relative amplitudes are unique properties of each human voice and many voice/speaker recognition systems depend on specifying their location as their principle of operation. [1].

2.2.2.2 Pitch

The frequency of the periodic (or more precisely quasi-periodic) excitation is termed the "*pitch*". It describes the time span between the opening and closing cycle of the vocal cords [1].

2.3 Classes of speech

A very coarse, yet widely used, classification categorizes speech segments into one of three classes based on the mode of excitation. A majority of normal speech sounds can follow this classification. Those three classes of speech include "*voiced*" sounds, "*unvoiced*" sounds, and "*plosive*" sounds.

2.3.1 Voiced sounds

Voiced sounds are produced when vocal cords vibrate as a result of the lungs generating sufficient pressure to open the vocal folds. Vocal cords vibration caused by this airflow has certain frequency that is dependent on the length of the folds and their tension. For most people the range is between 50 to 400 Hz and is referred to as the "*pitch*

frequency" component of speech. Voiced sounds have a high degree of recurrence at regular intervals throughout the pitch period [1].

2.3.2 Unvoiced sounds

Unvoiced sounds occur when the vocal folds are open allowing the air to pass freely from the lungs to the rest of vocal tract. It is characterized by a relatively flat frequency spectrum. "S", "F", "SH" are examples of unvoiced sounds.

2.3.3 Plosive sounds

Plosive sound is third category of sound covering sounds produced by turbulent airflow suddenly released from lungs into the previously completely closed vocal tract. "P", "B" letters are good examples of such sounds.

Figure 2.2 presents an example of voiced speech

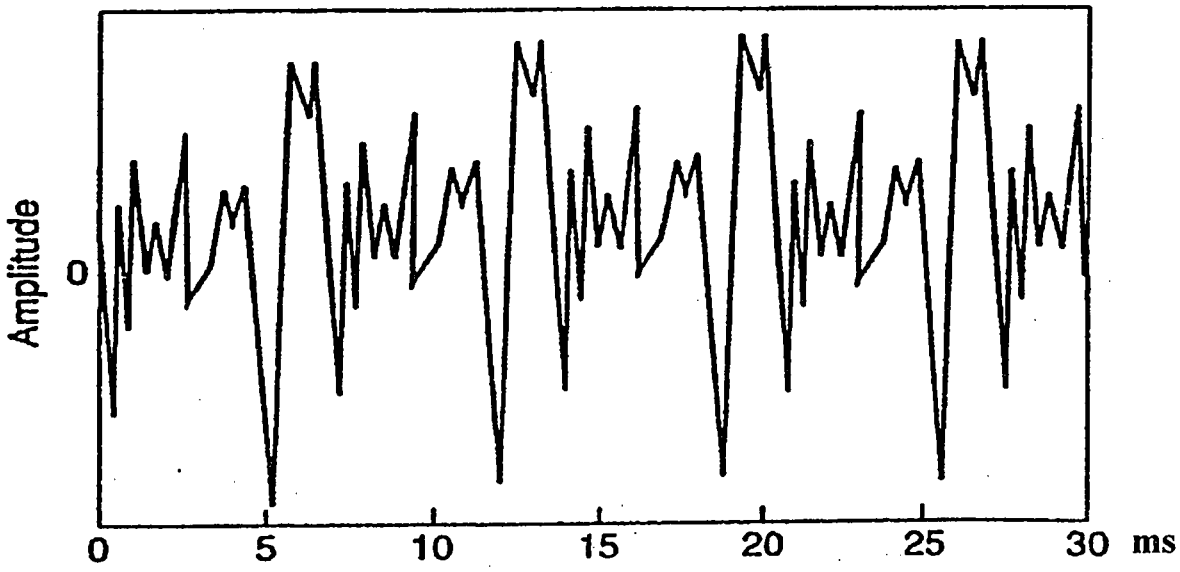


Figure 2.2 Voiced speech segment

2.4 Speech coders classification

Detailed examination of human speech led to three important observations that were used in three different classes of speech coding each emerging from one of these observations.

First, it was noticed that if speech is sampled at appropriate intervals, digitization would have hardly any effect on the reconstructed signal making it indistinguishable from the original speech. The only requirement is to have enough samples to carry the necessary information about the original waveform. This type of coding is referred to as "*waveform*" coding.

A second aspect of speech is that speech parameters that can be derived by analyzing the speech production process do not vary, in terms of milliseconds. This means that speech is quasi-stationary over a certain period of time and hence is sort of predictable i.e. can be synthesized. This second observation led to techniques that depend on modeling the speech production process. This class of techniques is referred to as "*vocoding*".

The third observation was that part of the speech is not audible so coding can be focused just on the audible or perceptible part. This led to a new approach, "*perceptual-based*" compression, which depends on coding only the audible part and hiding or masking the inaudible part through applying a certain window called the masking threshold, that has been calculated on experimental results in order to simulate the natural ear response to different frequencies.

2.4.1 Wave form Coding

Waveform coding can be simply defined as a process whereby an analog signal is digitized without requiring the knowledge of the physiological phases of the signal production. It can be viewed as a general method that depends only on the final form of the voice signal and can be applied, with minor modification, to other signal types. It is so considered the simplest family of speech compression algorithms.

Under this family, many practical systems can be categorized. Those include: Pulse Code Modulation (PCM) [11], Differential Pulse Code Modulation (DPCM) and Adaptive Differential Pulse Code Modulation (ADPCM) [22], [24].

We have chosen to present the PCM for three main reasons: First, PCM is the universal method of voice digitization used by communication carriers for transmission in Public Switched Telephone Networks (PSTN) and is proposed to be the standard of the IP switched Networks. Second, other systems of this family are based on this system and are considered more complex improved methods. Third, linear PCM signal is the input to the model based CELP algorithms like the G.729 and G.723.1 recommendations. It is also the input to the concealment algorithm that will be presented later in the thesis.

2.4.1.1 Pulse Code modulation

Pulse Code Modulation is a waveform coding technique that is accomplished through three ordered steps:

(1) Sampling: analog signal is band limited to 4 KHz then sampled 8000 times per second or once every 125 μs. The choice of the sampling rate is based on the Nyquist theorem [13]. 4 kHz was chosen as an upper limit to the frequency content of the speech signal based on the nature of human voice where most of the power is concentrated in the range between 300 Hz – 3300 Hz. The output of the sampling process is referred to as a Pulse Amplitude Modulated wave (PAM) [13].

(2) Quantization: Due to the fact that we cannot store values of the signal with infinite precision, a process has to be applied on the signal to limit the amplitude to discrete values.

(3) Coding: The amplitude of the signal of different values has to be transformed into binary values to be sent as bits to the receiver destination to faithfully reproduce the signal.

An eight-bit presentation would provide 2^8 or 256 possible values and would require a transmission rate of 8000 samples per second X 8 bits/sample or 64 Kbps. PCM was standardized by International Telecommunication Union (ITU) as Recommendation G.711 [11] and represents by far the systems that are most commonly used: μ-law in North- American system and A- law which is used in the rest of the world [13].

Since communication carriers have already invested billions of dollars over years in PCM-based communication systems, it is logical that when looking at the relatively new domain of IP great efforts are performed to make the existing coding system cope with the new communication domain.

2.4.2 Model based Compression

As previously discussed, the model-based compression depends on the characteristics of human speech and tries to synthesize the voice through modeling the vocal tract. Its objective is to simulate the speech production process at the transmitter to extract speech parameters and send them to the receiver. The receiver reproduces the speech from those parameters. These parameters are sent instead of voice segment itself, in order to perform more compaction in the bandwidth used. These parameters play the main role in shaping the excitation signal by acting as the coefficients (poles) of the vocal tract. The vocal tract is represented as an all-pole adaptive filter whose poles are updated according to a predetermined period that varies slightly according to the coding standard used. This system is given the term *"vocoder"*, which is the acronym for voice coder.

2.4.2.1 Operation of Vocoder

Vocoding is based on the assumption that the vocal tract is a linear system excited by either a series of periodic pulses, with a fundamental period equal to the pitch value, for voiced speech, or random noise in case of unvoiced speech (including plosive). The following figure shows the main components of a vocoder synthesis filter.

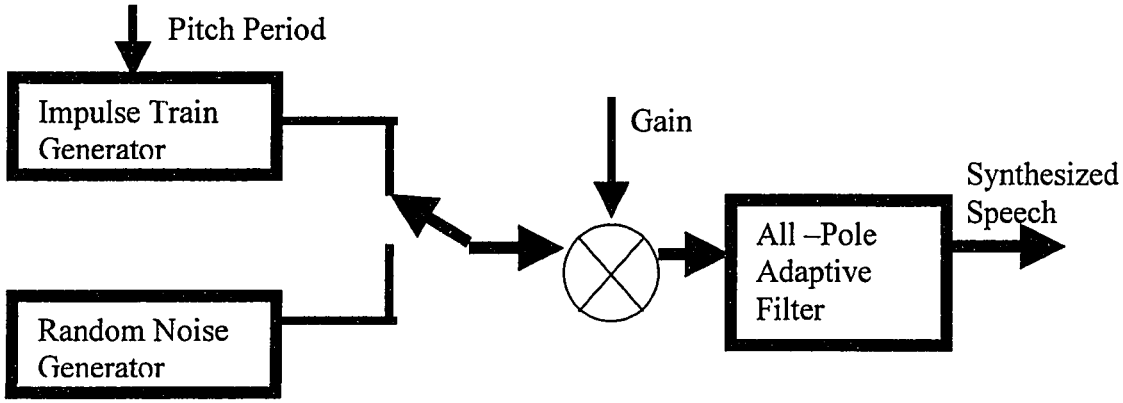


Figure 2.3: The Vocoder Synthesis Part

Code Excited Linear Predictor (CELP) is an improved version of the basic vocoder. It depends on a slight change in the excitation signal, which was previously assumed to be either pure voiced or pure unvoiced. This assumption does not match with certain letters and the beginning and ending of words in the middle of sentences [2]. The CELP model, in contrary, assumes that the excitation signal is a weighted mixture of both the impulse train (for the voiced part of the excitation) and the unvoiced excitation represented by random noise. The variable weighting allows for faithfully representation of different voice classes, varying from purely unvoiced to voiced segments. Also, extra information such as the positions of the pulses or the shape of the noise to be generated can be transmitted to improve the quality of the synthesized speech. The following figure shows the main block diagrams in a conventional general CELP synthesizer.

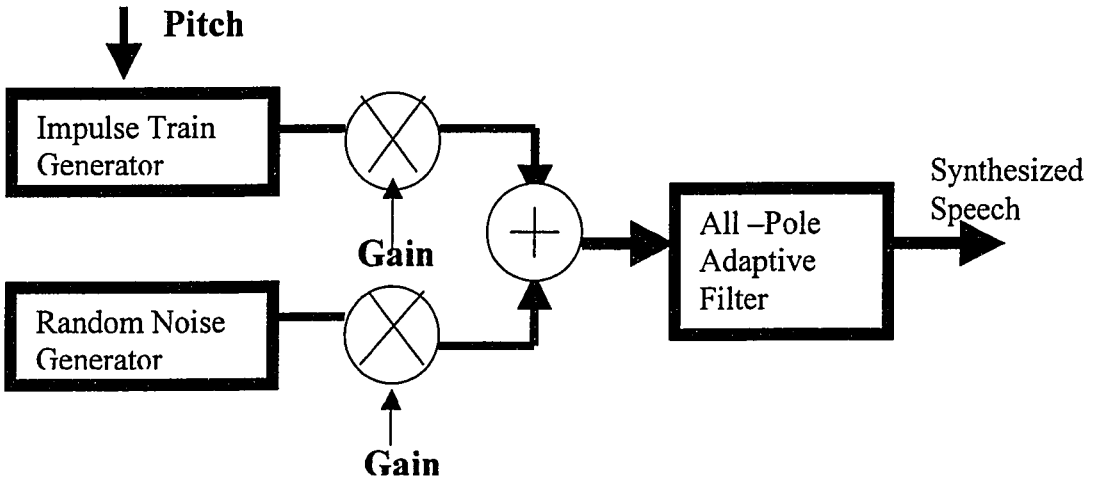


Figure 2.4: The Main Blocks of A General CELP-Based Speech Synthesizer

Parametric models depend on the assumption that voice is quasi stationary between 10 to 25 ms. The following parameters are then determined over a time frame of 10-25 ms.

- Coefficients of the all-pole filter which in fact are the resonants or formants of the vocal tract.
- A binary parameter to determine whether the voiced segment is voiced or unvoiced (case of normal vocoder) or two values for the gains of the voiced and the unvoiced blocks in the model (CELP model).
- Pitch value, which is used as the period for voiced speech excitation.

As mentioned earlier, extra information as the excitation (pulse shape, etc.) can also be transmitted to improve the coding quality.

2.4.2.2 Standard systems for the CELP coders

ITU-T, European Telecommunication Standardization Institute (ETSI) and US Department of Defence (DoD) have standardized many CELP model based systems. The table below shows the most used ones ordered in ascending order with respect to their bit rate [20], [35].

Standard	Bit rate	Algorithmic Delay	Year Finalized
FS-1015 LPC-10E	2.4 Kbps	111.5 ms	1984
FS-1016 CELP	4.8 Kbps	37.5 ms	1991
ITU-T G723.1 MPC-MLQ	5.3 Kbps or 6.4 Kbps	37.5 ms	1995
ITU-T G.729 CS-ACELP	8 Kbps	15 ms	1995
ITU-T G729 A CS-ACELP	8 Kbps	15 ms	1995
GSM RPE-LTD	13 Kbps	20 ms	1987
ITU-T G.728 LD-ACELP	16 Kbps	0.625 ms	1994

Table 2.1: International Standards for Model-based Speech compression

As we can see from Table 2.1, the main advantage of these systems is that they send only the speech parameters instead of the sampled signal thus saving in the bandwidth.

From the above table and the well known requirements of real-time IP application, compromising between the low bandwidth (which is considered an advantage of the CELP coders) and the algorithmic delay (which is the main disadvantage especially for real time applications that are to go through several coding/decoding intermediate repeaters where the algorithmic delay adds up at each intermediate node) we can see that some systems are not used despite having a very low bit rate due to the large algorithmic delay that they require at both the transmitter and receiver. Only three systems are widely used in the IP speech communication world namely, the G.723.1, the G.729 and the G.729.A [35]. They depend on analysis by synthesis: the coder has an embedded decoder in it that tries different values of random noise to get the best match with the natural unvoiced excitation of the speaker's speech.

2.4.2.3 ITU-G729.A, CS-ACELP

ITU introduced this standard in November 1995 as a part of sub-CELP model family: Conjugate Structured-Algebraic Code Excited Linear Prediction (CS-ACELP) method of speech signal compression at 8Kbps [8].

The G.729 Recommendation and a reduced complexity alternative specified in Annex A of that standard represent two voice coding methods that produce a high quality and high compression ratio voice coder that results in a low data rate for the transmission of the digitized speech.

CS-ACELP uses 10 ms frame size plus 5 ms look ahead, which results in a total of 15ms algorithmic delay. It uses a data rate of 8 Kbps to encode a voice conversation. The G.729 Annex A [10] CS-ACELP speech compression algorithm represents a reduced-complexity version of the G.729 coder. It has been developed for the use in multimedia applications that requires simultaneous voice and data transmission capability.
It is worth noting that even the high compression ratio of the G.729-A has been improved by some vendors who added a proprietary silence-suppression capability to the standard that reduces the average amount of bandwidth required to transport a voice conversation to approximately 4Kbps[18]. Such as silence suppression scheme was later standardized by the ITU-T in Annex B supplementary to the original standard.

2.4.2.4 The G.723.1 Recommendation

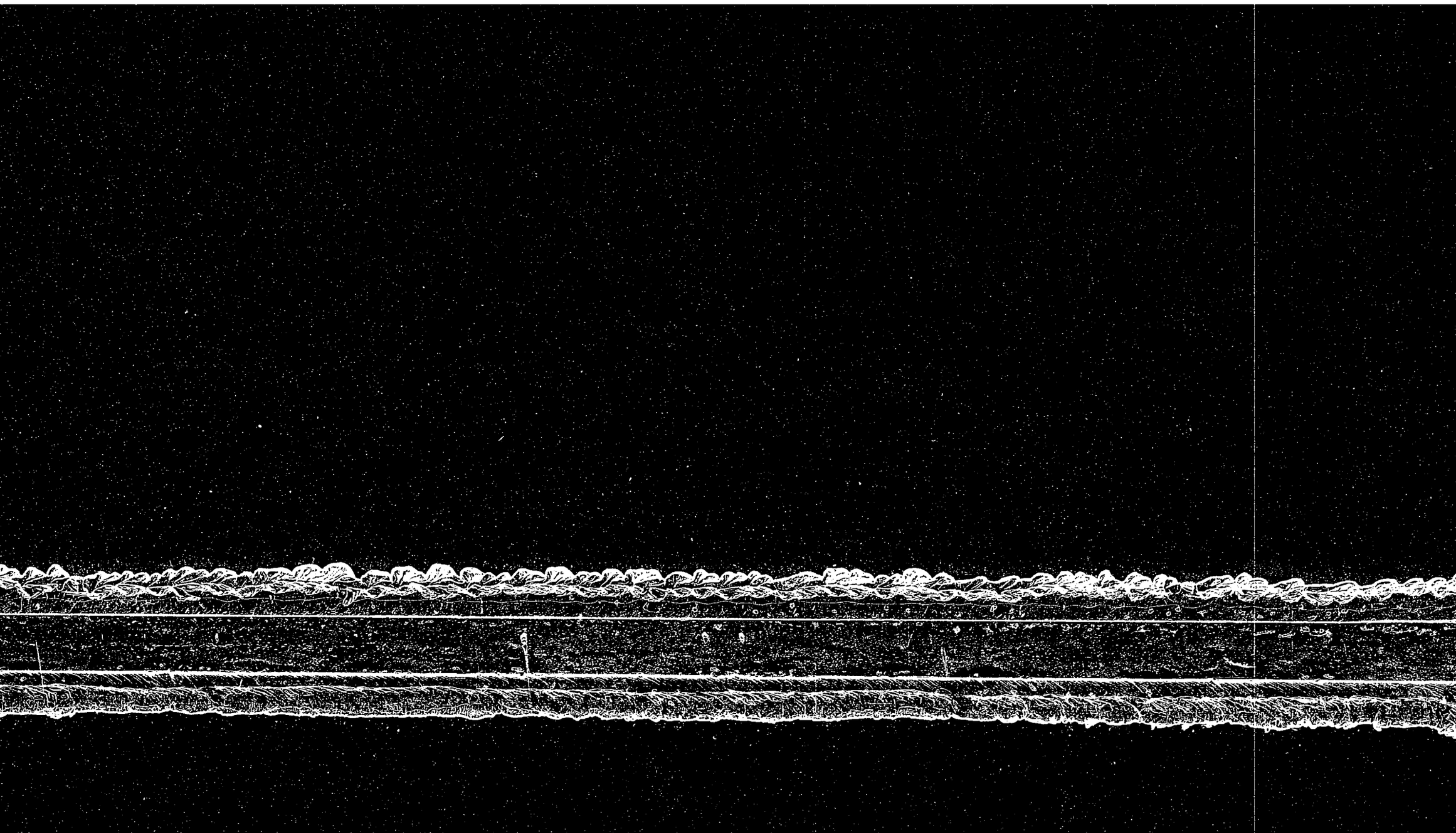
Encoded voice packets can be lost during transmission. This probability increases when transmitting over frame relay or IP networks. The detection of an error could result in the retransmission of the whole packet, thus introducing controlled delay to be added randomly to the algorithmic delay. This will also limit the ability of the receiver to reconstruct the speech in correct order due to finite buffer size problems. This delay is critical to the real time applications, in the CELP-based coders are used extensively.

For the above detailed reasons, frame erasure ability became an important consideration when the ITU examined several methods to standardize the audio portion of video conferencing. As a result the ITU-T selected a dual-rate code that is known as the G.723.1. The G.723.1 coder supports two rates: 5.3 and 6.3 Kbps with the higher bit rate providing better quality of the reproduced voice while the lower bit rate is optional for high traffic period to combat the problem of congestion in the traffic medium. There exists an optional part that supports the periods of silence using a Voice Activity Detection (VAD) technique which is similar in operation to the one commercially added to the G.729.A [18] and results in a variable lower operating rate that has an average between 2.65 and 3.15 Kbps.

The G.723.1 Recommendation was approved by the ITU-T in March 1996 and recommended by the International Multimedia Teleconferencing Consortium's forum to be the default low bit rate audio coder for the ITU-T H.323 standard. The ITU-T H.323 standard [30] defines the method for voice and video communication over packet –based networks that makes it one of the most favoured coding techniques for Internet telephony and voice conferencing. The G.723.1 Annex A [19] standardizes the previously mentioned silence suppression VAD techniques. The following table [35], [15] summarizes the main properties of the G.723.1, G.729 and its simplified Annex A.

Codec	**G.723.1**	**G.729**	**G.729A**
Bit Rate	5.3/6.4 kbit/s	8 kbit/s	8 kbit/s
Frame Size	30 ms	10 ms	10 ms
Processing Delay	30 ms	10 ms	10 ms
Look ahead Delay	7.5 ms	5 ms	5 ms
Frame Length	20/ 24 bytes	10 bytes	10 bytes
DSP MIPS	16	20	10.5
RAM	2200	3000	2000

Table 2.2 [35]: G.729, G.729A and G.723 Codecs Performance Parameters



2.4.3 Perceptual-based Compression

This compression category depends, as we mentioned before, on the fact that human ear can only detect a portion of the frequency range contained in the voice that is above the minimum threshold of audibility of the ear. Some other sounds are masked by a louder sound of neighbouring frequencies. This fact led to the idea behind perceptual-based speech compression: we do not need to code the part of speech that is not audible.

The principle of operation of such compression coders is to let the encoder distinguish between the audible and the non- audible part as a first step. The second step is the encoding of only the audible part and discarding the non-audible one. In the following figure we can see a typical perceptual compression algorithm.

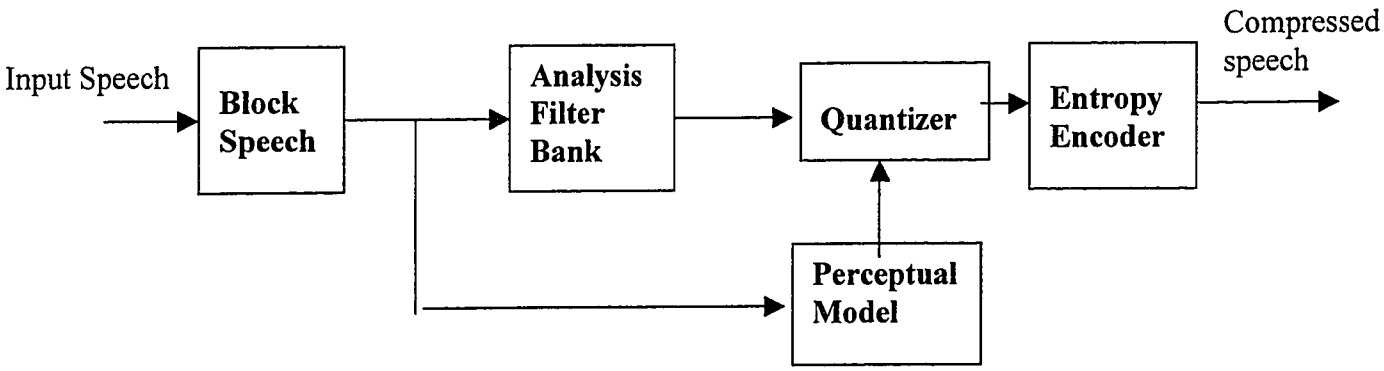


Figure 2.5: Block Diagram of a Typical Perceptual Encoder

The figure shows the block diagram of a typical encoder for the perceptual-model speech compression coder. The input to such coder is usually an 8 bits per sample digitized linear- PCM voice. This digital voice is divided in blocks between 256 samples to 2048 samples depending on the implementation. This is done in the "*Block speech*" step that introduces the largest part in the delay found in this coding category. Then, the output of the blocking stage is passed through two blocks simultaneously. The first one, the "*Analysis filter bank*", has the task of decomposing the speech into frequency bands that have same perceptual weight, [7]. These bands are referred to as "*critical bands*". The perceptual model determines the audibility of the different frequency bands in the speech.

The quantizer block allocates as few bits as possible to the less audible speech components, again depending on the perceptual model that will be explained later. Finally, the entropy block is optionally added to the encoder in order to perform extra bit compression. In this block more bits are allocated to speech symbols with lower probability of occurrence of different speech symbols; fewer bits are allocated to the more probable symbols.

We can see in Figure 2.5 that we have three blocks with the main task of performing bit compression. This leads to a logic expectation that this compression algorithm has a very low bit rate though it is more complex as it depends on many models and technical operations to be performed.

2.4.3.1 Perceptual Model

The perceptual model as we saw in the previous section is the central block of the encoder of the perceptual speech coder. It compresses the signal and lowers the number of symbols that are to be transmitted to the decoder without affecting the quality of speech. The degree of compression depends on the minimum audible masking threshold that determines the audibility for all the tones that are currently perceived by the ear. This masking threshold is mainly the same for all humans with slight variations that introduce hardly any difference in the judgment of the quality of the reconstructed speech. The masking threshold has two essential components that contribute to its formation. The first component is threshold in quiet. The second one is the summation of two sub-factors that are: temporal masking and simultaneous masking (frequency masking). These factors are to be discussed in brief in the next three sub-sections. Summing the effects of the threshold in quiet and the simultaneous masking forms the masking threshold for a speech block, which determines the degree of audibility of speech. Temporal masking is taken into account by performing a past addition process which adds the current value of the temporal masking to the past value of the already formed masking model.

2.4.3.2 Threshold in quiet

The first factor in forming the masking threshold model is the *"Threshold in quiet"* which is a measure on the sensitivity of the human ear perception of different frequencies. This model has been formed through intensive psycho-acoustic testing [7].

Through curve fitting techniques, the model is found to follow the following formula that depends only on frequency [2]:

$$ATH(f) = 3.46(f/1000)^{-0.8} - 6.5\, e^{-0.6(f/1000-3.3)2} + 10^{-3}(f/1000)^4 \qquad (2.1)$$

where f is the frequency in Hz.

Figure 2.6 shows the audible threshold in quiet versus the frequencies. As we can see in the figure the low value for threshold which indicates an audible frequency with low magnitude is concentrated at frequencies 200-10000Hz or in an approximated range between 500-8000Hz while frequencies that are outside this range are hard to hear and The further the frequency from this range, the more power it should contain to be heard. Thus noise is inaudible provided that it is below this hearing threshold.

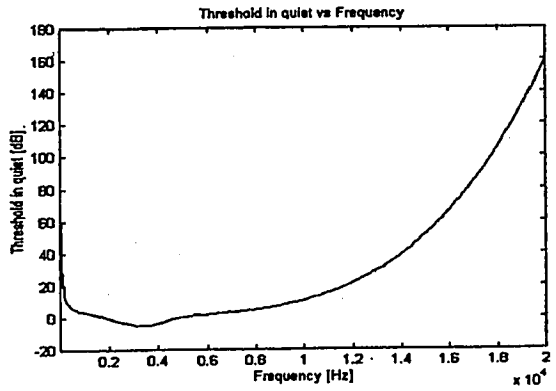


Figure 2.6: The threshold in Quiet [2]

2.4.3.3 Simultaneous masking

Simultaneous masking is simply the effect of a louder frequency component on relatively weaker neighbouring frequencies turning them inaudible even if they were above the threshold in quiet value.

In essence, the hearing threshold of Figure 2.6 is modified (upwards) in the presence of a given signal. This phenomenon is better observed through forming a Fourier transform on the speech block and then examining the masking that each frequency causes to the adjacent ones. For simplicity, it is generally assumed that simultaneous masking of several tones is the sum of the simultaneous masking performed by each tone. Figure 2.7 shows that the individual frequency component affects the audibility of the surrounding ones more than the more distant ones, which fall in another *"bark"*. Bark is a one critical band as defined in [7]. Another factor that determines the impact of the masking is the tonality of the masking frequency. Noise is known to cause more masking than normal tone [7].

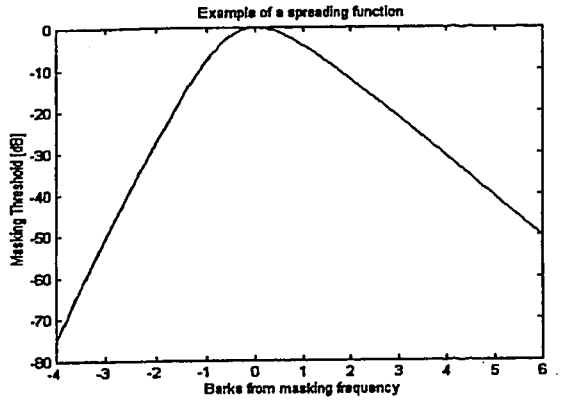


Figure 2.7: Spreading function [2]

2.4.3.4 Temporal Masking

Temporal masking is a measure of the audibility of frequency components in the present given a strong neighbouring tone that occurred within 200ms prior to the present time of analysis. This comes as a result of processing signal in block form [2]. This phenomenon has the same factors affecting it as the simultaneous masking except that higher power is necessary for the masking frequency to perform that effect in the future periods.

2.4.3.5 Practical perceptual-based systems

Practical coding is used commercially for audio compression to allow high quality at a very low bit rate. Also several international standards have been developed for perceptual

compression, for example: MPEG-1/ Audio that supports sampling rates of 32, 44.1 and 48 kHz and bit rates between 32 kb/s (mono) and 448 kb/s, 384 kb/s and 320 kb/s (stereo and layers I, II and III respectively), MPEG2: advanced audio coding (ACC) [26], which also supports multiple bit rates and multiple sampling rates. Most of the perceptual-based systems that are in the market now permit CD quality speech with a relatively low bit rate that is (usually below 64 kb/s). More details on the practical perceptual systems can be found in [23].

2.5 Comparison between different coding techniques

The following table presents a point form comparison among the three main families of coding. The characteristics investigated in the comparison are the major factors that decide the use of a certain coder in any type of networks. These characteristics are even more important when the choice is made for such a demanding network like the best effort IP network running real time applications. The restrictions in this type of networks become more demanding while the industry implementation requires less complexity.

Characteristic	**Waveform-based**	**Model-based**	**Perceptual-based**
Delay	Very low	low	high
Speech Quality	Toll quality	Less than toll quality	Toll quality or better
Bandwidth	High (PCM) or medium (ADPCM)	low	medium
Complexity	low	Medium/high	Medium/ high
Frame Erasure Ability	Fair (needs concealment tools)	Very good (built in concealment)	Fair
Other	No state-error problem	Problems of tandem connections	Permits to use CD quality of speech

Table 2.3: Comparison Among Different Families of Codecs.

From the above table we can see that although the perceptual based coders offer higher quality speech than the model based coders they are not widely considered as an option for a standard coder for IP networks due to the large delay associated with this coding technique. In fact, the model-based family has been under focus for the IP real time applications due to their low bandwidth and satisfactory frame erasure ability. The fact that this system depends on extracting the parameters of speech makes generating concealed speech segments to replace the original ones in the periods of speech loss relatively easy using the older parameters and the already built speech production model. However, it was discovered that due to the dependency on the previous speech parameters to produce the current segment the erasure period does not affect only the current lost speech segments but the effect extends to the next correctly received frames in a phenomenon known as " The state error" or the " Memory Effect" [2]. This is discussed in more details later in the thesis.

The waveform- based coders, particularly the PCM have gained privilege through years of worldwide usage in telecommunication world. They have the best known performance for tandems and exchanges multi coding/decoding chains [55]. They however require more bandwidth and do not exhibit good performance in the periods of speech loss due to the absence of concealment block (despite not suffering from the state error problem). That is why there is a definite need for a high quality low cost concealment algorithm for PCM codecs. If such an algorithm is found, it will make the PCM family a very strong candidate for the standard for speech transmission over IP networks that is until today does not have a unified standard codec [43].

2.6 Summary

In this chapter, the main characteristics of speech signals were presented. We categorized speech to be voiced, unvoiced or plosive sounds. The general properties of each of these categories were presented. Having obtained the basic knowledge about the speech physical and production nature, we then presented the different speech coding families. Each of these families depends on a unique property of the speech signal as principle of work. The standard for conventional telephony and a member of the waveform-based compression family, Pulse Coding Modulation (PCM), was introduced in more details since it will be used later in this thesis. It has been shown that PCM has the advantage of simplicity and good speech quality but requires a relatively higher bandwidth. Then we proceeded to the second family, namely the model-based family. The principles of work of both the vocoding-based coders and the Code excited Linear Prediction Coder (CELP) were introduced. Two of the most widely used standards of the CELP-based family were presented, the G.729 Code Excited Linear Prediction (CS-ACELP) and the G.723.1, both ITU-T standards. The third and newest family is the perceptual-based speech compression algorithm. A review of a typical perceptual model was introduced in order to understand how the algorithm can accurately distinguish and extract the frequency components that are not audible by the human ear and then encode only the audible part. Finally, the advantages and the disadvantages of using each family were presented along with the suitability of each category for real-time applications. We concluded that for IP telephony the choice of the speech coder should be limited to either the model-based coders (which are widely used in this type of applications) or the PCM (which has many advantages but lacks the proper erasure concealment ability). The perceptual based coders were excluded as they introduce large delay. This delay is intolerable in real time conversations.

3. Linear Prediction of Speech

3.1 Overview

In this chapter we present the linear prediction process since it provides the basis for the packet loss concealment algorithm we propose. Understanding the linear prediction process is also helpful in estimating the approximate complexity associated with the proposed erasure system and helps to provide better idea about the efficiency of such a concealment system in commercial use.

In fact, the Linear Prediction Coding (LPC) is more than a powerful popular coding technique. It is a complete mechanism of analyzing and restoring the speech signal to its primary form (simulating the speech production process). It is for this reason that many LPC-based coders exhibit excellent immunity to erasure due to their inherent concealment ability. However, the dependency of the present sample computation on the previous ones led to the problem of "State-Error", which has previously been addressed in Chapter 2. This problem limits the scope of this excellent performance in the erasure periods due to the fact that the loss extends to the next good samples after erasure.

When we started thinking of an efficient concealment algorithm for the G.711 PCM coding standard, which unfortunately lacks the ability to conceal lost packets, attention was focused on the model-based techniques that have proved good erasure performance. This is why a full understanding of the prediction process is necessary, as it will help a lot in implementing a complexity-reduced version of the prediction system that should be designed to give the PCM coder the same ability to recover the lost packets, with an added complexity as low as possible.

3.2 Linear Prediction Model

In Chapter 2 it has been shown that the vocal tract can be modeled by an *autoregressive (AR) model* or what is known as an All-Pole model. In such model the nasal cavity effect is neglected and so the voice production system includes only the pharynx and the mouth cavity. The AR model is best described through the following equation:

$$S(n) = \hat{S}(n) + r(n) \qquad (3.1)$$

$S(n)$ corresponds to the speech output sample, $\hat{S}(n)$ is the estimate of the current speech sample through the linear prediction equation:

$$\hat{S}(n) = \sum_{k=1}^{p} a_k S(n-k) \qquad (3.2)$$

Equation (3.2) establishes the recursive connection between the current output sample and the past samples. The set of parameters a_k are optimized in order to minimize the power of the difference between the estimated signal $\hat{S}(n)$ and the original signal $S(n)$. This difference is called the residual signal and is referred to as $r(n)$ in Equation (3.1).

This residual signal is also given the name " *the prediction error*" and is given as:

$$r(n) = S(n) - \hat{S}(n) \qquad (3.3)$$

$$= S(n) - \sum_{k=1}^{p} a_k S(n-k) \qquad (3.4)$$

These two identical equations are the main equations used to calculate the residual signal at the coder part. They actually form the basic principle in the analysis-by-synthesis part in the model-based codecs. The residual signal is extracted from the subject speech frame. The encoder either sends this residual signal or sends a pointer to some similar values, usually kept in memory, to the receiver. The decoder part uses this residual signal

to reconstruct $S(n)$, synthesized by modelling the speech by an all pole filter whose coefficients a'_k are close to the set of coefficients a_k optimized at the coder. The following figure describes Equation (3.4)

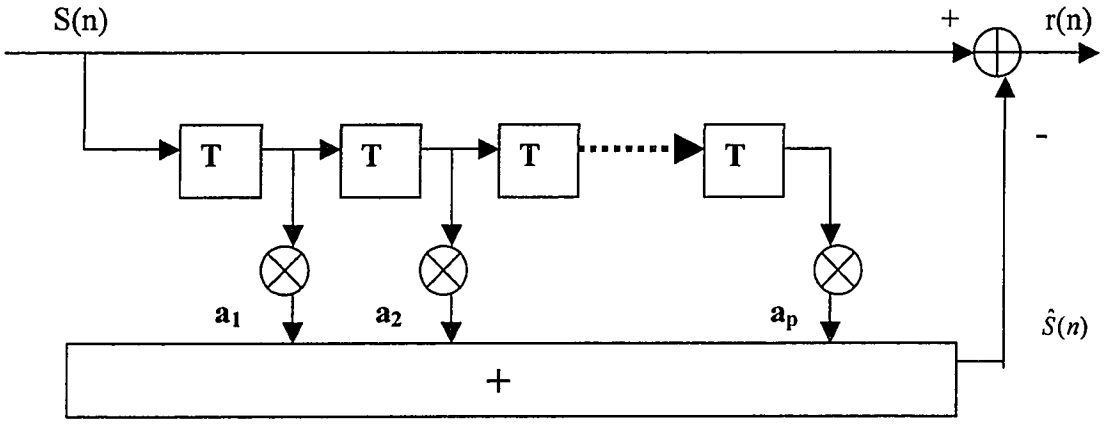


Figure 3.1: Linear Prediction Process

These equations are more easily implemented in the Z-domain

$$R(z) = (1 - \sum_{k=1}^{p} a_k z^{-k}) S(z) \qquad (3.5)$$

$$= A(z)\, S(z) \qquad (3.6)$$

Here, R(z) and S(z) are the z-transforms of the residual and the speech signals respectively and A(z) is the z-transform of what is known as the Linear Prediction (LP) analysis filter. This filter is used to extract the residual signal from the original speech signal. The coefficients set a_k plays the main role in the formation of this residual signal which is aimed to be as small as possible through the previously mentioned optimization of the prediction coefficient parameters. This optimization process will be addressed in more details when the prediction coefficients formation is described later in this chapter.

The above mentioned A (z) filter is the analysis filter located at the coder side. There exists an adjacent filter on the receiver side that is responsible for reconstructing the current speech sample from the corresponding residual signal and the past samples in a process known as the Linear Prediction synthesis. This filter, referred to as H(z), is called the LP- synthesis filter or the All- pole synthesis filter. It is given as [5]:

$$H(z) = \frac{1}{A(z)} \qquad (3.7)$$

In the following figure a block diagram of the Linear Prediction analysis and synthesis operations is shown:

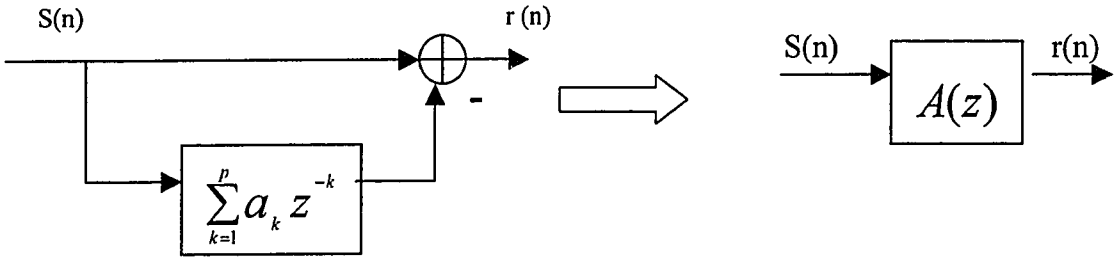


Figure 3.2 (a) The Linear Prediction Analysis filter [5]

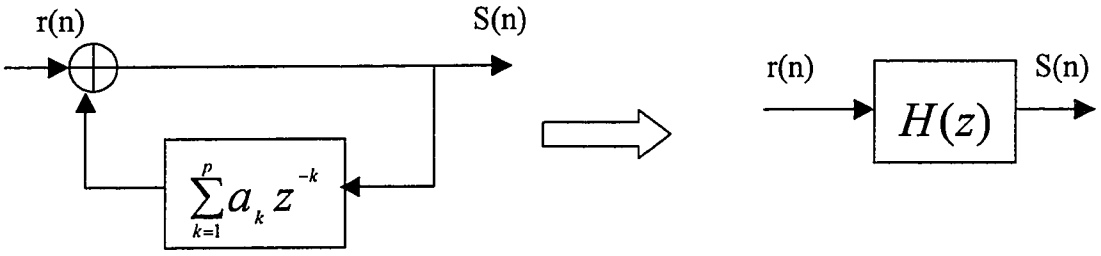


Figure 3.2 (b) The Linear Prediction Synthesis filter [5]

From the above figure, we can see that the prediction coefficients set a_k {k =1: P} is responsible for forming an estimate signal that should be as close as possible to the value of the actual speech sample. In fact the predictor coefficients are estimated by minimizing the energy of the prediction residual, E, given by:

$$E = \sum_{n=0}^{N-1} r^2[n] \qquad (3.8)$$

Where N is the number of speech samples in a frame. The number of the prediction coefficients or in other words the order of the prediction filter P is variable and depends on the application. This order of the linear prediction is selected according to a compromise between the transmission bit rate allowed, the computation time permitted and the size of memory allocated to the prediction on one side with the quality of the reconstructed coded speech and the spectral accuracy on the other side. In normal model-based speech coders, this number is usually in the range between 8-16 [32] with the most popular, widely used order of 10 in the G.723 and G.729 ITU-T standards. The basic concept is to allocate a pair of poles to each formant present in the speech spectrum and to allocate additional poles, usually between 2 to 4 poles to approximate the possible zeros. In the concealment of the coded speech however, it has been noticed that higher prediction order can lead to better results. In the ANSI standard B [54] the order implanted was 20 and in the system we propose, the order of the prediction is 50.

3.3 Calculation of the Linear Prediction Coefficients

In the previous section it has been shown that the vocal tract model can be faithfully represented by the H (z) synthesis filter. This filter must be a time-varying filter whose coefficients are changed with time. Because the vocal tract moves relatively slowly, speech can assumed to be a random process whose properties vary slowly. This led to the basic short-time stationary assumption used for LPC analysis. This assumption is valid for short intervals up to 30 ms. The LPC system should determine how frequently these

coefficients should be updated based again on compromising between the computational load that increases with updating the coefficients more frequently and the speech quality that relatively degrades with assuming longer periods of stationary speech signals. These coefficients are, as previously mentioned, estimated by the least-mean squares method to minimize the energy of the residual signal. There are two widely used methods for estimating the LP coefficients [5], [32]:

- The Autocorrelation
- The Covariance

Both methods are primarily time-domain techniques that are easily implemented on DSP processors. As the proposed method for the LPC-based concealment of the PCM coded speech presented in this thesis uses the autocorrelation method, we will focus on how to compute the LP filter by using this method. We will also discuss the various approaches to calculate the autocorrelation in order to select the appropriate least costly method from the computational prospective.

3.3.1 The Autocorrelation Method

In the autocorrelation method, a moving window is used to divide the speech into stationary analysis frames. This is illustrated in Figure 3.3. For illustration purpose only, a window of triangular shape is used to indicate the autocorrelation window.

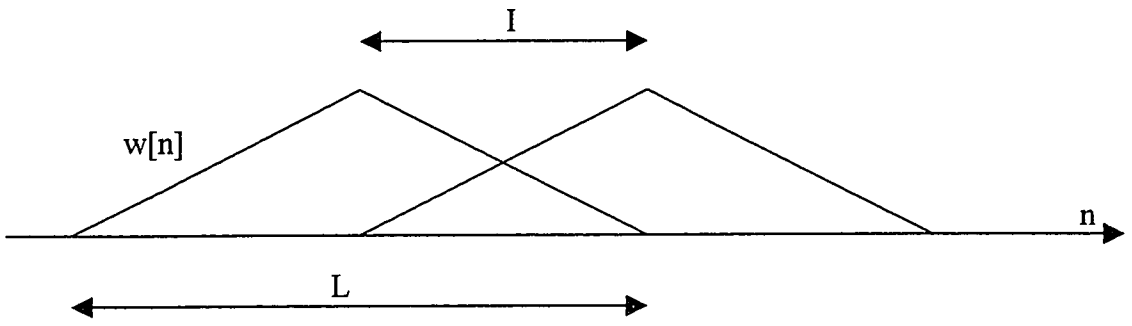


Figure 3.3: Sliding Window Application to Speech Signal for Autocorrelation Analysis

The above figure shows the method of applying the window w[n] on the speech signal. The speech frame length is referred to as L while the symbol I refers to the overlap period between successive windowed frames. This period in most cases is half the frame length. For each placement of the window, usually from 10 to 30 ms apart [32], the speech signal is windowed to create one analysis frame of the signal. The result is a signal that is infinite in extent, but is zero everywhere outside the window. Thus it is possible to compute the autocorrelation function for the entire windowed signal. The windowed signal is referred to as $S_w(n)$ and is calculated according to the equation:

$$S_w(n) = S(n)\, w(n) \qquad (3.8)$$

After the windowing step, the autocorrelation of the windowed speech signal is computed. The autocorrelation function of the windowed signal $S_w(n)$ is

$$P_{ss}(l) = \sum_{n=0}^{N-l-1} S_w(n)\; S_w(n+l) \qquad l = 0, 1, \ldots P \qquad (3.9)$$

where N is the size of the autocorrelation window and P is the order of the filter.

The minimum square error within the interval $0 \le n \le N-1$ is defined as:

$$E = \sum_{n=0}^{N-1} r^2(n) \;= \sum_{n=0}^{N-1} \left(S_w(n) - \sum_{k=1}^{p} a_k\, S_w(n-k) \right)^2 \qquad (3.10)$$

To optimize the LP filter coefficients, this energy should be minimized. Setting the partial derivative of the energy with respect to the filter coefficients to be zero produces:

$$\frac{\partial E}{\partial a_k} = 0 \qquad 1 \le k \le p \qquad (3.11)$$

By substituting Eq. (3.10) in Eq. (3.11) we obtain the following equation with P unknown coefficients of the LP parameters set a_k:

$$\sum_{k=1}^{p} a_k \sum_{n=0}^{N-1} S_w(n-l)\; S_w(n-k) = \sum_{n=0}^{N-1} S_w(n-l)\, S_w(n), \qquad 1 \le l \le p \qquad (3.12)$$

By considering equation (3.9) and noting that the autocorrelation function is even with $P_{ss}(l) = P_{ss}(-l)$ we can see that the above equation is equal to:

$$\sum_{k=1}^{p} P_{ss}(|l-k|)\, a_k = P_{ss}(l) \qquad (3.13)$$

The above equation can be represented in the following matrix form:

$$\begin{bmatrix} P_{ss}(0) & P_{ss}(1) & \cdots & P_{ss}(p-1) \\ P_{ss}(1) & P_{ss}(0) & \cdots & P_{ss}(p-2) \\ & & \ddots & \vdots \\ \vdots & & & \\ P_{ss}(p-1) & \cdots & & P_{ss}(0) \end{bmatrix} \begin{bmatrix} a_k \\ a_2 \\ \\ \\ a_k \end{bmatrix} = \begin{bmatrix} P_{ss}(1) \\ P_{ss}(2) \\ \\ \vdots \\ P_{ss}(p) \end{bmatrix} \qquad (3.14)$$

For the computation of the coefficients a_k, the equation with the inverse of the autocorrelation matrix requires computations of the order P^3 and may be numerically unstable. To reduce the computational complexity and improve the numerical stability, the recursive Levinson-Durbin procedure can be used [14], to exploit the symmetry and the Toeplitz structure of the matrix to be inverted. The computational complexity of the Levinson- Durbin algorithm is of order P^2. The procedure is:

$a_m(k)$ is the k^{th} coefficient in a vector or array of m coefficients, $1 \le m \le P$, $1 \le k \le m$

initialization:

$$a_1(1) = -\frac{P_{ss}(1)}{P_{ss}(0)} \qquad (3.15.1)$$

for $2 \le m \le P$:

for $k = m$:

$$a_m(k) = \frac{-P_{ss}(m) + \left[P_{ss}(m-1) \cdots \; P_{ss}(1)\right]\left[a_{m-1}(1) \ldots a_{m-1}(m-1)\right]^T}{P_{ss}(0) + \left[P_{ss}(m-1) \cdots \; P_{ss}(1)\right]\left[a_{m-1}(m-1) \ldots a_{m-1}(1)\right]^T} \quad (3.15.2)$$

for $1 \le k < m$:

$$a_m(k) = a_{m-1}(k) + a_m(m)\, a_{m-1}(m-k) \quad (3.15.3)$$

At the end of the procedure, the P coefficients $a_p(k)$ in the array a_p are the P[th] order AR model coefficients $a\ (k)$.

From the above equation we can calculate the number of MACs required for the Levinson-Durbin algorithm as follows:

1. Equation (3.15.1) includes 1 division that is calculated only once.
2. Equation (3.15.2) includes:
   (a) 1 division per loop that is repeated (p-1) times: (p-1) total divisions.
   (b) In the numerator: (m-1) multiplications per each loop with the value of ***m*** increasing by 1 with each running of this recursive loop. ***m*** starts at 2 and ends at P so (m-1) have initial value of 1 up to (p-1). The total number of multiplications of that step in the algorithm is given by:

   $$\sum_{m=1}^{p-1} m = \frac{(p-1) \times p}{2}$$

   (c) In the denominator: if the output value of the 2 matrices multiplication is saved in memory, we will only need 1 multiplication for each loop and so the total number of multiplications that this step contributes for the whole algorithm execution is (p-1) multiplications.

3. Equation (3.15.3): Same case as step 2.b. Here we have (m-1) multiplications that are repeated recursively with the value ***m*** increasing by 1 with each

execution and so the value of (m-1) starts at 1 and ends at (p-1) and it contributes to the total number of multiplication that the algorithm executes with the same number as step 2.b $\sum_{m=1}^{p-1} m = \frac{(p-1)\times p}{2}$ multiplications.

From the above steps we can estimate the total complexity of the Levinson-Durbin algorithm. We can follow the same procedure as in the ITU-T G.711 Appendix. A [12] and both the ANSI standard A and B for the G.711 concealments [48], [54]. In these standards, a single Multiplication and Accumulation (MAC) is given the weight of one DSP cycle and a division or square root process is estimated to consume 10 cycles while a comparison process is given the weight of 2 DSP cycles. Hence the total complexity (number of multiplies) of the Levinson-Durbin algorithm is given in terms of the prediction order P as:

$$1\times 10 + (P-1)\times 10 + \frac{P\times(P-1)}{2} + (P-1) + \frac{P\times(P-1)}{2} = P\times 10 + (P-1) + P\times(P-1)$$

$$\downarrow \qquad \downarrow \qquad \downarrow \qquad \downarrow \qquad \downarrow$$

$$1 \qquad 2.(a) \qquad 2.(b) \qquad 2.(c) \qquad 3.$$

Thus the total complexity is given by: $P^2 + 10\times P - 1$ MACs

It is clear that this complexity is of the order P^2 and is considered reasonable for the DSP processors. In fact, in most of the model-based codecs where the LPC filter order P is small, the complexity of the Levinson-Durbin algorithm is less than that required for the autocorrelation calculation of the speech samples. This complexity is considered negligible compared to the complexity of searching the code-book used to store the excitation values if any is used (as in the case of the G.729 and G.723 ITU-T model-based standard codecs). Briefly, the Levinson-Durbin algorithm is considered an affordable algorithm for most of the DSP processors and does not constitute the main complexity concern when a certain LPC coder is designed.

However, the square relation between the LPC coder order P translates any increase in the LPC order to a corresponding squared increase in the complexity load. For a high-order LP filter the complexity of the Levinson-Durbin algorithm starts to get high and so does the computational load required for the autocorrelation calculation, as more coefficients are needed. Eventually, optimization should be done to the autocorrelation-based method of the LP coefficients calculation. While the Levinson-Durbin algorithm is already optimized to reduce the complexity of the LP coefficients estimation to the order P^2, the implementation of the autocorrelation function in the time-domain can be inefficient for excessively large number of coefficents. Actually, it is more efficient in this case to calculate the autocorrelation function in the frequency domain using the Fast Fourier Transform (FFT). This is discussed in more details in the next sections.

3.3.2 Autocorrelation Calculation

We have discussed in the previous section the method of calculating the LP coefficients using the autocorrelation values of speech samples. In this section, we shall focus on the different techniques of calculating the autocorrelation function and the efficiency of each of these techniques.

For a finite energy signal, the autocorrelation function is defined in the time-domain as:

$$P_{xx}(l) = \sum_{n=-\infty}^{\infty} x(n)\, x(n+l) \; = x(n) \; * \, x(-n) \qquad (3.16)$$

For a finite sequence of N samples, the above equation becomes

$$P_{xx}(l) = \sum_{n=0}^{N-l-1} x(n)\, x(n+l) \qquad 0 \le l \le N-1 \qquad (3.17)$$

The autocorrelation function is an even function where $P_{xx}(l) = P_{xx}(-l)$. In the case of the calculation of LP-coefficients of order p we need only the first p values of the autocorrelation outputs and the equation in this case becomes the same as equation (3.9)

$$P_{xx}(l) = \sum_{n=0}^{N-l-1} x(n)\, x(n+l) \qquad l = 0, 1, \cdots P \tag{3.18}$$

In terms of the computational complexity we can compare the complexity of both the complete autocorrelation calculation in equation (3.17) and the (P+1) coefficients autocorrelation calculation in equation (3.18):

- In Equation (3.17) we can see that the total number of MACs is:

$$N + (N-1) + (N+2) \cdots + 1 = \frac{N \times (N+1)}{2}$$

- In Equation (3.18) the total number of MACs is reduced to:

$$N + (N-1) + (N-2) \cdots + (N-p) = \frac{(N+N-p) \times (p+1)}{2} = N \times (p+1) - \frac{p \times (p+1)}{2}$$

We can see from the above that the time-domain calculation of the autocorrelation function becomes more costly if the number of coefficients required increases and eventually approaches the maximum complexity present in the calculation of the whole set of autocorrelation coefficients evaluated from Equation (3.17).

There exists another method of calculating the autocorrelation function. This method uses the FFT and translates the calculations to frequency domain. It provides the complete set of autocorrelation coefficients and thus is not widely used in the model-based codecs that need only a small number of coefficients. However, in the cases where we need to calculate the complete autocorrelation function this frequency domain method proves to be very efficient and often requires a lower complexity than the time-domain implementation.

3.3.3 Autocorrelation Calculation in the Frequency Domain

In this method the autocorrelation calculation is performed in the frequency domain with the aid of the Fast Fourier Transform (FFT).

The basic equation of the autocorrelation calculation in the frequency domain is given by:

$$r_{xx}(k) = X(k) \times X(-k) \qquad 0 \le k \le 2N_1 - 1 \qquad (3.19)$$

where $r_{xx}(k)$ is the Fourier-transform of $P_{xx}(l)$, $X(k)$ is the Fourier transform of the signal x (n) and N_l is the next higher power of 2 to the original sequence length N. It is known that in order to perform the convolution of two time-domain sequences in the frequency domain we have to append to both sequences zeros to get new sequences having the length equal to the sum of their original lengths. Thus, we need to perform the $2N_l$- point FFT on the signal x(n) to get $X(k)$ to be able to perform the convolution operation.

The calculation of the autocorrelation function in the frequency domain is achieved through the following ordered steps:

1. Calculating the $2N_l$- FFT of the signal x (n)

   $$x(n) \xrightarrow{FFT} X(k) \qquad (3.20.1)$$

2. Performing the multiplication of the signal $X(k)$ by $X(-k)$

   $$r_{xx}(k) = X(k) \times X(-k) \qquad 0 \le k \le 2N_1 - 1 \qquad (3.20.2)$$

3. Calculating the Inverse Fast Fourier Transform (IFFT) of the signal $r_{xx}(k)$

   $$r_{xx}(k) \xrightarrow{IFFT} P_{xx}(n) \qquad (3.20.3)$$

The complexity of the autocorrelation calculation in the frequency domain differs between complex signals and real signals. In the following section we present complexity estimation for the above equations for a general complex signal that could have an imaginary part. An optimization of the complexity for a real sequence input and output is then presented in another section.

3.3.3.1 Complexity Estimation of Autocorrelation function in the Frequency Domain

For a general signal, the complexity of the implementation of the autocorrelation function using the FFT can be estimated based the equation set (3.20):

1. **FFT of size $2N_1$**: this is equivalent to $N_1 \times \log_2(2N_1)$ complex multiplies.
2. **Multiplication of $X(k) * X(-k)$**: This requires $2N_1$ complex multiplies.
3. **IFFT of size $2N_1$**: this is equivalent to $N_1 \times \log_2(2N_1)$ complex multiplies.

The total complexity is: $2 \times N_1 \times \log_2(2N_1) + 2N_1$ complex multiplies.

Knowing that a complex multiplication is equal to four real multiplications and 2 real additions, the above total complexity should actually be multiplied by 4, which yields: $8 \times \left(N_1 \times \log_2(2N_1) + N_1\right)$ real multiplies.

3.3.3.2 Complexity Estimation of Autocorrelation In The Case of Real Input

Optimization techniques to reduce the complexity of the FFT operation for a real sequence g (n) are presented in [14] and are summarized here.

1. Computing the Discrete Fourier Transform (DFT) of g(n), a $2N_1$-point real sequence, using only N_1-point FFT as follows:
   First, we define

$$\begin{aligned} x_1(n) &= g(2n) \\ x_2(n) &= g(2n+1) \end{aligned} \qquad 0 \le n \le N_1 - 1 \qquad (3.21.1)$$

   Thus, we have subdivided the $2N_1$ real sequence into two N-point real sequences. Then let $x(n)$ be a complex valued sequence defined as

$$x(n) = x_1(n) + jx_2(n) \qquad 0 \le n \le N_1 - 1 \qquad (3.21.2)$$

   From the linear property of the DFT operation we have:

$$X(k) = X_1(k) + jX_2(k) \qquad (3.21.3)$$

$X_1(k)$ and $X_2(k)$ can be then extracted from $X(k)$ by applying the following two equations:

$$X_1(k) = \frac{1}{2}\left(X(k) + X^*(N_1 - k)\right)$$
$$X_2(k) = \frac{1}{2j}\left(X(k) - X^*(N_1 - k)\right) \qquad (3.21.4)$$

The division by 2 is not necessary as we are not interested in absolute magnitudes of the autocorrelation coefficients i.e. the absolute magnitudes are not necessary for the LP coefficients extraction or the pitch estimation. The division by j and the complex conjugate operations can be seen as changing the sign and storage location of real and imaginary parts. Thus the above equations can be seen as addition operation.

Finally, we must express the $2N_1$- point DFT in terms of the two N_1- point DFTs, $X_1(k)$ and $X_2(k)$. To accomplish this we proceed as in the decimation in time FFT algorithm to get the following expression for the FFT of g (n) expressed as $G(k)$

$$G(k) = X_1(k) + e^{-j\frac{\pi k}{N}} X_2(k) \qquad k = 0, 1, \cdots N_1 - 1$$
$$G(k+N) = X_1(k) - e^{-j\frac{\pi k}{N}} X_2(k) \qquad k = 0, 1, \cdots N_1 - 1 \qquad (3.21.5)$$

From the above equations we can see that actually the complexity of a $2N_1$ - FFT and IFFT is reduced to the complexity of a N_1- point FFT or IFFT and some additional computation. By observing that at least the 2nd half the $2N_1$ real input sequence is zeros for the case of autocorrelation calculation we can find that actually one of the FFT stages can be dropped. The complexity evaluation of the above real signal FFT algorithm yields:

$$4 \times \left(\frac{N_1 \times (\log_2(N_1) - 1)}{2} \right) + 4 \times N_1 = 2N_1 \log_2 N_1 + 2N_1 \qquad \text{real multiplies (a)}$$

This is the complexity of the FFT operation. However, in the case of the IFFT operation the output sequence is real and thus the equation (3.21.5) will actually yield 2 N_1 real multiplies (real by real and imaginary by imaginary) instead of 4 N_1. However the fact that the IFFT stages in this case are eight (not reduced to seven as in the FFT where it was reduced to seven because of having more than half of the sequence zeros), the complexity if the IFFT is again the same complexity of the FFT.

$$4\times\left(\frac{N_1\times(\log_2(N_1))}{2}\right)+2\times N_1 = 2N_1\log_2 N_1 + 2N_1 \qquad \text{real multiples} \quad \text{(b)}$$

2. The multiplication of $X(k)\times X(-k)$: From the DFT properties we know that for real sequences the DFT of the autocorrelation is $r_{xx} = |X(k)|^2$. We also have the real sequence property that $X(k) = X^*(N-k)$; the multiplications will be real by real and imaginary by imaginary. Thus a total reduction of factor 4 is obtained because of symmetry. The complexity of this step is reduced to $2N_1$ real multiplies.

From 1, 2 we can see that the actual total complexity of estimating the autocorrelation in the frequency domain for real signals is reduced to:

$$\boxed{4\times N_1\times\log_2 N_1 + 6N_1} \qquad \text{real multiplies}$$

3.3.4 Window Consideration

The first step in the autocorrelation calculation is to apply a window to the signal to get a finite length sequence and thus be able to calculate the autocorrelation function. This fixed length window, referred to as the *analysis window*, can have different shapes. The right shape of the window is important as it allows different samples to be weighted differently. In model-based codecs that adopt the autocorrelation calculation through overlapped windowed speech frame, a window without discontinuities in the time domain is used. It is preferred so that the corresponding side lobes of the window in the frequency domain are low and thus the undesirable ringing effects that could occur with the application of a sharp edge window are eliminated.

The use of the Hamming window is very common in speech analysis. It is a raised cosine window defined as:

$$w(n) = \begin{cases} 0.54 - 0.46\cos\left(\dfrac{2\pi n}{N_w - 1}\right) & \text{for } 0 \le n \le N_w - 1 \\ 0 & \text{otherwise} \end{cases} \qquad (3.22)$$

Other tapered window such as Hanning, Blackman, Kaiser and Bartlett are used on a relatively smaller scale. The window can also be hybrid as the window used in the G.729 coders [10].

In the ANSI concealment standard B for the G.711 coded speech frame erasure, the Hamming window has been employed as the analysis window for the autocorrelation calculation. This standard actually builds a complete analysis unit that is very similar to the model-based coders to extract the speech parameters. It then follows with a synthesis unit to build up the concealed frame using these parameters. The choice of the Hamming window in this case is logical as it follows the rules developed for the model-based coders, namely using a tapered window for autocorrelation calculation [25].

In the system proposed in this thesis however, we have find out that the application of a regular rectangular window gives better results.

$$w(n) = \begin{cases} 1 & for\ 0 \le n \le N_w - 1 \\ 0 & otherwise \end{cases} \qquad (3.23)$$

This could be due to the fact that in the application intended we do not recommend giving small weight to the most recent speech samples in the last correct frame as in the case of applying a Hamming window. This will limit their contribution to the LP formation. This does not present any controversy with the concept of using the Hamming window in the ANSI standard, as the approach we implemented is different and thus no link should govern the choices of the analysis windows that both techniques use.

3.4 Pitch Determination

Pitch determination is crucial for the proper operation of LPC coding. The quality of the LPC synthesis depends, up to a major extent, on how accurate the pitch period is being estimated. As the main and the most obvious property of the voiced speech, the pitch plays a principal role in the formation of the excitation signal. In the vocoder type of model-based coders, it is the source of the voicing decision, which controls the type of the excitation signal, either: impulse train (voiced excitation) or random sequence (unvoiced excitation). The voiced excitation completely depends on the pitch period in forming the pulse train signal that has its period equal to the pitch period.

In the CELP-based codecs, the pitch contribution to the formation of the excitation signal is known as "*the long term prediction*", implemented using " *adaptive code book approach*" [9], [8], [10].

In the following sections, several pitch detection algorithms will be discussed. Among these systems, we shall focus on the autocorrelation approach as the most popular approach in pitch detection. It has also been the method implemented for the pitch detection in our proposed concealment algorithm.

3.4.1 Pitch Detection Categories

Pitch period can be defined in different ways depending on the domain we define it in. In the time-domain, we can define it as *the period at which the voiced speech segments are repeated.* This definition is not completely accurate as the speech segments are not exactly repeated, but a similar, very close in waveform, signal is repeated. In the frequency domain, it is defined as " *The reciprocal of the fundamental frequency of the signal*" [1]. Eventually, the different significances of the pitch definition were the reason why many approaches have emerged for its detection. Some of these approaches are very theoretical but difficult to implement in real systems while others already exist in the industrial application of speech coders. In the following two sections we shall briefly discuss the most important, widely used pitch determination techniques.

3.4.2 Frequency-Domain Approach

In frequency domain approaches, the goal is the determination of the fundamental frequency in a speech segment. If the speech is voiced, several closely located peaks in the frequency spectrum or the |FFT| of the signal will appear [5]. The distance between two consecutive peaks is the fundamental frequency f_o (the reciprocal of the pitch period).

3.4.2.1 Cepstral Pitch Determination

In another approach for pitch estimation, the concept of "*Cepstrum*" [14] is applied. Assuming that the voiced speech results from the convolution of the glottal excitation sequence and the impulse response of the vocal-tract, in the frequency domain this is

equal to the multiplication of the frequency response of these two spectra, using the fact that:

$$\log(AB) = \log A + \log B \qquad (3.24)$$

Thus, the multiplication process can be transformed into an addition process.

Assuming the excitation signal is x(n) and the impulse response of the vocal tract is h(n) then we have the output speech expressed as:

$$y(n) = x(n) * h(n) \qquad (3.25)$$

Taking the FFT of both sides, the corresponding equations in the frequency domain are:

$$Y(f) = X(f) \times H(f) \qquad (3.26)$$

Applying the logarithmic rule from equation (3.24)

$$\log(Y(f)) = \log(X(f)) + \log(H(f)) \qquad (3.27)$$

Taking the IFFT of both sides of the equation:

$$C_y(n) = C_x(n) + C_H(n) \qquad (3.28)$$

where:

- $C_x(n) = IFFT\left[\log(X(f))\right]$ is called the cepstrum of the excitation signal x(n), which is responsible for the fast variations in $Y(f)$ and thus is characterized with a cepstrum that is concentrated in the high values of n in $C_y(n)$.

- $C_H(n) = IFFT\left[\log(H(f))\right]$ is the cepstrum of the impulse response of the vocal tract model that has the main role in the formation of the slow variations in $Y(f)$. Thus, the cepstrum of this slow varying $H(f)$ is mainly concentrated in the small value of n in $C_y(n)$.

- $C_y(n) = IFFT[\log(Y(f))]$. This is the cepstrum of the resulting speech signal, which is composed of the addition of the cepstrums of the excitation and the vocal tract impulse response.

From the perspective that each of the cepstrums is concentrated in two different regions, we can subdivide the cepstrum of the resulting speech signal into its two main components through the use of a high-pass or a low pass window in the time-domain. Having isolated the cepstrum of the speech excitation we can obtain the pitch component by reversing the logarithmic process.

$$C_y(n) \longrightarrow C_x(n) + C_H(n) \qquad (3.29)$$

$$C_x(n) \xrightarrow{FFT} \log(X(f)) \qquad (3.30)$$

$$\log(X(f)) \xrightarrow{Exponential} X(f) \qquad (3.31)$$

3.4.3 Time Domain Approach

3.4.3.1 Maximum Likelihood In Time

We have discussed in the last section one of the most widely used frequency-domain pitch detection methods. That method is based on the frequency-domain interpretation of the significance of the pitch value as the reciprocal of the fundamental frequency in the spectrum.

The time domain approach to pitch detection, in the same context, is built on the time-domain pitch definition. For voiced speech segments, the pitch value refers to the span of time during which the voice signal exhibits a maximum likelihood with the neighbouring periods. The similarity of pitch period speech segments has been the key for the most popular and most widely used pitch detection technique that is namely "*The Autocorrelation Approach*".

The autocorrelation or the maximum likelihood approach of pitch detection is performed through searching the peaks of the autocorrelation coefficients of sufficiently long speech segments that contain at least two pitch periods. It is known that for a periodic input sequence, the autocorrelation function yields a maximum output at the period value of the input. Conceptually, the voiced speech segments are semi-periodic and by applying the autocorrelation function on these segments we can detect the pitch value to be the first maxima (after the peak at "0" sample) in the resulting sequence.

This simple idea has been even more elaborated by applying constraints on the permitted values of the pitch periods. The range of acceptable detected pitch value differs according to the application and in fact different ranges have been used for the different ITU-T standard codecs. In G.728 [17], for example, the lowest permissible value is 20 samples of 8 kHz sampled speech or 2 ms while in other codecs this value can be raised up to 40 samples or 4 ms.

The range of pitch values differs from male voices to female voices. The female voices have smaller pitch periods in the range 3 or 4 ms up to 7 or 8 ms while the male pitch values occupy the upper range between 7 or 8 ms up to 20 ms in some cases [5], [12]. The most widely used range for pitch detection is 40 samples up to 120 samples. This range covers most of the possible pitch values and offers acceptable performance.

In the error concealment of the G.711 PCM coded speech, the ITU-T standard G.711 Annex A [12] is totally based on the long-term prediction. It basically depends on estimating the pitch period and then playing the last correctly received pitch period. This concealment tool employs a similar time-domain method to detect the pitch value, following the above range of pitch values of 40 to 120 samples.

As we will show, the concealment algorithm for PCM coders presented in the thesis can be combined with the G.711-A standard concealment tool to improve its performance. It is thus essential to present an estimate of the complexity load of the time domain

approach of pitch detection. This pitch detection method will actually contribute to the total complexity estimate of the concealment technique we are presenting

3.4.3.2 Complexity Calculation of the Autocorrelation approach

To determine the overall complexity:

1. Calculate the autocorrelation function of a sufficiently long speech segment.
2. Search for the peaks of the autocorrelation outputs. This is equivalent to a comparison process.

We have already estimated the complexity of the autocorrelation function in the previous section. In the concealment algorithm proposed in this thesis these autocorrelation coefficients are used twice. They are used to estimate the LP coefficients and to extract the pitch value.

The second step is the only complexity contribution of the pitch detection unit to the total complexity of a concealment algorithm proposed approach in Chapter 7. Eventually, we can see that this step can be optimized to reduce its complexity by performing it on two stages: a coarse search and then we follow by a fine search. Coarse search means we search for the peaks at relatively wide span of samples. Locking on the maximum values of the coarse search a fine search then follows to accurately estimate the pitch value from the relatively narrow searching zone.

This two-stage subdivided approach to perform the comparison significantly reduces the complexity of the search step especially considering the fact that it is more computationally exhaustive than the multiplication accumulation (MAC) process. In the ITU-T and the ANSI concealment standards a comparison is estimated to consume two processing cycles compared to 1 processing cycle for a MAC. Choosing to follow the same ranges of samples for the coarse and fine search regions as the ITU-T standard

G.711 Annex A, we can estimate the complexity of the search step to be the same as the complexity estimate in the G.711 Annex A that is the following:

$$86\ Comparisons \times 2\ \ Cycles / Comparison = 192\ Cycles$$

This is equivalent to 192 MACs and is a negligible complexity indeed.

However, if the autocorrelation function is calculated in the time domain to extract limited number of coefficients (to be used in the LP coefficients calculation and as initial conditions for the inverse LP-filter), the pitch determination complexity will include the complexity presented in the 1st step. The total complexity in this case will be equivalent to that is presented in the ITU-T G.711-A [12].

$$3764\ MACs + 86\ Comparisons \times 2\ \ Cycles / Comparison = 3976\ Cycles.$$

3.5 Summary

In this chapter, we reviewed the different mathematical operations involved in the Linear Prediction (LP) process. This was necessary as the new concealment algorithm, which we will present later in this thesis, relies on Linear Prediction to estimate the missed segments in the speech stream. The main operations discussed were the LP coefficients estimation, the autocorrelation calculation and the pitch detection. As we discussed in Chapter 2, the synthesis filter at the decoder side is represented as an all-pole adaptive filter. The LP coefficients are implemented as the poles of this filter and thus have the main role in shaping the resulting speech signal. There are two methods to estimate the LP coefficients set, namely the covariance method and the autocorrelation method. The autocorrelation method is the one that is usually implemented in the model-based coders and thus was chosen to be presented in more details. In autocorrelation method, the LP coefficients are estimated by minimizing the energy of the residual signal. This residual signal is equal to the difference between the original speech segment and an estimated signal produced by the synthesis filter. We presented the complete mathematical

explanation of the LP coefficients estimation using the autocorrelation method. We then proceeded to the Levinson-Durbin algorithm used to extract the LP coefficients. This algorithm ensures stability of the estimated LP coefficients and actually reduces the complexity from cubic order (proportional to the number of LP coefficients) to square order.

The autocorrelation function is essential in the estimation of both the LP coefficients and the pitch. It constitutes a large part of the complexity of the concealment algorithm that we present. Thus, we had to perform an intensive investigation of the possible methods that could help in reducing the complexity of this algorithm. It was shown that the calculation of the autocorrelation function in the frequency domain is more efficient from the complexity prospective than the direct time-domain calculation especially if the input signal is real valued as in the case of speech signals. The complexity estimates of both the time domain and frequency domain calculation of the autocorrelation were presented. We also discussed the optimized algorithm for frequency domain calculation of real-input sequences.

We then proceeded to the pitch determination. The pitch period can be estimated in time domain or in the frequency domain where its reciprocal, namely the fundamental frequency, is estimated. One of the frequency domain pitch estimation techniques was presented. In the time domain, the most widely used pitch determination algorithm is the maximum likelihood in time. This algorithm depends on calculating the autocorrelation function and then searching for the peak of the resulting coefficients. This method is in fact the one we proposed to estimate and extract the pitch period in our proposed concealment technique. Thus, a complexity estimate of this pitch estimation method was calculated. It was mentioned that there exists another option that could be used in our algorithm to detect the pitch value. This pitch detection algorithm used in the ITU-T G.711-A concealment algorithm can be used in the case we don not calculate the whole set of autocorrelation i.e. use the time-domain autocorrelation approach to calculate only the autocorrelation coefficients used in the LP coefficients extractions. Both techniques are valid and produce very similar results.

4 Voice Perception and assessment tools

4.1 Overview

This chapter deals mainly with the assessment tools used in communication systems to distinguish between the qualities of different speech coders. Quality assessment is critical to the coding techniques existing or the new ones that emerge every year. New communication links, especially optical fibres, enabled communication systems to tolerate large bandwidth signals more than ever. In fact, the bandwidth limitations become less restrictive with the progress of physical medium technology. Nowadays, engineers re-evaluate coding systems that were accepted as standards with the main advantage of low bandwidth requirements. Such bandwidth efficient systems that do not guarantee high quality become more and more obsolete. Eventually, the main criterion in evaluating codecs is becoming quality again.

A second major reason for considering the quality as the principal factor in the choice of codecs is the user satisfaction. Users receiving data at the end-machine put the quality as the main issue to consider. Complexity and implementation become just the task of engineers. Actually, assessment tools can play a major role in more powerful non-intrusive monitoring of voice quality [37].

Assessment tests are also a principal tool used in the work associated with the new concealment technique developed in this thesis. They provide an unbiased judgment on the improvement of the quality that has been achieved compared to existing standards. This chapter begins with presenting a model of the perception of voice by the ear to understand how human factors affect the evaluation of the quality.

4.2 The Speech Chain

A helpful way of demonstrating what happens in the perception model is to begin with the chain of events employed in transmitting the speech information referred to as the *speech chain* [6]. Figure 4.1 shows the flow of the speech chain.

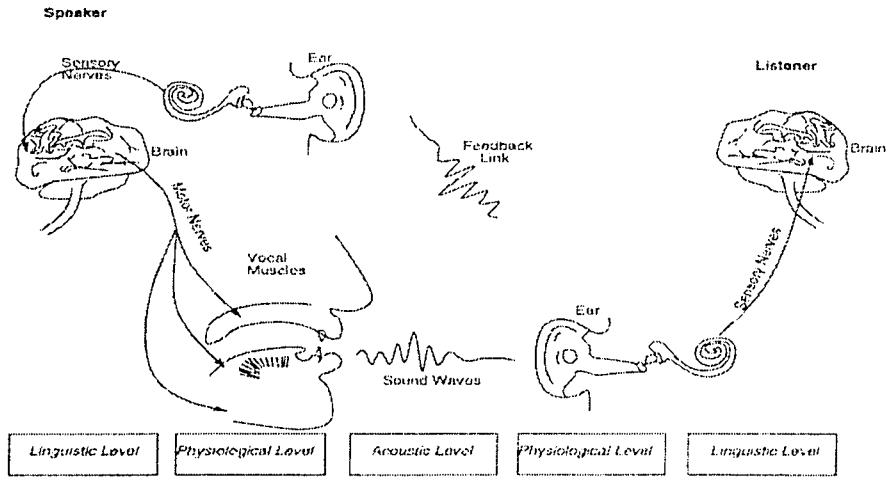


Figure 4.1: Speech Chain [33]

A speaker first arranges his thoughts, decides what he wants to say and forms appropriate words to express these thoughts in a *linguistic* way. This is done under the supervision of the speaker's brain that performs the mental operation of logical formation of speech, grammatical structure, etc. It then activates the vocal organ nerves that control the tongue, jaw, lips and vocal cords in order to physiologically produce the speech [6].

The listener, on the other hand, receives the speech signal and activates the hearing mechanism. The hearing mechanism starts with the acoustic nerve (a sensory nerve) that perceives nerve impulses and delivers them to the brain. These nerve impulses activate the auditory section in the brain to translate it into a recognizable and understandable impression to the listener.

The discussion above reveals that the voice starts as a *linguistic information* at the speaker and ends also as *linguistic information* at the listener. In between, this linguistic

information is translated into the physiological level as it is pronounced and then descends to the acoustic level that is delivered to the listener, who performs the inverse chain of translation back to the linguistic level.

4.3 Assessment Tools

4.3.1 Overview

Assessment tools emerged in response to the need of service providers to evaluate the service delivered to the customers. Quality is an important aspect in voice transmission and the principal factor that is evaluated by users when choosing to subscribe in voice services.

In this section, a general classification of assessment tools will be first introduced. Following, this ITU-T standard of Perceptual Evaluation of Speech Quality (PESQ) namely the P.862 will be presented in more details. This tool is used in this thesis to evaluate the quality of the concealment algorithm proposed and to assess the quality of the two concealment standards we compared our technique to. It was used to produce the results we present in Chapter 7 and Appendix A. Thus, it is essential to present its principle of work and emphasize its high fidelity.

4.3.2 Classification of Assessment Tools

Assessment tools can be classified either according to the nature of the used tool [39], [38], [2], or according to the generality of the test performed [16]. The second classification is not widely used. Most communication systems prefer to follow the first classification, as it is well defined and no ambiguity can be faced when distinguishing between its classes.

For a given tool, assessment techniques can be categorized into either subjective or objective tests. Subjective tests are based on the fact that only human perception determines the quality. Thus, these tools rely on people to assess the quality. This method is considered the most efficient and preferred method in quality evaluation [7]. A more detailed specification on how these tests are performed is covered by ITU.P830 [40].

The other type of assessment tools is the objective tests. This is simply the assessment of the speech quality through some mathematical analysis of the speech signal without the interference of a human as a listener. Objective assessment tools are cheaper and faster compared to the subjective tests. However, most of the systems targeting very accurate evaluation prefer using subjective tests. The International Telecommunication Union (ITU) P.800 [39] standardized the objective and subjective tests most often used.

4.3.3 Subjective Tests

Subjective tests can be defined as *the assessment tools that depend on the interaction of the human to the voice files under inquiry.* It is usually the method applied when accuracy of the test is essential and more crucial than the cost or time aspects. The subjective tests are performed either by trained groups of people or is simply done through random groups of people. They are irreproducible tests, due to their human nature and there is a high time consumption related to the tests and expensive costs.

Current speech research efforts rely on two major categories of subjective tests. Intelligibility tests most often follow the Diagnostic Rhyme Test (DRT) paradigm [32]. In this test, the listener is asked to distinguish between word pairs that differ in only the first consonant sound [16]. The listener sees both written words first and then hears one of the pair. The result is reported as a percentage of correct response according to the following formula:

$$DRT = \frac{Correct - Incorrect}{Total} \times 100 \qquad (4.1)$$

The other category of the subjective tests is the quality test. It attempts to rate how the speech sounds relative to the presence or the absence of degrading coding artifacts [7] .As some coded speech may exhibit "mechanical" or "buzzy" performance but still remain understandable, subjective quality tests assess the point while intelligibility tests will fail.

The commonly used method is the Absolute Category Rating (ACR) [53], where the quality of speech can be rated as a value between 1 and 5 according to the following scale [39]. This scale is known as the Mean Opinion Score (MOS).

- 5 = Excellent
- 4 = Good
- 3 = Fair
- 2 = Poor
- 1 = Unacceptable

The outcome of the ACR test can be biased due to several variables such as the speech material, speaker voice characteristics, presentation order and time effects. Another method was proposed to be more immune to such factors, it is referred to as Degraded Category Rating (DCR) and the output result is presented according to a new scale, known as the Degraded Mean Opinion Score (DMOS). It is similar to the ACR except that the quality of the coded speech is compared to a reference file quality [40].

The procedure is that for each speech file the subject listener listens to a reference file and then to a compressed or modified one. The DMOS scale in this case is assigned values between –3 to 3, to indicate the relation between the original and the subject file. The score follows the following scheme [39]

- 3 = Much better
- 2 = Better
- 1= Slightly better
- 0 = About the same
- -1 = Slightly worse
- -2 = worse

- -3 = much worse

Another more realistic scale, from 1 to 5 was presented in [16] as follows

- 5 = Degradation is inaudible
- 4 = Degradation is audible but not annoying
- 3 = Degradation is slightly annoying
- 2 = Degradation is annoying
- 1 = degradation is very annoying

It is worth noting that, although the DCR score seems, *conceptually*, to be superior to the ACR, the existing DCR accuracy may vary among users in a manner that is more noticeable than the variance in the accuracy of the ACR is. This makes people prefer to use the stable ACR method in quality judgment [53].

4.3.4 Objective Tests

Objective tests came to the scene due to the need of both fast and cheap judgment tools to assess the speech quality. The main advantage of this category of tests is inherently tied to their nature of not depending on the human factor, which may result in variable results. Thus, reproducible results are expected from objective tests. They can also be performed on a large scale as no psychological variations are encountered. On the other hand, it is a difficult task to find mathematical formula or group of formulas to indicate the quality of speech.

Several formulas have been utilized as a measure of the objective speech quality.
The Signal to Noise Ratio (SNR) can be easily computed according to the equation:

$$SNR = 10 \log_{10} \frac{\sum_{n=0}^{N-1} s^2[n]}{\sum_{n=0}^{N-1} (s[n] - s'[n])^2} \qquad (4.2)$$

Where s[n] is the original signal, s'[n] is the synthesized decoded output speech. It provides a good basis in the computation of the quality of the waveform coders, but no useful clue can be extracted to indicate the quality of the two other categories, namely the model-based and perceptual based coders [7].

Another measuring tool is the Spectral Distance. Its efficiency comes from the characteristic of not depending on the phase of the synthesized signal compared to the original one as a factor of measurement. Signals with the same frequency content but different short-term phase can sound similarly [7], this makes this method capable of capturing coding degradation due to transits, temporal discontinuities which are more noticeable by human ear. Thus this method is more compatible with the vocoders. The Euclidean distance can be calculated for a certain LP coder according to the equation:

$$\text{Distance}_{Euclidean} = \sqrt{\sum_{i=1}^{P}\left(k_{s_i} - k_{s_i'}\right)^2} \qquad (4.3)$$

Where k_{s_i} and $k_{s_i'}$ are the coefficients of the original and synthesized LP analysis and P is the order of LP analysis.

A third tool is the one standardized the ITU-T recommendation g.107 [47]. This mathematical tool is known as the E-model. The E-model is designed to predict the quality of the audio conversation based on the transmission parameters. These transmission parameters are mainly the type of the codec, the delay encountered in the conversation and the percentage of packet loss. The objective of the E-model is to quantify the impairments caused by those transmission parameters into a rating R. This rating is used as indication on the user satisfaction in a way similar to the MOS score. The range of the R rating is between 0 and 100. The user satisfaction mapping of this rating is shown in the following table [45]:

R-value range	**90-100**	**80-90**	**70-80**	**60-70**	**0-60**
Speech transmission quality category	best	high	medium	low	poor

Table 4.1: Speech Transmission Quality Categories defined in ITU-T G.109

4.3.5 Perceptual Objective Tests

More efficient new approaches standardized by the ITU-T are designed to predict the MOS rating of a listening test. The Recommendation P861 [41] standardizes an objective Perceptual Speech Quality Measure (PSQM). This system maps the original and the coded / decoded synthetic version into a perceptual frequency representation based on the Bark spectral representation, introduced in Chapter2.

Time and frequency masking is taken into consideration, as well as the non-linear perceptual power levels. After this transformation comes the step of perceptual domain comparison applied to both the original and the degraded signal, then the level of the difference in the perceptual domain is mapped back to the MOS number based on an experimentally derived mapping function [41].

4.4 Perceptual Evaluation of Speech Quality ITU-T P862

The objective method proposed in the P.862 recommendation is aimed at predicting the subjective quality of end-to-end speech. The PSQM method, as described in the ITU-T Recommendation P.861 [41], was only recommended for assessing the quality of speech codecs. It was not able to take account of variable delay, filtering and burst distortions. PESQ [42], on the other hand, was implemented to take proper account of such effects through transfer function equalization and time alignment. It also proposes a new algorithm for averaging the distortion over time. The key process of the PESQ operation is to use a perceptual model analogous to the psychological representation of the original and degraded signal in the human auditory system (refer to section 3.3) taking into account perceptual frequency "*Bark*" and loudness "*Sons*". The following figure taken after [42] shows the conceptual comparison process:

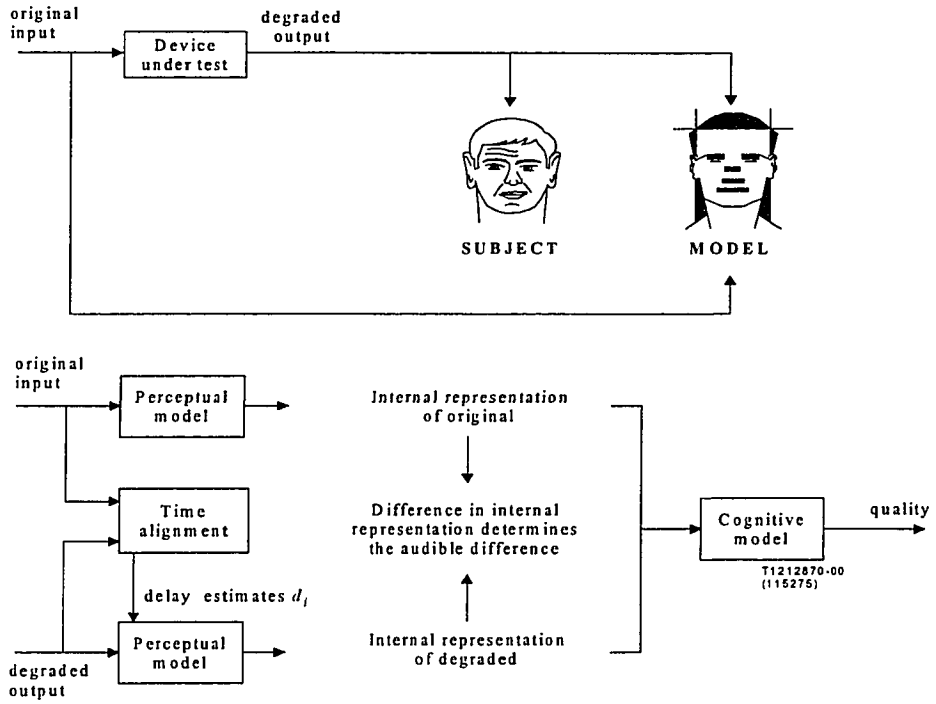


Figure 4.2: Overview of the philosophy used in PESQ [42]

PESQ compares the original signal X (t) with a degraded signal Y (t). The output of the PESQ is the prediction of the subjective assessment quality that would be given to the degraded signal Y (t). An optimum quality score is given to the reference signal X(t). This makes the MOS score of the subject degraded signal Y(t) comparable to ACR scores.

A proof of the good performance of the P.862 perceptual test can be revealed by considering the correlation it exhibits with the subjective scores in a benchmark. This correlation was reported to be around 0.935 [42] for both the known and unknown data. The absolute error between the objective and subjective results investigated was on the average less than 0.25 MOS (±0.25 on a 5-point scale) for 72.3% of the conditions studied and less than 0.5 MOS (±0.5 on a 5-point scale) for 91.1% of the conditions.

Another proof of the excellent quality of assessment of this tool is provided in [34]. It has proved that the G.711 PCM speech files and its distorted files are accurately evaluated by the tool. A single case in which this tool deviates from the actual subjective test is one

distortion type: replacement of speech by silence. This distortion causes all perceptual models some difficulty in predicting MOS. Up to about 50 ms of front and back-end clipping can have little to no subjective impact. However, clipping during speech, e.g. packet loss concealment by silence is harshly rated by subjective tests. PESQ scores are found between these extremes. 50 ms clipping typically causes the PESQ MOS to fall by around 0.5 regardless of the location. In all the cases we investigated in the thesis, we have not developed any concealment through silence. Thus, we expect that the scores we obtained with the aid of this tool are faithful indication of quality of the different concealment techniques we tested.

4.5 Summary

This chapter presented a review of the objective and subjective speech quality assessment tools. Although different classifications can apply to the quality assessment tools, they are normally classified as subjective or objective tests. Subjective tests depend on humans to assess the quality of the subject speech files. They are thus irreproducible and time consuming, yet they are the main test implemented if manufacturers seek accuracy. On the other hand, objective tests implement mathematical algorithms without the involvement of human and thus are cheaper and faster but their accuracy is almost always less than subjective tests.

However, a new family of objective tests that simulates the subjective scores has emerged. The most accurate tool of this family is the ITU-T (PESQ) tool P.862. It simulates the performance of the subjective tests by using a sophisticated model of human perception to estimate how human listeners would assess the quality of the subject speech file. This tool is the one we use along with the different implementation stages of the new concealment algorithm presented in this thesis to assess the quality of the resulting speech files.

5. Impairments and Concealment Algorithms

5.1 Overview

The last few years witnessed an explosion in the information technology field and a great advance in the communication systems. This captured the interest of engineers, system designers and individuals to make full use of the emerging new communication systems, such as Internet Protocol (IP), Asynchronous transfer Mode (ATM) and the new trends in the physical transmission media like the Asynchronous Digital Subscriber Lines (ADSL), to deliver voice instead of relying on the traditional technique of the Public Switched Telephone Networks (PSTN).

Configuring these communication systems to adapt to voice transmission is quite a challenge, as they were developed for the non real-time data communication. Being a real-time data, speech communication imposes more demands and constraints that need to be fulfilled by system designers. Impairments that are easily tolerated for non-real time transmission can turn into a very annoying and clearly perceived artifact in the speech stream.

As the main objective of this thesis is to present a new solution for the frame erasure problem, a more detailed view of the impairments in the IP transmission media and their effects on the degradation of the speech quality will be helpful to fully appreciate the impact of using such concealment systems in repairing the PCM stream at the receiver.

5.2 IP technology Expectations and Drawbacks

Circuit switching and packet switching are the two main technologies implemented on a large scale in computer and telecommunication networks. Circuit switching, as in the case for the conventional telephone systems, requires a *physical* path to be established

between the source and the destination of the communication. It hence provides a guaranteed Quality of Service (QoS) and easier network provisioning for the system managers. On contrary, IP networks, which were originally implemented for non-real time data communication, are implemented as packet switching-based networks. Packet switching networks are connection-less networks. Data packets are transferred through a Virtual Path (VP) between the end-points. This virtual path consists of different combinations of Virtual Circuits (VCs) and the delivery of messages to the dedicated destination depends on the routers between these virtual circuits [21], [51]. Hence, a specific Quality of Service (QoS) cannot be guaranteed. In practice, system providers try to maintain a prescribed quality to the subscribers through provisioning the system and troubleshooting the problems occurring in a manner that is referred to as "*best effort*" network management [15].

Although IP networks have this structure, it is still tempting to use them to integrate the voice over data communication. Investments have been dedicated to repair the impairments usually encountered in this type of communication networks.

5.2.1 Main Advantages of delivering voice over IP networks

Building the voice telephony system on top of the existing data communication streams opens new horizons for internet telephony that go beyond the simple speech conversation service offered by the conventional telephony systems. Some of these advantages are described below [36], [2], [44].

(1) Integrating Data, Voice and Fax:

This advantage is of special importance to the enterprises that now depend on IP networks, especially Local Area Networks (LANs), to transfer data file and faxes through the enterprise. Integrating voice over such services results in cost saving as only one network will be needed to support all the services inside the company.

(2) Speech grading:

The conventional networks support only one type of speech coders: PCM waveform coders at 64 kb/s. On the contrary, the flexibility of the IP network enables system providers to make use of new technology in improving speech quality by implementing more sophisticated coding systems, especially the MPEG systems. Cost efficiency, the delay encountered and the quality of speech in the presence of impairments are the major challenges that face the realization of such ideas, but the ability of IP system to adapt to new compression categories cannot be ignored.

(3) Cheaper voice calls:

In the IP network, which is originally a data communication network, speech packets will share the link with other ordinary data packets and thus no special link is dedicated to voice transmission. This makes it cheaper to support voice calls over the IP networks, while in the case of PSTN networks dedicating a link to the call and more expensive equipments makes the service more expensive.

(4) Single messaging:

The main advantage of integrating voice with normal data in IP systems extends to benefit not only enterprises but also the individuals. Almost every person has his or her own e-mail address, telephone number and perhaps cellular or pager number. Using a single terminal, the computer in this case, to access messages will be faster and wider than the PSTN, which can only offer the voice message service.

(5) Enable video Conferencing:

The fact that video can also be integrated on the IP network can push this advantage beyond the financial benefit to a more advanced means of communication of special importance to large continental enterprises. Teleconferencing can also play a major role for individual customers especially in long-distance communication along side or in place of Internet Telephony.

5.2.2 Challenges Facing Integrating VoIP

IP networks are connection-less networks. Traffic between two end-points through routers invariably produces some impairments that are of major concern in delivering real time, high quality voice streams. The main impairments associated with speech transmission over IP network are delay, jitter and packet loss [43], [44]. Other impairments continue to hold from the nature of speech transmission rather than the nature of technology deployed e.g. echo.

5.2.3 Delay

Considering the fact that real-time data is sensitive to delay, we can reveal the serious problem facing telephony and communication on an IP network originally designed for non-real time data. Many sources of delay emerge in the network structure. A cautious and strict delay budget must be calculated in order to maintain a certain level of service.

As the ITU-T recommendation G.114 [50] stated, a one-way delay of 0 to 150 ms between two end points in speech conversation is acceptable. Delay of 150 to 400 ms is permitted in special cases of long-distance calls, provided that the users of the service are informed of this partially tolerable delay. However, a one-way delay of more than 400 ms is not permitted as it degrades the quality of the speech severely.

Figure 5.2, reproduced from ITU-T G.114 [50], shows the effect of delay on the percentage of perception difficulty. These values show no significant difference between 45 ms and 300 ms delays based on the "percent difficulty" score. At 500 ms delay, the percent difficulty score is approximately the double (an increase from 7.3% to 15.8%). However, this value is still considerably small compared to the value in the case when no echo canceller is used. In this case, the score jumps to more than 60% even with the presence of echo- suppressor [50].

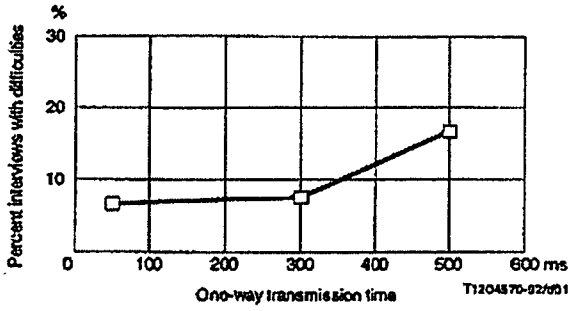


Figure 5.2: Effect of long one-way delay on the difficulty of conversation with an echo -canceller in the circuit [50]

In fact, delay is considered the principal problem in the IP networks. It enhances the effect of other impairments in the network. The negative impact of echo, for example, is exacerbated in the presence of long delays and its concealment process gets more complicated [61]. The ITU-T Recommendation G.131 [62] states that:

- "Echoes arriving after very short delays, about 20 ms, are generally imperceptible because they are masked by the physical and electrical side tone signal. Thus, no echo concealment is needed in this case".
- "Echo becomes increasingly annoying with increasing mouth-to-ear delay".

Thus, for an IP network, an inherently a large delay network, a very powerful echo canceller is essential. The ITU-T recommendation G.165 [63] presents a reference for the design of echo-cancellers. Commercial cancellers especially implemented for IP networks have been presented, some designs for echo- cancellers that follow the ITU-T standard can be found in [61]. The detailed explanation of the operation of these echo-cancellers is beyond the scope of the thesis but more information could be found in [61], [4].

5.2.3.1 Sources of Delay In IP Networks

The following figure, 5.3, reproduced from [44] shows the different sources of the delay in the IP network.

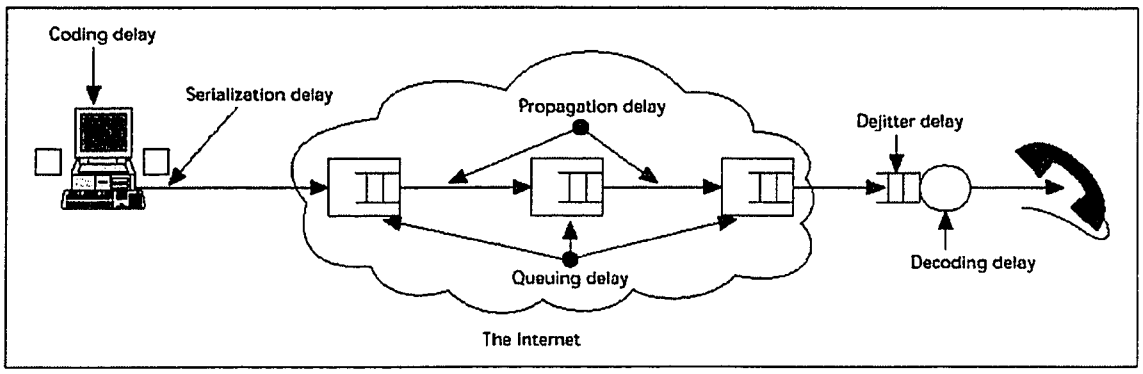


Figure 5.3: Major Sources of Delay in IP Networks [44]

(1) Algorithmic Delay:

The primary function of the coder is to convert an analog signal into a digital signal. It also performs compression to reduce the bandwidth requirement. In Chapter 2, it was noted that such compression comes on the expense of a delay. In other words, the compression ratio is almost always proportional to an added delay. This delay at the encoder depends on the coder used. Two main factors contribute to the total encoding delay. The first one is the *look-ahead* delay, which represents the waiting for future frame samples to be encountered in the encoding of the present frame, in order to exploit any correlation in successive voice frames. The other delay factor is the frame processing delay, which accounts for the time needed to apply the compression algorithm on the subject speech frame. Typical values for these two delays are shown in the following table for the most commonly used coders, taken after [44] with accordance with the data for the ITU-T recommendations of the subject coders [8], [9], [11].

Coding Standard	Frame-processing Delay (ms)	Look-ahead Delay (ms)	Total Encoding Delay (ms)	Typical Decoding Delay (ms)
G.711	Equals the packet size.	0	Equals the packet size.	Depends on the packet size.
G.729	10	5	15	7.5
G.723.1	30	7.5	37.5	18.75

Table 5.1: Encoding and decoding delays

It can be noticed from Table 5.1 that the PCM G.711 standard does not suffer from the codec delay, however the IP implementation of the G.711 requires the speech to be chunked in frames of sizes varying between 10-30 ms and so the processing delay of such wave form coder equals the time required to packetize the speech into the predetermined size frames. The other model-based coders add to their algorithm complexity an extra disadvantage of a look-ahead delay.

(2) Serialization Delay:

It is defined as "*the time required to place the subject packet on the transmitter*" [44]. The value of this delay depends mainly on the speed of the transmission line. It is almost negligible compared to the other delay components. It is typically 0.05 μ sec to place 1 byte over a 64 kb/s line, while in case of an OC-3 line, which runs at 155 Mb/s the delay is even more reduced to only 0.05 μ sec [44].

(3) Queuing Delay [27]:

This delay is one of the main factors that contribute to the loss of the speech packets in the transmission process. It also adds to the total delay experienced in the IP network. The queuing delay mainly depends on the structure of the routers and the nature of the packets that share the network with the speech packets. Longer Transmission Control Protocol (TCP) packets, which are non-real time packets, can significantly increase the processing time in the router. The statistical nature of the arrival process and the service algorithm can significantly alter the queuing delay from one router to another. The solutions for reducing the queuing delay will be presented with the proposed solutions for the packet loss.

(4) Propagation Delay:

It is defined as " *the time required by the signal to travel from one point to another*" [21]. This delay becomes significantly high where long distances are involved. Long distance fixed delay is apparent in the calls routed by the satellites, especially the geostationary (GEO) satellites, which has typical one-way propagation delay of 270 ms [64], and High Earth Orbit (HEO) satellites that are not widely used due to their impractical excessive propagation delay (although having excellent wide area coverage). In case of Low Earth Orbit Satellites, although the signal travels a significantly shorter distance due to its small coverage area (foot print), a problem of handing over causes the delay to turn variable and the problem of jitter starts to appear. This is discussed in more details when we come to explain jitter.

(5) Other Sources of Delay:

The implementation and configuration of the VoIP connection can cause specific delays to appear. In dialup networks, for example, some delays caused by the modems can cause the total delay to jump from 100 ms (if using digital lines) to 500 ms [28]. In the PCs, operating system inefficiencies and the sound cards delay have been a major concern for the packet voice system [35]. These types of delays add up significantly to the total delay budget.

5.2.4 Jitter

Jitter can be defined as " *the variance in the inter frame arrival times at the receiver*" [15]. It occurs due to variable queuing delays in the network. Sometimes, packets belonging to the same stream may take different paths. This is especially the case if VoIP is implemented on Low Earth Orbit (LEO) satellite. In the case of real time data, if any packet is inordinately delayed and does not arrive in time at the receiver, it will be considered lost; if this happens too often the quality of speech will be affected significantly.

The solution to this problem is what is referred to as " *the play out buffer* " or the jitter buffer [21]. The task of this buffer is to hold the voice packets that arrive at the receiver and try to rearrange them, according to their Real time Protocol (RTP) sequence number before sending them to the audio port [15], [2]. The jitter buffer adds to the total delay between the transmitter and the receiver and has to be very carefully tailored as long buffers are not permitted due to their large delay, and short buffers do not function well as they are confined to capture and rearrange only a few packets [51]. The typical value of the jitter buffer delay is 50 ms [4]. A new approach to implement intelligent adaptive de-jitter buffer is evolving. These intelligent buffers adjust automatically according to network variability [29]. Cisco, Hypercom and Nitrex , among others have already implemented these adaptive de-jitter buffers for commercial use [44].

5.2.5 Packet Loss

Packet loss is a common phenomenon in all packet switching networks, based on the best-effort model. Unlike in PSTN networks, depending on routers for setting the path of the data packets and buffering different IP packets from different sources along with speech is the main reason for packet loss. If there is no space in the buffer then congested routers drop the tail of their queues, and excessively delayed packets are considered "*Time- out* " packets. New packets are not acceptable in a full queue in a manner known as " Blocking", which plays a crucial role in determining the efficiency of the

transmission capacity and is measured by *Erlang* units [21]. As more people use the IP network and data packets start to overload the routers, the congestion problem occurs more frequently and thus the probability of packet loss increases [60].

Channel errors also contribute to packet loss. Almost all channels are prone to errors due to noise in the ambient medium. The ITU-T Recommendation G.113 [49] states that "Bit Error Rate (BER) less than 10^{-6} "have hardly no- effect on the voice quality". Studies in [58] showed that the packet loss increases up to 25% in case of severe channel noise. Other impairments like the delay and jitter can also result in packet loss as will be discussed later. Packet loss can severely damage the quality of speech. For example, for the G.729 it was shown that a packet loss of 3%, random erasure, brings the quality down by 0.5 points on the MOS scale, while reaching 5% degrades the MOS score to less than 3 points [43]. Packet loss of 5% is perceptible for most people if a model-based compression is used [51]. Despite the low quality, some commercial system are still allowing up to 10% packet loss in their systems [44]. The impact of the packet loss varies depending on two main factors:

(1) The statistics of the erasure: random or burst.

(2) The type of the codec implemented.

Burst erasures are more perceptible than random erasures. For coders that have memory, like the model coders, the effect of burst erasures on state error is more pronounced [2]. State errors extending to the next good frames received get larger in both magnitude and duration in the case of burst erasures.

A number of techniques have been used to address the problem of packet loss. Some of these techniques focus on reducing the probability of packet loss while others address the repairing of the damage caused by frame erasure.

The main objective of this thesis is to identify a new concealment technique for the lost packets when a PCM codec is used. Thus, a more detailed presentation of the

existing solutions and treatment for packet loss will be presented in the next section in order to set the stage up for the concealment method to be proposed.

(A) **Reducing the probability of packet loss**:

(1) Network upgrade:

Packet loss in the IP networks results mainly from insufficient link bandwidth and/or from router congestion. New technologies have been dedicated to produce an upgraded and more robust infrastructure for the IP network. Using ATM networks or Synchronous Optical Networks (SONET) can provide IP links with speed in the order of mega bits/s, while using Wave Division Multiplexing (WDM) can provide up to several terra bits/s. High-speed switch-based router technology is going through phenomenal improvement making it likely that a solution will soon be found for the congestion problem [27].

(2) Higher priority to speech frames:

This method gives priority to real-time data packet (recognized from the header, which contains an RTP header [15]). These real-time packets get the priority in the routing queue.

The priority level that is given to such packets prevents the possibility of serving non-real time data while real time packets are in the queue. However, it does not permit stopping the processing of the current non-real time packet when a real-time one arrives. It also does not give the priority to an older (with older time stamp) speech packet to be served instead of a newer one that precedes it in the queue, unless both packets are from the same source [56].

(3) Header compression:

This proposed method is aimed at reducing the queuing delay associated with the transmission of low bit rate speech packets, such as G723.1, where the frame size is 24 bytes for intervals of 30 ms [9]. Each frame is

conveyed over the IP network in the payload of the IP packet, which is usually called the IP datagram. It is very likely that each IP packet includes at least a header of 40 bytes in IPv4 [21].

This header consists of IP, User Datagram Protocol (UDP), and RTP protocol headers. This results in an inefficient utilization of network resources and longer time processing data at the routers (router delay). It also results in either the loss of other speech packets waiting for a relatively long time in the queue or the loss of the processed packet itself. Excessively delayed packets are more prone to get lost in an intermediate router (placed in the tail of the queue that is more frequently dropped). Even if those packets do not get lost in their way they arrive too late to be played at the receiver side [57]

In [57] it is claimed that the header size can be significantly reduced to only 2 bytes without the cyclic redundancy Check (CRC). The detailed explanation of the method of header compression is beyond the scope of this thesis. For more details, refer to [57], [56].

(4) Non real time packet segmentation:

Even if the voice packets have priority over non- real time data, they will be forced to wait until the transmission of the non-real time packets currently being served ends. Therefore, large non-real time packets or TCP packets should be segmented in this context.

Currently, TCP packets of more than 500 bytes are very common, and this results in a delay of more than 40 ms [56]. According to [59], business users of VoIP prefer delay below 10 ms in the access networks. The "Header- compression & segmentation" [57] method can meet this requirement by segmenting the TCP packets into smaller ones less than 100 bytes long. However, this method can turn risky if the TCP packets segmented are the ones carrying the signalling messages such as call setup

and tear down. In this case, the loss of these packets can actually increase the packet loss rate [64].

(5) Frame interleaving:

Interleaving voice frames across different packets can reduce the effect of packet loss. The procedure is based on segmenting speech frames and distributing them over many packets. This is done to ensure that previously consecutive frames are separated at the transmitter and thus the effect of the transient impairments in the channel (especially the burst noise) will be more distributed among speech segments and are thus less perceptible. The receiver on the other side will do the opposite operation of rearranging the speech frames in the appropriate order to be played back.

The main disadvantage of this method is the increase in delay, both at the transmitter and the receiver. The frames that were originally consecutive in time are spread over several packets. However if this method can be implemented, under the constraints of the budget delay, it will be an attractive method to combat burst losses, as it does not introduce any over-head in the network. [44]

(B) **Concealment of Lost Packets**:

In this category of packet-loss treatment the objective is to repair the damage resulting from the missing packets at the receiver end. This is done either by retrieving the lost packet itself by the error control methods or by reducing the perceptual effect of the gap in the speech stream by concealment methods.

(1) Forward Error Correction (FEC):

In FEC, redundant information from the same speech frame is contained in consecutive packets. In case of lost packets, this redundant information will enable the reconstruction of the lost speech segment. Several

techniques to carry this redundant information over the RTP and IP protocol have been presented [66]. The detailed presentation of the algorithms of such techniques is beyond the scope of the thesis, but in general this method adds a lot of overhead to the transmission and requires more bandwidth. It also puts extra load on the routers and may, in case of heavy traffic, contribute to congestion of the network [52].

(2) Silence substitution:

In this method, the content of the arriving packets is played back to reconstruct the speech. If a packet is lost in the network, a silence period, i.e. zero filled-frame, is played back instead of the lost one. Some commercial systems adopt this concealment technique for its simplicity, e.g. the Internet M bone [44]. This simple technique does not require any memory or any mathematical treatment of the signal.

Experiments showed that silence substitution caused "Voice Clipping" [60], which deteriorated the quality of voice significantly. This effect is more magnified if the loss period is large. Typically, silence substitution should not be used for packet sizes larger than 16 ms or loss percentages exceeding 1% [44].

(3) Noise substitution:

In this philosophy, the service provider camouflages the signal in the lost packets using another signal that has the same statistical properties. These statistical properties are extracted from the last packet received.

This approach depends on the fact that PCM is a waveform coding method and thus mimicking the shape of the lost signal will produce similar sound effects, and the listener will not notice the difference. This is indeed the case when the only sound perceived is the background noise. However, in case of voiced speech periods this method results in relatively poor

performance. Still, it is a better method compared to the silence substitution.

(4) Playing The Last Good Packet

This method depends on the assumption that for short packet periods (10 ms for 80 samples packet size), there will not be a large amount of new information to be delivered to the listener and that no severe change is encountered in the transition from the last good frame to the lost one.

This assumption holds for low percentage, widely spaced (not bursty) losses. This method has proved to be the most efficient one due to the fact that in the frequency domain, it is the only one that gives similar values for the speaker's vocal tract properties. Studies in [67] show that in the presence of echo canceller the resulting concealed signal can be severely deteriorated. Eventually the application of the noise substitution concealment methods (like the one proposed in [67]) can lead to better results than packet repetition.

(5) Interpolation Between Last and Next Good Packets:

In this method a more sophisticated implementation is proposed by a mathematical interpolation between the good packets surrounding the lost one. It, hence, demands a delay of one packet to wait for the next packet. If the next packet received is good a further computational delay is added before playing the concealed packet.

(6) Standard concealment techniques:

In this category of concealment techniques, the lost frame is replaced by a synthesized or concealed frame that is developed through the application of standardized concealment algorithms. These sophisticated algorithms were developed as the contribution of research groups and accepted (i.e.

recommended) by the international communication committees in the International Telecommunication Union (ITU) and the American National Standards Institute (ANSI). For PCM coding these standards are namely, the G.711 Annex A of (ITU-T) and T1-521-1999 Annex A and T1-521-2000 Annex B of the ANSI. These standards differ from the commercial methods in that they are more complicated and well developed as analysis and synthesis systems. Therefore, they give much better speech quality for the concealed frames.

In the following two sections, we will introduce these standards in more details. Giving the highest performance possible among all the concealment technique, they have been the natural choice to compare with for the concealment system that we propose. It will be shown in Chapter 6 and Chapter 7 that the new concealment algorithm is actually superior in quality compared to these two standards.

5.3 The ITU-T G.711 Annex A Standard [12]

This concealment algorithm is also referred to as the ANSI concealment standard Annex A: T1-521-1999. It is in fact a low complexity concealment algorithm that does not require more than 0.5 MIPS for packets of size 10 ms. Its principal idea is based on Reverse Order Replicated Pitch periods (RORPP). This technique will be discussed in more details in the next subsections. It is the basic standard we use for evaluating the quality of our new proposed algorithm that is why it is discussed in more details while the other standard algorithm of the ANSI (ANSI Annex B T1-521-2000) is only presented briefly.

To add this concealment algorithm to a system that currently does not conceal losses, changes are required at the receiver side only. It is designed to work with the conventional sampling rate of 8 kHz and frame sizes of 10 ms, but with slight modifications other sampling rates and different frame sizes can be accommodated.

5.3.1 Algorithm Description: [12], [48]

Figure 5.3 shows a graphical example of how the algorithm works with a 20-ms (i.e. two-frames) erasure on a voiced segment of a male speaker speech. The top waveform ("Input") shows the input. The location of the erasure is delimited at 10 ms intervals by 3 dotted vertical lines.

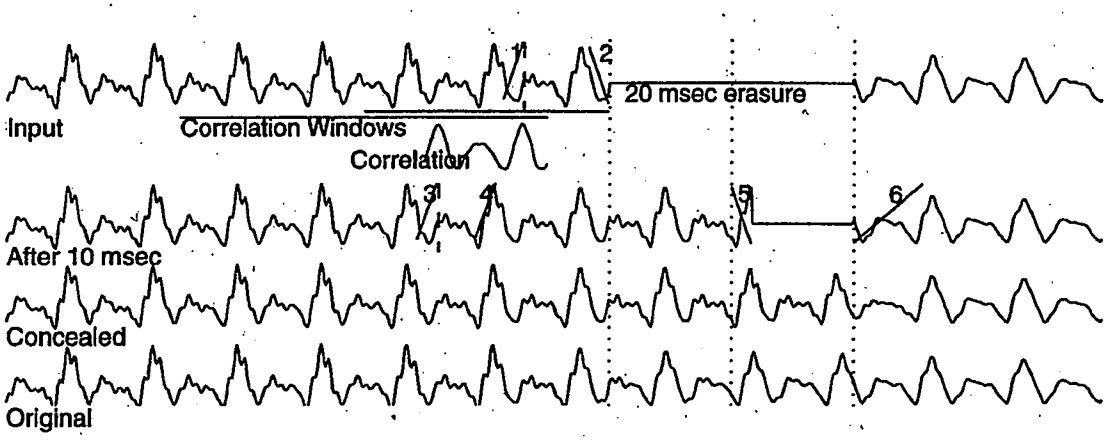


Figure 5.3: Reverse Order Replicated Pitch periods (RORPP) Packet Loss Concealment Algorithm [48]

The algorithm steps are:

5.3.1.1 Normal (No-Loss) Operation

During normal periods of good packets, the decoding process operates regularly and the receiver sends the output to the audio port. In order to provide the Packet Loss Concealment (PLC) algorithm with the required memory in case future packets would get lost, two operational changes are required at the receiver:

1. A copy of the decoded output is saved in a circular history buffer that has a length of 48.75 ms (390 samples). This circular buffer is actually used to calculate the current pitch- period and extract waveforms during an erasure.
2. The output is delayed by 3.75 ms (30 samples) before it is sent to the audio-port. This algorithmic delay is required to perform an Overlap Add (OLA) at the start of the lost packet and thus enables smooth transition between real and synthesized signals.

5.3.1.2 First Lost Packet

At the start of an erasure period, the circular history buffer is copied to a non-circular buffer, called the pitch buffer. This pitch buffer is used for the duration of the erasure. An additional copy of the most recently received ¼ pitch period of correct speech, under the ramp labeled "2" in Figure 5.3, is saved in what is called the last_quarter buffer to help in the formation of synthesized speech in case the erasure lasts for more than 10 ms.

5.3.1.3 Pitch Detection

The normalized cross-correlation technique of pitch detection is applied on the most recent 20 ms of speech in the history buffer with the previous speech. The pitch is estimated by finding the peaks at intervals from 5 (40 samples) to 15 ms (120 samples), corresponding to frequencies between 66 Hz to 200 Hz. This pitch range follows the choice used in the ITU Rec. G.728. While G.728 uses a lower bound of 2.5 ms (20 samples), here it is increased to 40 samples so the same pitch is not repeated more than twice in a single 10 ms erasure period. To reduce the complexity of the pitch calculation, the search is divided into two stages, a coarse search followed by a fine search. The coarse search is formed on a 2:1 decimated signal, and the fine search is performed in the vicinity of the peak of the coarse search. Actually, the complexity can be even lowered by skipping the fine search but a corresponding degradation of quality is expected in this case.

On Figure 5.3, the horizontal lines labeled " Correlation windows" show the windows. The upper line corresponds to the most recent 20 ms of speech before erasure and is the reference signal. The lower line is the 20 ms correlation window that slides back at taps 40 to 120 samples. The graph labeled "Correlation" under the window represents the output of the correlation that has a peak shown by the dashed vertical line at end of ramp"1".

5.3.1.4 Synthetic Signal Generation for First 10 ms

For the first 10 ms erasure, the best results are obtained through simply repeating the last pitch period with no attenuation. Only the most recent 1.25 pitch periods of the pitch buffer are used in this case. The result of this operation is shown in the "After 10 ms" waveform. To ensure a smooth transition between the real and the synthesized signal and different pitch periods an OLA is performed with a triangular window on ¼ of the pitch periods between the last and the next to last pitch periods. This is performed as follows: ¼ pitch period of the signal starting at 1.25 pitch periods from the end of the pitch buffer is multiplied by an up- sloping ramp (window1 on the figure) and then added to a corresponding ¼ pitch period at the end of the same 1.25 pitch period that is multiplied by a down-sloping ramp (ramp2 on the figure). The result of the OLA replaces both the tail of the pitch buffer and the tail of the history buffer (area under ramp2). The receiver also replaces the original tail in the last good frame by the synthesized tail and sends it to the output port. This introduces the algorithmic delay of 30 samples corresponding to the maximum possible tail for a pitch of 120 samples.

The synthesized signal for the 10 ms during the erasure is generated by placing a pointer one pitch period back from the end of the pitch buffer, and copying the samples to the output. In the case where the pitch period is shorter than 10 ms, the last pitch period is repeated multiple times during the 10 ms erasure. The history buffer updates its contents with the resulting signal and so if the erasure progresses the buffer always has a smooth continuous signal in it.

5.3.1.5 Synthetic Signal Generation after 10 ms

If the next packet is also lost, the erasure period will be at least 20 ms. In this case, the repetition of the same pitch period no longer yields good synthetic signal. It actually results in unnatural beeps and harmonic artifacts. This is especially noticeable if the erasure occurs in regions of rapid changes, e.g. if it lands in an area of high power unvoiced speech or stops at the end of the words, in such cases playing more pitch periods increases the variation in the signal. To add another pitch period to the pitch

buffer the pointer is placed two pitch periods back from the start of the erasure. An OLA is performed on the ¼ pitch period before the new added pitch period (Area under ramp3) and the last-quarter buffer saved from the last OLA operation (area under ramp 2). The result replaces the tail of the pitch buffer. For the second lost 10 ms the pitch buffer is thus the region between the dashed vertical line and the first dotted vertical line in the "After 10 ms" waveform, with the exception that the last ¼ pitch period is the result of the above OLA.

When the number of pitch periods used in the pitch buffer increases, it is important that the transition among the synthesized segments be smooth. This is achieved through continuing the output of the existing pitch buffer for ¼ pitch period at the start of the second lost packets (area under ramp5 on the figure). We then update the pitch buffer, keeping the buffer pointer synchronized with the correct phase, and do an OLA with the output from the new pitch buffer (ramp 4).

5.3.1.6 Attenuation

Following the same concept implemented in the G.729 and G.728 concealment algorithms, long erasure periods necessitate some attenuation of the reproduced packets. As the number of lost packets increases, the synthesized signal is more likely to diverge from the real signal. If attenuation is not applied, strange artifacts could occur. For the first 10 ms of erasure the synthesized signal is not attenuated. At the start of the second 10 ms, the synthesized signal is attenuated linearly with a ramp at a rate of 20% per 10 ms. After 60 ms, the synthesized signal is zero.

5.3.1.7 First Good Packet after an Erasure

To ensure smooth transition between the first good packet and the synthetic signal in the previous lost frame or frames, the same technique of smoothing is implemented again. The synthesized speech from the pitch buffer extends beyond the end of the erasure. It is then mixed with the actual signal using an OLA. The length of the OLA depends both on

the pitch period and the length of the erasure. For a short erasure, a ¼ pitch period window is used. For erasures lasting more than 20 ms, a 4ms per 10 ms of erasure is added up to a maximum of the whole packet size (10 ms) to be mixed with the synthetic signal. For the case we study (20 ms erasure) this is indicated by window 6 on Figure 5.3, which has a length of (¼ pitch period +4 ms).

5.3.2 Complexity Estimation of The ITU-T G.711-A

The algorithm complexity is estimated to have a peak rate of approximately 0.5 DSP MIPS. The main source of the complexity load comes from the calculation of the autocorrelation. This calculation is done only at the beginning of the first lost frame and so the average complexity is much lower. The model used for the complexity estimation is actually the same one we followed in calculating the different mathematical operations developed in Chapter 3. It assumes one DSP cycle for the MAC operation and 2 cycles for the Compare and 10 cycles for the Divide or Square root operations.

5.4 The ANSI Annex B (T1-521- 2000) Standard

In this standard, the algorithm of the model-based coding is employed to provide concealment. Model-based coders have the ability to synthesize the lost signal in the erasure period due to their inherent memory that keeps the parameters of analyzed speech, namely the long term and short term excitations, and makes use of the vocal-tract model of the all-pole filter previously introduced in Chapter 3.

Noting that a speech signal can be assumed stationary for short periods, the synthesized signal is developed by exciting the all-pole model with the previous frame excitation and the initial conditions of the last frame samples. This is exactly the same concept implemented in model-based coders. The main difference is that the G.711 PCM codec

lacks the speech production model (the all-pole filter) and the appropriate excitation signal. Such excitation signal is always extracted in the case of model-based coders. Thus, this ANSI standard model implements an analysis/synthesis model of speech that extracts the LP coefficients as well as the long-term and the short-term excitations from the last good frames, and synthesizes the signal through these parameters. The following figure, reproduced from [54], shows the block diagram of the concealment algorithm.

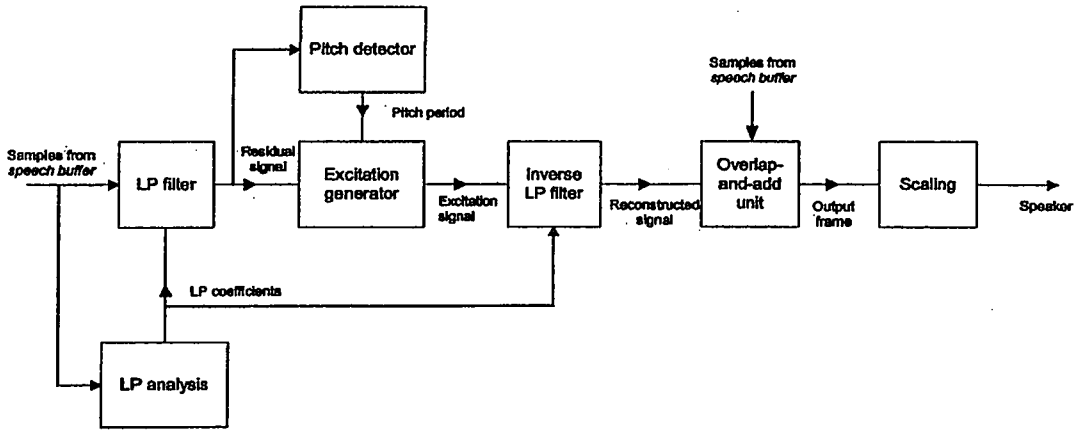


Figure 5.4 [54]**: Block Diagram of ANSI-B Algorithm for The First Lost Packet**

The algorithm uses a LP filter of order 20 and 120 samples to calculate the autocorrelation function. It introduces a 5 ms algorithmic delay that is used to ensure smooth transitions between the synthesized signal in the first lost frame and the last good 40 samples. It also develops another 40 samples of smoothing with the first good frame arriving after erasure.

This algorithm can accommodate packets of sizes 10, 20 or 30 ms. The peak complexity estimation depends on the size of the packet in the system, with the 10 ms packets having a peak complexity of 2.3 MIPS. More details on the algorithm can be found in [54].

5.5 Summary

This chapter presented many topics of particular interest to the IP telephony. It started with defining the IP network. The many benefits of VoIP over the regular telephone service were given for both enterprises and customers. Then we proceeded to answer the logical question: if VoIP has all these advantages over conventional PSTN, why do we still use the regular service? The answer was simply because of the many challenges that face the new technology. The different impairments that degrade the quality in a VoIP system were then discussed in detail. These are the high delay in the system, the jitter present in the IP network and the high probability of packet loss. The different solutions for the delay problem were introduced. It was also shown that the jitter problem can be effectively solved by a playout buffer that translates the jitter problem to a problem of added delay and packet loss. We then proceeded to the major problem of packet loss. This problem is the main impairment we address in this thesis. The proposed erasure concealment algorithm, which we shall present in the next two chapters, is actually implemented to reduce the impact of the lost packets in the PCM speech stream. The different available solutions for packet loss for the PCM coders (ITU-T G.711 standard PCM codec) were thus discussed in more details. We started with presenting the commercial systems that are widely used in the different IP systems. We then presented the two standard concealment algorithm of the ANSI, namely T1-521-1999 (which was actually the same as the ITU-T concealment tool for G.711 codec and known as G.711-A) and T1-521-2000, which we refer to as ANSI-Annex B. The ITU-T G.711-A standard was discussed in more detail as we compare our new algorithm concealment quality to the concealment quality that this algorithm provides.

6. Design History

6.1 Overview

Chapter 5 presented an overview of the problems that face real time voice transmission over IP networks. Among these problems we highlighted the packet loss problem. This type of impairment can be severe in best effort networks. Packet loss can, in some cases, reach a rate of 25% [58]. Several solutions for this problem have been discussed in the last chapter as well. Packet loss concealment procedures are aiming at camouflaging gaps in the speech stream. These concealment techniques are of special importance for the waveform codecs (PCM mainly) that do not have a built-in speech model that is capable of providing speech parameters. In model-based coders case, erasure is usually effectively concealed by the re-synthesis of missing speech segments using the previous frame excitation parameters and synthesis filter coefficients.

The concealment methods presented in the previous chapter are most effective for about 40-60 ms of missing speech. Erasure periods longer than 80 ms are generally muted as the speech signal loses its quasi-stationary characteristic [65]. These techniques vary from very simple ones that smooth the edges of the gaps to produce less annoying clicks to more advanced ones that replay the good packets (causing harmonic artifacts and beeps [65]). More sophisticated techniques are presented in the ANSI standards. The ANSI-T1-521-1999 (ANSI-Annex A) is actually the same as the ITU-T standard concealment tool of the PCM G.711 coded speech. The other ANSI standard ANSI-T1-521-2000 (Annex B) is implemented by Nortel as its prototype concealment tool [25]. The following figure, reproduced from [46], shows the MOS score of three different coders: The G.711 (with and without PLC) and the most popular model-based coders, the G.729 and the G.723-A. The packet loss concealment implemented for the PCM coder is the ANSI-standard B.

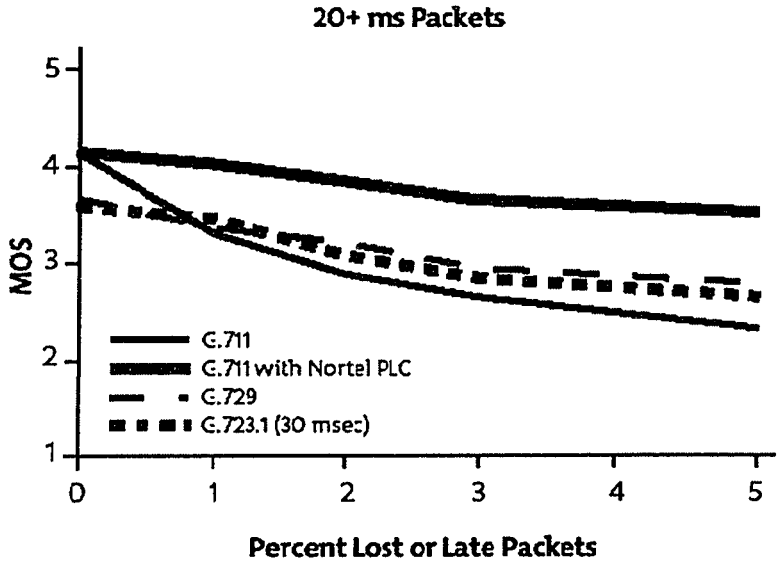


Figure 6.1: Comparison between MOS Score of Different Coders [46]

We can see from the figure that the addition of the PLC algorithm to the G.711 codec caused a radical change in its MOS score. Instead of having the worst behavior among the inspected codecs, the PLC unit boosted the G.711 codec performance. It gave the coder a fair chance to compete with the inherent concealment erasure capability of the model-based codecs.

The main contribution of the thesis, as we mentioned before, is to present a new erasure concealment algorithm for the ITU-T G.711 codecs. This chapter presents and explains the different phases of the proposed prediction system. The final form of the proposed new algorithm is presented in details in the next chapter. Here, we highlight only the main and most important changes we have added along the research. The design history presentation is meant to be a clear explanation of how the final form is developed.

6.2 Principle of Work

This new technique depends on the well-known LP prediction model. A sufficiently large order filter is used to account for the voice changes. The main equation of the prediction is:

$$\hat{S}[n] = \sum_{i=1}^{P} a_i \times S[n-i] + b[n] \qquad (6.1)$$

where b is the input excitation and is usually implemented as a white noise or periodic pulse component. a_i , $1 \le i \le P$ are the LP coefficients, $S[n]$ is the n th speech sample and $\hat{S}[n]$ is the nth predicted speech sample.

We can use the quasi-stationary nature of speech and use the same LP coefficients a_i from the last good speech frames. However, at the receiver side the input excitation of the present lost speech segment cannot be extracted (as it is of random nature). At that early stage of work we were assumed that the input excitation could be ignored. Equation 6.1 then reduces to:

$$\hat{S}[n] = \sum_{i=1}^{P} a_i \times S[n-i] \qquad (6.2)$$

6.3 Test Pattern

The test patterns that we implemented for the different development stages placed losses at repeated periods. The speech file was virtually divided into groups containing a number of packets. The number of consecutive good packets was set to the minimum memory length that we needed to compute the autocorrelation function and extract the history samples to be used in the synthesis of the concealed packet. In the early stages the number of good packets was two. The size of the pattern repetition period or chunk is determined by the loss duration. For single packet loss, the size is three and the pattern used is **0 0 1:** where "0" indicates no loss and "1" indicates the erasure. Thus, two good frames are used to predict a concealed one, which replaces the lost frame. The loss rate in

this case is 33%. There exist three possibilities to have a single predetermined lost packet in a group of three packets. These possibilities are:

1. **1 0 0:** first case
2. **0 1 0:** second case
3. **0 0 1:** third case

In the case of double loss (2 consecutive packets are lost) the test pattern is repeated every four packets and is of the form: **0 0 1 1**. In the early stages we relied more on the single packet concealment test. The results of the double packet loss are presented only for the last and most improved phase of the design. This is the phase that is modified in Chapter 7 to produce the final form. The random loss test was also postponed to be applied only on the final form in Chapter 7.

6.4 Test Files

The tests were performed on a group of males and females (two male speakers and two female speakers, 10 files per speaker). More details on the test files are presented in Chapter 7. The complete sets of results are presented in Appendix A. Each speaker set of results contains different tables representing the possible offset point of the test pattern (different cases).

6.5 The 1st Phase

In this phase we started with different LPC orders and settled on the implementation of order 50. The autocorrelation used for the LPC filter was implemented on the previous 160 samples (2 packets). The prediction process depends on the short-term prediction (Using Levinson Durbin algorithm) with a sufficiently long filter (chosen to be of order 50). We can generally justify the choice of this filter as follows:

The female pitch is usually shorter than 50 samples, which makes the 50-order LPC filter capable of covering at least one pitch period of the female speaker. For 80 samples lost packet it is possible to lose what is equivalent to two or three pitch periods of female speech during the erasure. If so, the quality of the female speech is severely affected. The LPC filter of this order is able to successfully account for the lost pitch periods. In case of male however, the situation is different. Long pitch periods that can, in some cases, extend to 160 samples make it possible that the lost frame lands on a small energy speech region and thus the degradation due to loss is minimized.

Other lower orders proved to produce a significantly lower MOS score. Table 6.1 shows the results for a male wave file and a female wave file with different LPC order values. These values are the MOS scores (estimated by P.862, the PESQ model of the ITU-T previously presented in Chapter 4) of a totally predicted file: it is a file that contains the predicted frames from the LPC function and does not contain any correct packet. Each frame is predicted using a correct speech samples. Such a file can be thought of as an ordered storage containing all the predicted packets and thus accounting for all possibilities of loss locations (for a single packet loss). It thus provides a faithful measure of the accuracy of the prediction. This was helpful at the first stage when we were defining the main design parameters (the filter order specifically). It was found out afterwards that this file helps in estimating the performance of the consecutive packet loss and the more similar this file is to the original one the better the burst erasure concealed packet prediction is.

File / LPC order	Male1 (first file)	Female1 (first file)
10	1.313	1.080
20	1.525	1.283
30	1.570	1.527
50	1.782	2.025
70	1.994	2.027
80	2.079	1.973

Table 6.1: The scores for different LPC orders

These results were expected from the above discussion on the significance of the order 50 for the AR model used. Small values are not sufficient especially with the absence of long-term predictor. However, Excessively large values can work in the opposite direction and deteriorate the quality of the estimated signal.

6.6 The 2nd Phase

The second phase aimed to improve the results of the first phase. It was noticed through examining the files that the absolute error (difference between correct and concealed frames) increased along the frame. This phase has added two items that continued to hold in the next phases.

1. Update the prediction process in the middle of the frame: The updating here involves redoing the whole function, starting with calculating new autocorrelation values (including the new predicted 40 samples i.e 200 samples autocorrelation window instead of 160) and then run the Levinson-Durbin algorithm to calculate new values.

2. Multiplying the LPC coefficients with a correction factor. This correction factor was estimated experimentally. It is referred to as λ^i : where i=1:50 and.

The value of λ is 256/253. The application of this factor on the LP coefficients helped in reducing the effect of the decay in the concealed speech segment produced from the model.

Figure 6.2, shows the process of the prediction:

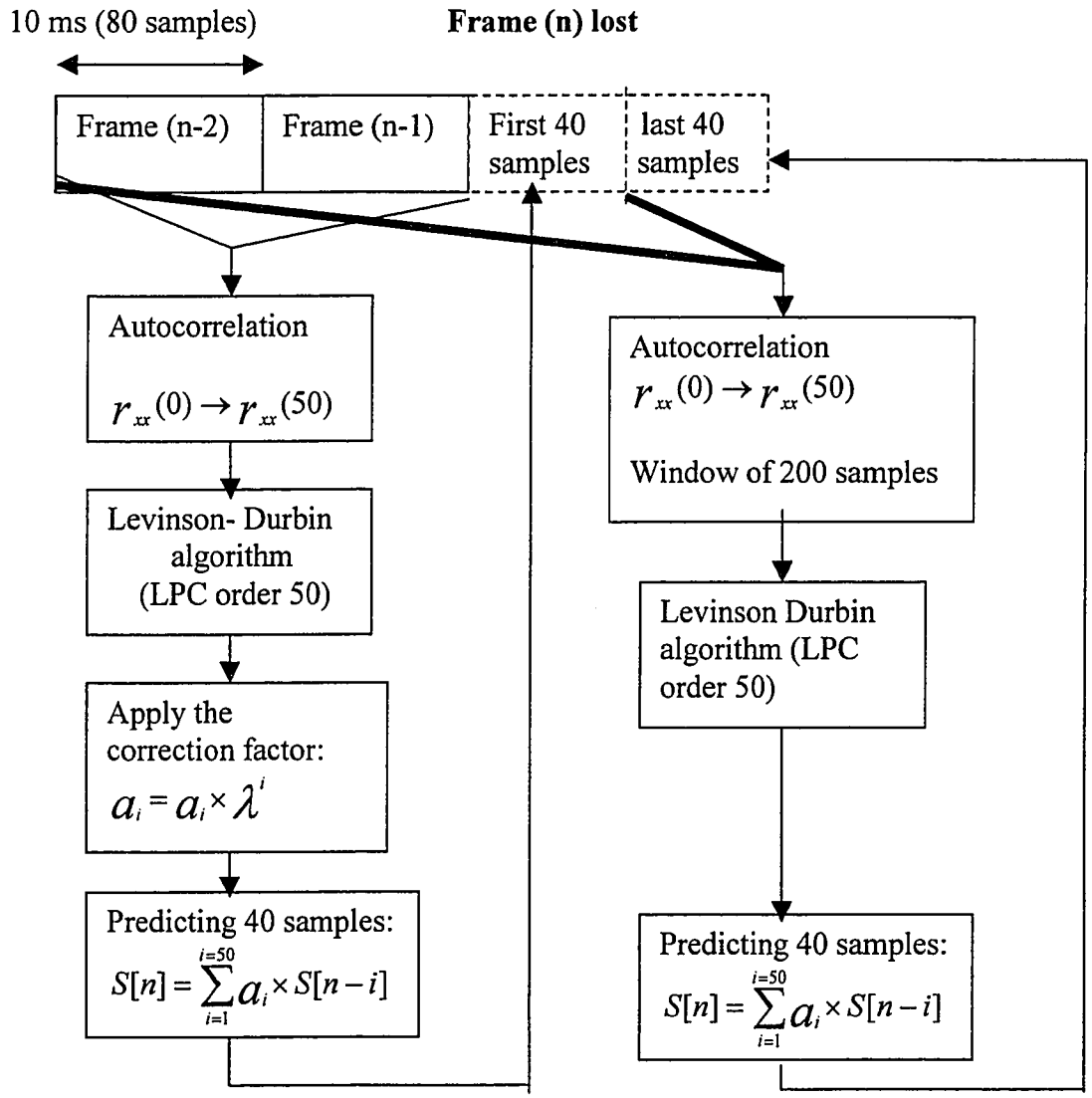


Figure 6.2: 2nd Phase Prediction Model

The improvement over the 1st phase is clear from the scores of the same test files used in Table 6.1, order 50:
Male: 1.951 (0.169 better than the previous case)
Female: 2.183. (0.158 better than the previous case)

The results of this phase were compared to both the packet-repetition technique and the ITU-T concealment standard G.711-A. The complete set of results is shown in Appendix A (tables: A-1). The 2nd phase model is referred to as "*Prediction*", the ITU-T G.711- A concealment tool is referred to as "*Standard1*" and the packet repetition is referred to as "*Repetition*".

The scores of the 2nd phase, which depend on the short-term prediction and does not use any pitch predictor, are very close to the results obtained by the ITU-T standard. The 2nd phase model provides a slightly smaller score in the case of male (less by 0.07 in case of 1st male and less by 0.17 in case of 2nd male). In case of female speakers, the 2nd phase prediction model shows a slightly better performance (better by 0.07 in case of 1st female and 0.17 in case of 2nd female). The commercial packet repetition technique produces a much lower performance than either the prediction (short-term prediction) or the standard ITU-T techniques.

For the long-term predictor ITU-T standard G.711-A, pitch repetition is used. The fact that the male pitch period is longer than the female pitch period makes us expect that fewer pitch periods are played in the duration of 10 ms (the duration of the lost packet), while in case of female speech more pitch periods are played in the 80 samples frame loss duration (most female pitch periods are below 50 samples). Thus, more artifact effects are expected when applying the standard ITU-T concealment technique to the female files. On the other hand, the new technique uses an LPC filter of order 50 and no-input excitation. We can expect that the 50th order filter will be capable of covering at least one pitch period in case of female speakers. The decaying effect, resulting from the fact that the model is a system with no input, was found to be less severe in this case, possibly because the previous fifty samples (i.e. initial conditions) always contain some energy

peaks. In case of male speakers, with pitch periods generally higher than 50 samples, the decaying effect is more severe, possibly because the previous samples may not contain any energy peaks.

This phase is presented mainly as documentation on the efficiency of the idea of the short-term predictor. It showed the validity and importance of accounting for the fast changes in the signal (the short- term component). It was not considered a final form of the system to be presented. Through studying the ITU-T concealment standard documentation [12] it was clear that the performance of the concealment tool depends also on another factor besides reproducing a good estimation of the missed signal. This factor is the smooth transition between the correct speech segment to the concealed one and the next transition after the erasure period.

6.7 The 3rd Phase

This phase is mainly concerned with improving the score of the resulting wave files without any modification to the prediction function. The best way to do this is through applying cross-fading between the predicted packet and the correct packet next to it. This cross fading is not applied on the transition between the last correct packet and the predicted one. We assume that this transition is smooth due to the use of the previous samples in the prediction. The smoothing function is simply implemented through the following consecutive steps:

(1) Prediction of an extra ten samples at the end of the frame (predicting 90 samples instead of 80) this extra part referred to as *"Tail"* presents the expectation of the next 10 new values of the next frame if it were lost.

(2) A decreasing window (Triangular) is applied to the predicted samples and the inverse (Increasing Triangle) is applied to the corresponding actual values obtained from the correct next frame.

(3) Summation is performed between the predicted and the good samples and the output of the summation replaces the first 10 correct samples at the beginning of the new correct frame.

The following figure shows the block diagram of the prediction process of this phase:

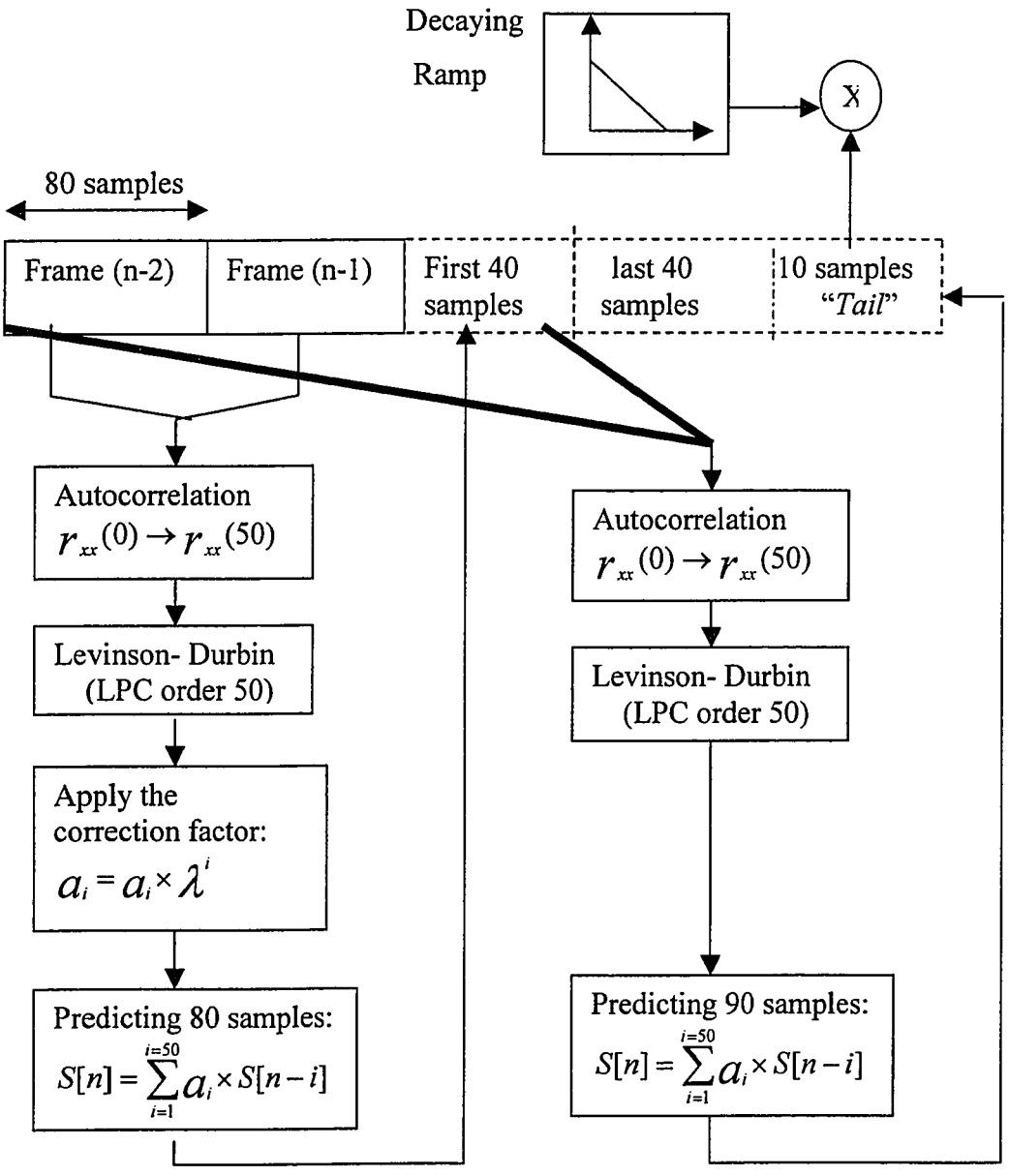


Figure 6.3: The 3rd Phase Prediction Model

Significant improvement has been achieved through applying the edge cross-fading between the predicted packet and next good packet. For male files the improvement reached 0.4 in some files. On average, improvement of more than 0.2 was achieved for the male files and improvement of more than 0.1 was achieved for female files. The results of this phase proved to be better than the scores obtained by the ITU-T standard algorithm G.711-A. The problem of lower score male files was eliminated and this showed that the step of smoothing is very efficient and thus it is kept along the next phases and is actually applied to the final form of the algorithm.

6.8 The 4th Phase

This phase was developed to improve the performance of the short-term prediction. This has been attempted by increasing the LPC order used to produce the 2nd half of the lost frame that seemed to still cause a lot of degradation while the 1st half absolute error seemed negligible. Various LPC orders have been tried so that the optimal one could be recognized. This selection showed that order 90 for the 2nd half of the lost frame is most probably the best choice for the filter implementation. Using a 90-order synthesis filter permits keeping the last good 50 samples (from the previous packet and used to produce the first half of the concealed frame) with the addition to the 40 new samples produced from the 50 LPC order filter. In this phase as well we implement the correction factor λ in the concealment of the 1st half frame (50 LPC order) while the second half prediction model is not using that factor. The complete set of results for the different test files are shown in Appendix A (A-3).

This phase showed some improvement in case of male files (0.06 on the MOS scale) while no significant improvement has been obtained in case of female files. The main reason we have implemented this phase was to reduce the significant bad performance of the system in case of burst erasure. The prediction system presented in the 3rd phase system showed to work well for single packet loss. For double packet loss however the scores were very bad and even worse by 0.1 than the packet repetition test for the male

files. By implementing this higher order prediction we were able to get reasonable scores for the double packet loss, especially when we used a LPC filter of order 130 for the second frame.

However due to the high complexity associated with implementing such excessively high order synthesis filter, we decided to give up the idea of improving the score through using a high order LPC filter. We started considering other approaches.

6.9 The 5th Phase

This phase is concerned with optimizing both the autocorrelation window size used and the gain we apply on the resulting concealed samples. In the previous phases we had used an autocorrelation window of size 160 samples (2 packets). The gain was applied on the concealed samples indirectly by applying the empirical factor λ^i on the LP coefficients used to synthesize those samples.

However, we found out that using an autocorrelation window of size 240 samples (3 packets) gives slightly better results. Also, applying a ramp gain on the resulting samples directly gave slightly better results on average compared to applying the exponential gain on the LP coefficients. In Tables (A-5) in Appendix A, a complete set of results on the comparison between autocorrelation window of size 160 and 240 is presented. This is of special importance because in the next phase and the final from of the algorithm we kept the size of the autocorrelation window to 240 samples. The step of updating the prediction in the middle of the file was also eliminated due to its high complexity. This is based on the principle that no new information can be extracted by the analysis of the concealed samples and that this updating operation is computationally exhaustive. The following figure shows the general block diagram of the 5th phase.

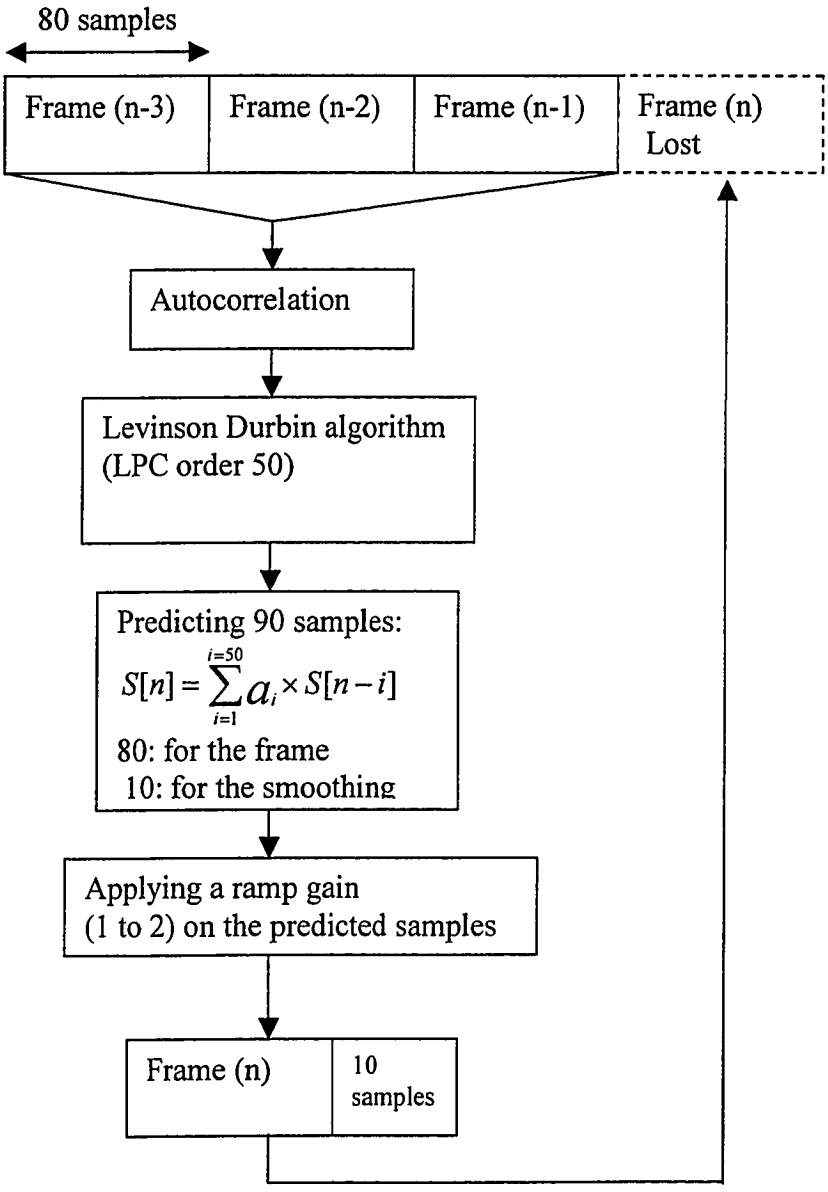


Figure 6.4: The 5th phase Prediction Mode

6.10 The 6th Phase

All the different attempts in the previous phases showed that the short term prediction we implemented was very effective in the single packet loss case. But the increasing decay in the concealed signal made it very difficult to keep good scores when the erasure lasted more than 10 ms. This decay is well expected considering that the prediction filter we implemented was a stable system with no input. The radical modification we made to the original prediction equation was to ignore the input excitation. We were obliged to do so as no good choice was available for this residual signal.

In this phase of algorithm development we reconsidered the choice of the residual excitation b. Recalling equations 3.3 and 3.4 the prediction error is expressed as:

$$r(n) = S(n) - \sum_{k=1}^{p} a_k S(n-k) \qquad (3.4)$$

The ideal case is when the LPC filter is capable of accounting for the whole correlation between the current sample and the past samples. In this case, the prediction error is the random excitation by which the speaker excites the vocal tract. However, if the LPC fails to extract the complete correlation between the successive samples, the residual signal turns to be coloured (has some correlation with the original speech signal).

In this phase we considered the use of a small percentage of the pitch-predicted signal resulting from the ITU-T G.711-A tool as the input excitation for the system. The residual signal added will then be coloured and having the same waveform as the pitch predicted signal and most probably the missing speech segment. This avoids the divergence of the prediction model output from the original signal waveform. The best results obtained were when we used a residual excitation equal to 0.01 of the pitch predicted signal. In this case we no longer need to multiply the LP coefficients by the exponential correction factor λ^i as the decay was already compensated for by the residual excitation.

This choice of input excitation gave the best scores among all the different phases for both the single packet loss and the double packet loss. However, it still gives lower score than the ITU-T standard tool (G.711-A) in case of double packet loss. The prediction model of this 6^{th} phase is modified in the next chapter to produce the final form of the prediction model. The results of this phase 6 for single and double packet loss are also presented in the next chapter. They are compared to the results of the final form of the algorithm and the results from the ITU-T G.711-A concealment standard.

6.11 Summary

In this chapter we presented a record of the major design stages we implemented along the research and led us to the final form, which we are going to present in the next chapter. We started with presenting the motivation for implementing an effective concealment algorithm for the PCM codec. It was shown that the PCM speech compression technique has high scores in the no loss conditions. This quality drops dramatically in situations of packet loss; however, with PLC algorithms the user satisfaction rating can be kept high. We then presented the main parameters in the test set-up, namely the test files and the loss patterns.

The main idea of the new concealment algorithm is to use the well-known Linear Prediction (LP) speech synthesis equation. At first, we were not able to find a proper value for the input excitation. Thus, we were obliged to ignore the input excitation and depend on the past samples only to predict the lost speech segment. This made the system suffer from a decay problem that increased along with the prediction. This resulted in the prediction of the second concealed packet (in case of burst erasure) dropping to lower scores than what the G.711-A concealment algorithm provides.

This decay problem was the major problem we tried to solve along the different phases. The different solutions that we tried were:

1. Multiplying the LP coefficients with a correction factor (larger than 1) and thus amplify the resulting samples.
2. Applying a ramp gain on the resulting speech samples.
3. Increasing the order of the prediction.

Actually all the above methods were successful to a point (especially with the first lost packet) but they failed to provide satisfactory results for the case of burst erasures of male files. The later have longer pitch values than the females and thus the filter in some cases may not have energy peaks in its initial conditions (i.e. 50 previous samples), and thus decays faster than in the case of female files.

We had to reconsider the impact of ignoring the input excitation b. This residual signal is known to be generally coloured and following the same waveform of the missed speech segment, i.e. the waveform of the pitch predicted replica of the signal. A small fraction of the pitch predicted replica (RORPP), extracted in the same manner as in the ITU-T G.711 Annex A, can be used to replace the input excitation and in the worst case will not make the system diverge from the original waveform.

The idea worked better than all the previous phases in case of single packet loss. In the case of the double packet loss it provided the best results among the different previously implemented phases but still gave lower average scores than the G.711-A standard tool. The final form of the algorithm deals with this problem. This is presented in more details in the next chapter.

7. New Linear Prediction Concealment Algorithm

7.1 Overview

The G.711 PCM coder is one of the most prominent candidates to be the standard of the VoIP due to its simplicity, robustness and high quality. The quality of the G.711 PCM coder is in fact the reference quality in the speech communication world. It is referred to as the *" Toll Quality"* and estimated to have a MOS score of 4.1 to 4.5. All the other coders usually have less quality, except for some complicated perceptual coders that are not practical for the real time VoIP application, as we previously indicated in Chapter 2. Figure 7.1 shows the MOS score of different codecs in the case of no packet loss.

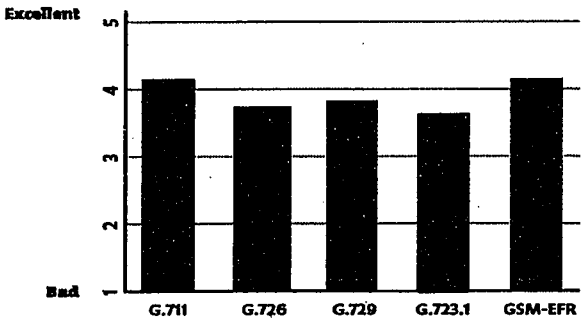


Figure 7.1: Results of Subjective Survey Assessing Codec Quality with Clean input Speech [65]

In fact, the Voice over Digital Subscriber Line (VoDSL) study in [45] showed that the G.711 PCM coder has superior performance over all other codecs including the G.726 ITU-T waveform standard [22]. It showed that in case of using G.711 for speech compression there is no codec impairments. However, if the G.726 standard is used instead, the R rating (previously mentioned in chapter 4) drops 7 points. The G.711 is also robust to bit errors and frame loss [55], as it does not exhibit "state error" [46]. State error is a very annoying problem with coders that depend on past values of speech samples, like the G.729 and other model-based systems. For model-based codecs, the loss of the n^{th} frame degrades the speech quality not only during the loss period but also for

several subsequent frames due to the system memory. The G.711 is also robust to Tandem coding errors [55]. This has been explained in Chapter 1.

Even though the G.711 system has many advantages it lacks the ability to conceal the loss of speech frames. Such ability to conceal loss is inherent in the model-based systems that model the speaker's speech during the coding operation. Many proposals have been implemented as concealment techniques to camouflage or hide the effect of chopping or clipping of the lost frames and help to reduce the large drop of quality that could occur in case of packet loss. In Chapter 5 we have presented several commercial concealment systems as well as the standard techniques of the ITU-T G.711 Annex A, which is based on the Reverse Order Replicated Pitch Periods (RORPP) as principle of work and the Linear Prediction LP-based ANSI concealment standard Annex B "*T1-521-2000*".

This chapter is dedicated to the main contribution of the thesis: a new improved concealment algorithm for the ITU-T G.711 PCM standard. It is aimed at explaining the final form of the proposed prediction system. The design steps or the different phases that led to this concealment algorithm have been presented in the previous chapter, Chapter 6 " The Design History". We separated the presentation of these phases from the explanation of the final form to provide a complete focus on the genuine work we finally realized. The proposed concealment algorithm is derived from well-known prediction steps and will be shown to be simple to use and provide high quality.

The performance of the proposed algorithm will be compared to the packet- repetition method, the standard concealment tool of ITU-T (the G.711 Annex A) and the ANSI standard (T1-521-2000) Annex B. It has been difficult to run the source codes provided with this last standard due to some unstable concealed packets. The only way we were able to get stable outputs from this tool is by resetting the loss indicator of the concealed packets that exhibited this unstable behavior, and playing the corresponding correct packets instead. The same modified loss pattern was then applied to both the new concealment method and the ITU-T standard in order to apply the same loss locations on all the resulting files of the three concealment tools under inspection. The results obtained

from this ANSI standard tool are thus considered indicative only. We present them just to indicate that the proposed concealment method can produce better results than the existing two concealment standards.

7.2 Principle of Work

This new LP-based concealment technique is based on the prediction with a sufficiently large order filter that is capable of accounting for the voice changes.
Recall the main equation of the prediction as given in Chapter 3:

$$\hat{S}[n] = \sum_{i=1}^{P} a_i \times S[n-i] \quad + \; b[n] \qquad (7.1)$$

In the case of lost packets the input excitation b[n] is very difficult to estimate. In model-based coders *b* is usually a multi-pulse or a white noise component. Modifications to the above equation have been made based on what is available at the receiver side. In the modified model, where the input excitation b is unknown, the equation 7.1 becomes:

$$\hat{S}[n] = \sum_{i=1}^{P} a_i \times S[n-i] \qquad (7.2)$$

Experiments with this initial form of the new proposed concealment method, presented in Chapter 6 " Design History", showed that the modified model performs well in the case of 1st lost packet, while longer duration loss showed weak performance. That is expected from a model with no-inputs (because there will be a large drop over time in the concealed signal envelop).

This was the initial principle of work that we began with. It was later noted that this excitation input b can be replaced with a small fraction of the long-term prediction of the lost frame. Here, the long- term prediction of the lost frame refers to an (RORPP) Reverse Order Pitch Period Replication of the lost frame, estimated by placing the pointer one pitch period to the past of the start of the loss location. This is done in a manner

similar to the pitch-repition based concealment algorithm implemented in the ITU-T standard G.711-Annex A [12], [48].

Thus, using a small percentage of the long-term excitation we can express the input excitation to the model we get:

$$b[n] = \hat{S}[n] \times G \qquad (7.3)$$

G =0.01 was found to give the best results.

Equation 7.1 can then be expressed as:

$$S_1[n] = \left(\sum_{i=1}^{p} a_i \times S[n-i] \right) + \left(\hat{S}[n] \times G \right) \qquad (7.4)$$

The calculation of the long-term prediction is very negligible in complexity because the autocorrelation function has to be calculated anyway in this LP-based algorithm for the estimation of the LP coefficients set. Thus, the complexity needed to estimate the pitch-period is limited to the comparison process (search between peaks) as we will present later. However, the inclusion of 1% of the long-term prediction as an input in equation 7.3 and 7.4 helped in reducing the damaging effect of the increasing decay of the resulting signal. This has been previously discussed in Chapter 6.

Next, we propose to modify the algorithm by using a weighted summation of the short-term prediction $\left(\sum_{i=1}^{P} a_i \times S[n-i] + G \times \hat{S}[n] \right)$ and the long-term prediction $\hat{S}[n]$ to provide a better approximation to the original signal. Thus the final form of the prediction algorithm becomes:

$$S_1[n] = \left(\sum_{i=1}^{p} a_i \times S_1[n-i] \right) + \hat{S}[n] \times G \qquad (7.5.A)$$

$$S[n] = S_1[n] \times \alpha + \hat{S}[n] \times \beta \qquad (7.5.B) \qquad (7.5)$$

where α and β are summation weights that should have the summation of unity . The best results were obtained with $\alpha = 0.7$ and $\beta = 0.3$. S_1 [n] denotes the short-term (i.e.

LPC) prediction. This part of the equation is actually the last phase in the development process of the algorithm in Chapter 6. The final form that we propose is thus the addition of this phase to the long- term prediction $\hat{S}[n]$.

The signal $\hat{S}[n]$ is actually obtained by applying a gain on the pitch-estimated signal, as in the implementation of the RORPP (pitch repetition) developed in the ITU-T standard algorithm. This gain is fixed and equal to 1 for the first lost packet. For the consecutive lost packets however this value is decaying in a linear way by 20% per 10 ms (starting at the original value 1). This decay will cope with the natural decay in the LP synthesis filter output. Although the LP synthesis filter decay was considered harmful earlier because of its fast rate, a natural smooth decay is preferred when the erasure period lasts more than 10 ms [12], [48]. There is no contradiction present in this point as we try to add the long prediction contribution to compensate for the fast rate decay resulting form applying the model of Equation 7.2, which has no input excitation. We also need a smooth slow decay in the output on the long run to avoid or minimize the artifacts that could result from the long erasure periods.

7.3 Proposed Error Concealment Algorithm For G.711

The new Linear Prediction-based concealment is a receiver- based concealment algorithm. We assume that the audio input signal is sampled at 8kHz and each packet contains 10 ms (80 samples) of audio. The algorithm uses the well- known speech production model, previously discussed in chapters 2 and 3, and depends on the principle of work described by equation 7.5. The basic operation of the algorithm is to reconstruct the speech model and its excitation parameters during the erasure period based on the information extracted from the previously correct speech sequence received.
The algorithm depends on the following set of parameters, variables and buffers:

- *History buffer*: The most recent 30 ms of good speech stream is stored to be used in the prediction of the lost frame.

- *Overlap buffer*: This buffer contains a 1- ms (10 samples) extension of the generated concealed signal multiplied by a down slopping ramp, to be added to the 10 samples start of the next good frame multiplied by an uprising ramp. This addition is necessary to ensure smooth transition between the concealed packets and the correct packet after erasure. It is worth noting that because the good packets before erasure were used (directly) in the speech model to predict the concealed packet with linear prediction, a smooth transition between the last good speech samples and the next predicted samples is realized directly and there is no need to add another overlap unit at the end of the last good packet as in the ITU-T G.711-A standard concealment algorithm [12], which introduces 3.5 ms algorithmic delay, or as in the ANSI standard Annex B that adds 5 ms delay [54].
- *LP coefficients*: The LP coefficients are used to form the poles of the all-pole synthesis speech filter. We have pointed out in the last chapter that the order of the prediction filter implemented is optimized to be 50. These coefficients are calculated at the first packet of a lost speech segment and stored for the use with consecutive lost packets.

- *Pitch period buffer*: The pitch period estimated segment for the first packet of erasure is stored and used with consecutive lost packets. We can follow the ITU-T standard and use more than one pitch period if the erasure lasts for more than 10 ms. The 30 ms history buffer can provide at least two pitch periods for maximum pitch values of 120 samples.

- *G*ain: The contribution of the current long prediction sample in the formation of the excitation signal b in equation 7.1. This value is fixed and equal to 0.01.

7.4 Detailed Algorithm Implementation

7.4.1 Normal (No-Loss) Periods

During the normal operation of the PCM decoder (period of no loss), the receiver decodes the received packets and sends the output to the audio port. Meanwhile, in order to support the concealment algorithm, a copy of the decoded output is saved in a history buffer that is 30 ms (240 samples) long. The history buffer is used to calculate the autocorrelation function, estimate both the pitch and the LP coefficients, extract the long term excitation and provide the past samples $S[n-i]$ $1 \le i \le p$ where p is the order of the prediction filter (in equation 7.1).

7.4.2 First Lost Packet

A lost speech segment contains at least one lost packet but may contain more. The majority of the computational load is in the first 10 ms of erasure (the 1st lost frame). Figure 7.2 shows a block diagram of the principal blocks of the algorithm.

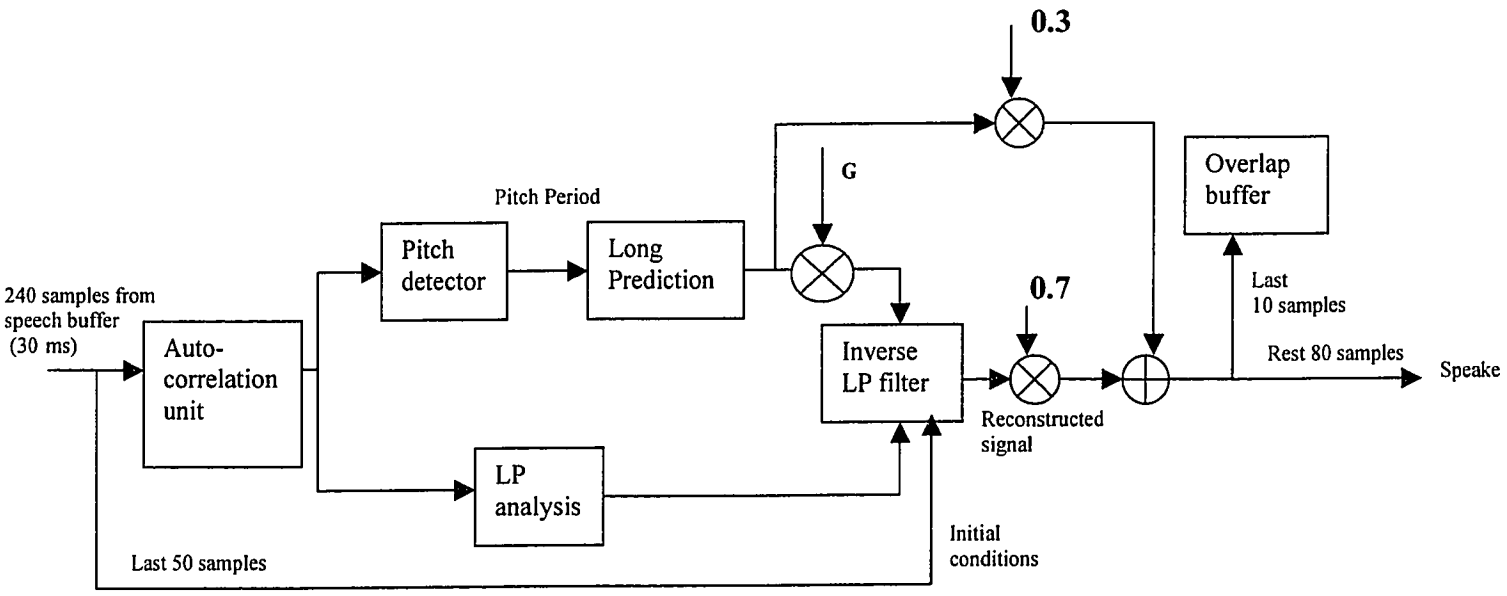


Figure 7.2 Block Diagram of the New Algorithm for the First Lost Packet

At the start of the erasure period, the autocorrelation function of the last 30 ms (240 samples) of speech is calculated. The output of the function is used by both the pitch detection unit and the LP analysis unit. The Pitch detection unit estimates the current value of the pitch by searching among the peaks of the autocorrelation coefficients and returns the most recent pitch period, which will be used twice. It is first multiplied by the gain G, which is equal to 0.01. This re-scaled pitch period is used as the short-term excitation of the speech production model. More detailed explanations on this point are presented when discussing the synthesis LP filter. The pitch period component extracted is also multiplied by a factor of 0.3 and added to the resulting output of the synthesis LP filter, which should be multiplied by a factor of 0.7.

The LP coefficients are calculated in the LP-analysis block that implements the Levinson- Durbin algorithm of LP estimation, which has been previously discussed in detail in Chapter 3. The LP prediction order was chosen to be 50, as we mentioned before. These 50 coefficients are used as the poles of the LP-synthesis filter which is the model of the speech production. The output-concealed samples are calculated by applying equation 7.5.

7.4.2.1 Autocorrelation Calculation

The autocorrelation block is used to calculate the autocorrelation function of the most recent 30 ms of correctly received speech, to be used for pitch prediction and LP analysis. The autocorrelation calculation can be done either in the time-domain or in the frequency domain as it has been discussed in Chapter 3. It has been pointed out that the autocorrelation calculation, *from the complexity point of view*, is more efficiently performed in the frequency domain by using the FFT analysis particularly for real signals as explained in Chapter 3. For the time domain calculation of the 240 autocorrelation the number of MACs required is 28920 MACs while in the frequency domain the corresponding complexity for the calculation of the same number of coefficients is 9728 MACs only.

7.4.2.2 Pitch Detection

The pitch detection block has the task of copying the most recent pitch period samples to be used in the formation of the input excitation. This is done by first estimating the pitch period by finding the peak of the autocorrelation function performed on the speech samples in the speech buffer. This search process is in fact a comparison process among the values of the autocorrelation coefficients. According to the complexity calculation model followed by the two standard concealment techniques, the ITU-T standard and the ANSI standard, a compare operation is estimated by 2 cycles or what is equivalent to twice the complexity of a Multiply Accumulate (MAC) operation. The complexity in this case is still not high, but based on the G.711 Annex A, lower complexity can be achieved by considering a reasonable range of pitch values typically between 40 and 120 samples and performing the search in two stages, the coarse search at taps of 5 samples followed by a fine search in the vicinity of the narrower range obtained from the coarse search. This idea works well in a standard that depends to a large extent on accurate estimation of the pitch period so it was logical to follow the same procedure in pitch detection in our algorithm to minimize the overall complexity of the algorithm.

7.4.2.3 LP Analysis and Synthesis

The LP analysis block computes the LP coefficients of the previous speech samples from the last good packet before erasure. An LP order of 50 is used. This order has been optimized in the first phase of the new algorithm design as we have explained in the previous chapter. The LP coefficients are passed to the synthesis LP filter to be used in the synthesis of the concealed speech samples of the first lost packet. They are also stored for the use with consecutive lost packets.

The synthesis LP filter models the speech production at the current period of erasure. It is an all-pole filter working on the same principle of model-based coders and follows equation set 7.5, which is shown here again:

$$S_1[n] = \left(\sum_{i=1}^{p} a_i \times S_1[n-i] \right) + \hat{S}[n] \times G \qquad (7.5.A)$$

$$S[n] = S_1[n] \times \alpha + \hat{S}[n] \times \beta \qquad (7.5.B)$$

7.4.2.4 Overlap Add Unit

The previously discussed synthesis model is used to produce the concealed packets for the lost frames. In fact, we have modified that model to produce 90 samples per lost frame instead of 80 samples. The last 10 samples are the predicted values of the packet next to the lost packet. If the next packet is lost then these values are played as the concealed samples of that lost frame. But if the next packet is not lost than these samples are multiplied by a decaying ramp and added to the corresponding first 10 samples in the new correct speech sequence that are to be multiplied by an uprising ramp. The output of the addition is played instead of the first 10 good samples after erasure. This cross fading process guarantees a smooth transition from the concealed speech segment to the good packets condition.

7.4.3 Consecutive Lost Packets

If the erasure lasts more than 10 ms (one packet period) no new parameters are calculated. We re-use the previously obtained parameters used for the first lost packet concealment with the slight modification of changing the long-term estimated period samples, as in ITU-T G711-A. These samples were previously multiplied by a fixed gain that was equal to 1 in the first frame case. In the case of consecutive lost packets, they are multiplied by a decaying ramp starting at the initial value 1 and decaying at a rate of 0.2 per 10 ms. This ramp multiplication introduces smooth decay increasing along the loss period. Eventually, at 60 ms of continuous erasure the long-term prediction is zeros and we return to the no input short-term prediction, decaying model we implemented in Equation 7.2.

7.4.4 First Good Packet after Erasure

As it has been mentioned in the overlap-add block we multiply the first ten samples of the first good frame after erasure by an up-sloping ramp and then add it to the tail (the ten predicted samples at the end of the 90 samples produced from the synthesis filter) to form the resulting signal that is sent to the audio port instead of the original 10 samples.

7.5. Delay of the Proposed Algorithm

The algorithm does not produce any algorithmic delay as the concealed samples are calculated from the past good speech stream. The only source of delay is the computational delay, which can be reduced by executing the PLC algorithm after every good packet, before it is known if the next packet is lost or not. If the next packet is lost the synthetic signal will be ready immediately to be sent to the audio ports. If the next packet is not lost the synthetic signal is just dropped.

7.6. Complexity Calculation

Following the general rules of complexity calculation that were used to estimate the complexity of the two standard concealment algorithms, ITU-T G.711-A, ANSI- T1-521-2000 (Annex B), we now estimate the required complexity of our proposed algorithm. A MAC operation is weighted by 1 DSP cycle, a Square root or division is weighted by 10 cycles and a Compare operation is given the weight of 2 cycles. The complexity of the whole technique is equal to the summation of the computational complexity of its blocks. We have previously addressed the complexity estimate of each of these blocks in Chapter 3. Here we present a brief, final form of the computational load of each of the blocks found in the concealment technique. The following table summarizes the set of equations we used to implement the algorithm.

Function	Equation
Autocorrelation:	• $x_1(n) = g(2n)$ $x_2(n) = g(2n+1)$ $\quad 0 \le n \le N-1$ • $x(n) = x_1(n) + jx_2(n) \qquad 0 \le n \le N_1 - 1$ • $x(n) \xrightarrow{FFT} X(k)$ • $X_1(k) = \frac{1}{2}\left(X(k) + X^*(N_1 - k)\right)$ • $X_2(k) = \frac{1}{2j}\left(X(k) - X^*(N_1 - k)\right)$ • $G(k) = X_1(k) + e^{-j\frac{\pi k}{N}} X_2(k) \qquad k = 0, 1, \cdots N$ • $r_{xx}(k) = \lvert G(k)\rvert^2 \qquad 0 \le k \le N$ • $r_{xx}(k) = r_{xx}(2N-k) \quad N+1 \le k \le 2N-1$ • $X_1(k) = r_{xx}(2k)$ $X_2(k) = r_{xx}(2k+1)$ • $X(k) = X_1(k) + jX_2(k) \qquad 0 \le k \le N_1 - 1$ • $X(k) \xrightarrow{IFFT} x(n)$ • $x_1(k) = \frac{1}{2}\left(x(k) + x^*(N_1 - k)\right)$ • $x_2(k) = \frac{1}{2j}\left(x(k) - x^*(N_1 - k)\right)$ • $P_{ss}(n) = x_1(n) + e^{-j\frac{\pi k}{N}} x_2(n) \qquad k = 0, 1, \cdots N-1$
LP coefficients estimation:	initialization: $a_1(1) = -\frac{P_{ss}(1)}{P_{ss}(0)}$ for $2 \le m \le P$: for $k = m$: $a_m(k) = \frac{-P_{ss}(m) + \left[P_{ss}(m-1) \cdots \; P_{ss}(1)\right]\left[a_{m-1}(1) \ldots a_{m-1}(m-1)\right]^T}{P_{ss}(0) + \left[P_{ss}(m-1) \cdots \; P_{ss}(1)\right]\left[a_{m-1}(m-1) \ldots a_{m-1}(1)\right]^T}$ for $1 \le k < m$: $a_m(k) = a_{m-1}(k) + a_m(m)\, a_{m-1}(m-k)$
Pitch Detection:	Coarse search followed by fine search in the same manner as the G.711-A Complexity can be extracted directly from [12], [48] which yields 192 MACs
Synthesis:	$S_1[n] = \left(\sum_{i=1}^{p} a_i \times S_1[n-i]\right) + \hat{S}[n] \times G$ $S[n] = S_1[n] \times \alpha + \hat{S}[n] \times \beta$

1. **The autocorrelation block**:

The autocorrelation function is calculated for 240 samples of input real sequences using the FFT- based frequency domain approach, for which a complete guide has been presented in Chapter 3. The complexity share of this block is equal to what we have presented in section 3.3.3.2:

$$4 \times 256 \times \log_2 256 + 6 \times 256 = 9728 \text{ MACs}$$

The value 256 used for the calculation in the closest higher power of 2 for the window size of the autocorrelation function (240 samples).

This value is calculated based on the number of multiplies only. Since it would probably not be possible for the additions required by the algorithm to be always implemented in parallel with the multiplies, then considering the additions would probably slightly increase the computational complexity. Note that an alternative approach could be used to calculate the autocorrelation. This approach is to calculate in the time- domain the first 51 autocorrelation coefficients for a window of size 160 and calculate the rest in the same manner as in [12]. The calculation of the 51 coefficients corresponds to 6885 MACs (Equation 3.18). The algorithm for the autocorrelation calculation in[12] corresponds to 3764 MACs thus the total complexity would be 10649 MACs. If a window size of 240 is used, the number of MACs would be 14729 MACs. The difference in quality is negligible between 240 samples and 160 samples autocorrelation. The main reason behind the choice of the 240 samples autocorrelation window was that the FFT approach has the same complexity in case of 240 or 160 autocorrelation window (the next power of 2 is 256 in both cases).

2. **The LP analysis**:

Like the autocorrelation function the estimation of the Linear Prediction coefficients has been explained in details in Chapter 3. A complete derivation of the complexity estimation of the Levinson- Durbin algorithm used to calculate the LP coefficients

was derived in Chapter 3. The complexity is presented in terms of ***P*** the prediction order, which is equal to fifty in our case. So the complexity share of this block can be directly calculated as:

$50 \times 10 + (50-1) + 50 \times (50-1) = 549 + 50 \times 49 = 2999$ MACS

3. **The Pitch Prediction**:

The pitch prediction process is a simple search for the peaks of the autocorrelation outputs. We have pointed out that this search can be even reduced by following the method presented in the ITU-T standard concealment. The complexity share of this block if we apply the two- stage search procedure that we described on the range between 40 samples to 120 samples is thus reduced to the comparison steps performed in the ITU-T G.711 PCM concealment tool [12]. The typical value is then 86 compare operations each weighted to be the double complexity of a MAC operation and thus the total complexity is equivalent to: 172 MACs.

4. **The Synthesis LP filter;**

The formation of the output speech frame is done through applying equation 7.5. The application of equation 7.5.A implies the multiplication of the 50 past speech samples by the corresponding LP coefficients and the addition of an excitation formed through the multiplication of the long- term excitation by the gain G.

$$S_1(n) = \sum_{i=1}^{50} S_1(n-i) \times a_i \; + \; \hat{S}(n) \times G \qquad 1 \le n \le 90 \qquad (7.5.A)$$

The number of samples calculated is ninety; the first eighty samples are placed in the concealed packet and the last ten are sent to the Overlap Add unit. From the above we can see that the complexity can be estimated to be $50 \times 90 + 90 = 4590$

5. **The Weighted Addition**:

In this block we add the short- term prediction multiplied by a weight of 0.7 and the long-term prediction multiplied by weight of 0.3. We can see that in terms of the number of multiplication required we need two 90 samples multiplications and so the total complexity of this block is 180 MACs.

From the above steps we can see that the total computational effort required for the concealment of the first lost packet in the erasure is:
$4590+172+2999+9728+180=17665$ MACs.

This is for 10 ms frame, or 1.765 Million MACs per second. Estimating the million MACs to be equivalent to 1 MIP the algorithm complexity is estimated at a reasonable 1.74 MIPS. That is affordable by most DSP processors. This is the peak complexity of the algorithm for the concealment of the first lost packet, while for subsequent erasure periods the complexity is reduced to the speech synthesis only, which is equivalent to around 0.5 MIPS.

It is worth noting that the ITU-T standard complexity is estimated to be 0.5 MIPS while the ANSI Annex B concealment standard has a complexity estimated at 2.3 MIPS for 10 ms packets.

7.7 Test Environment

7.7.1 Test Tool

The assessment tool used to produce the results of the concealment techniques is the Perceptual Estimation of Speech Quality (PESQ) standard P.862 developed by the ITU-T [42]. It is the newest and most accurate tool [34] in the perceptual based standards that has proved to give reliable estimation of the subjective quality tests. It depends on

comparing the distorted subject file to the original file and produces a score indicating the Absolute Category Rating score that would be estimated if subjective tests were applied. This has been discussed in detail in Chapter 4. The score is given in the range [-0.5 4.5]. A lower score indicates lower speech quality and a higher score indicates higher quality of the investigated file.

7.7.2 Test Files

The test was performed on a set of four speakers; two males and two females. They have been referred to in the results as: Male1, Male 2, Female1 and Female 2. Each of those speakers has 10 speech files to investigate, each containing two sentences in English of duration eight seconds. The format of the files was .wav files (Linear PCM). Thus, the total number of speech files is forty.

7.8 Results

The results presented in this section are organized into three main categories:

1. **The repeated loss pattern**:

   This category of tests is following the same test pattern previously applied in Chapter 6. This loss pattern produces one or two lost packets separated by three good packets. Recalling from the algorithm design, we depend on the last three packets in memory to produce the present lost packet. Thus in this test we are always using three good packets to produce the concealed ones. This test is applied for single and double packet loss. The final form of the algorithm is tested with the 6^{th} phase scores from the previous design history to show the improvement in the score, especially in the case of double packet loss. The comparison also includes the standard concealment algorithm of the ITU-T.

2. **The modified regular loss pattern**:

This test was especially developed because of the unstable behavior of the ANSI standard Annex B (LP-based PLC tool). We have stated earlier in this chapter that unstable behavior for some concealed packets was experienced when running the source code. Actually, when applying the regular single loss pattern, the numbers of packets that went unstable were mostly for the case of male files. In the case of female files the number was small and varied between two to twelve packets out of two hundred concealed packets. The documentation of the ANSI standard concealment algorithm (T1-521-2000) states that the scores expected from this algorithm are comparable to those obtained from the Annex A (T1-521-1999), which is the same ITU-T G.711 Annex A tool we are comparing with.

The test starts by applying the regular repeated loss test pattern on the ANSI concealment algorithm. The resulting concealed frames are tested for their amplitude. If an instability condition is detected this concealed packet is dropped and the corresponding correct packet is played instead. The test pattern is then modified to invert the current loss indicator. The modified test pattern is applied afterwards on the new proposed concealment algorithm and the ITU-T algorithm.

3. **The random loss test**:

This test category applies a random loss pattern of certain percentage on both the new concealment algorithm and the ITU-T standard algorithm. The different percentages tested are: 5%, 10%, 25% and 50%.

The concealment algorithm are given the following notations:

1. **New**: The new proposed concealment algorithm in its final form.
2. **Last Phase**: The phase 6 in the algorithm development. This is the phase described by the equation: $S(n) = \sum_{i=1}^{p} S(n-i) \times a_i + G \times \hat{S}(n)$.

3. **Standard1**: Denotes the ITU-T concealment standard algorithm.
4. **Standard2**: Denotes The ANSI LP-based standard algorithm (Annex B).

The complete sets of results are presented in Appendix B. Here, we only present the final average score of each speaker Male1, Male 2, Female 1, Female 2 for each of the previously mentioned tests.

7.8.1 Repeated Loss

The results in Appendix B are presented in table form for the four different speakers. Each speaker is presented as a separate case and has five tables. Each of the first four tables represents a different loss location while the last table contains the average score of the four tables above. Each column in the individual table represents a certain speech file (they are denoted by numbers starting from 1 to 10). Each row in the table represents a distinct concealment algorithm.

7.8.1.1 Single Packet Loss

In this subsection the scores obtained from the PESQ ITU-T assessment tool P.862 are presented for the single packet loss case. These results were developed by applying the same loss pattern implemented in the previous development phases. This pattern virtually divides the packets into groups of four where three packets out of the four are good and the fourth is bad. There are thus four different configurations for the loss pattern. They are presented as:

1. Case 1: The first packet in the group is lost.
2. Case2: The second packet in the group is lost.
3. Case3: The third packet in the group is lost.
4. Case 4: The fourth packet in the group is lost.

The complete results of each of the above cases are presented in Appendix B (B-1 Tables). The finale average scores of the four speakers are summarized in the following tables.

First Male:

	Avg.
New	3.01
Last_Phase	2.99
Standard1	2.83

Second Male:

	Avg.
New	3.01
Last_Phase	2.99
Standrard1	2.89

First Female:

	Avg.
New	2.8
Last_Phase	2.83
Standard1	2.65

Second Female:

	Avg.
New	2.75
Last_Phase	2.75
Standard1	2.58

We can see from the above tables that in the case of female speakers, the last phase (Phase 6) score is slightly higher on average than the final form of the algorithm. This is expected as we perform a weighted addition with the G.711-A concealed packets that have much lower score than the last phase concealed packets. This is not the case in the male files because the difference in the score between the last phase of the algorithm and the output of G.711_A is less than the female cases. However, this negligible loss in the female score is affordable especially considering that this addition will help in reducing the decay effect in the burst erasure case.

The following figure, Figure 7.3, also presents the final averages of each speaker for the three different concealment techniques, namely the New Algorithm, Last Phase and Standard1.

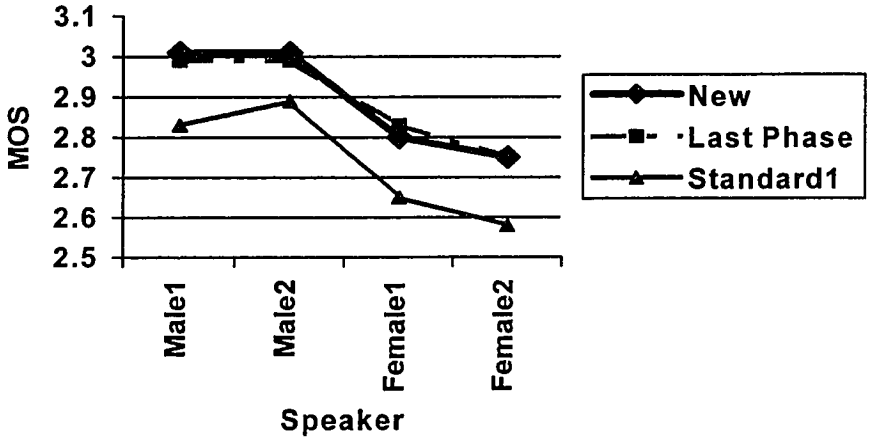


Figure 7.3: The Average Results of the Single Repeated packet loss

7.8.1.2 Double Packet Loss

The loss pattern in this case virtually divides the packets into groups of five. Thus, three packets out of the five are good and two consecutive packets are bad. There are five different configurations for the loss pattern. We present four of them below. They provide good indication on the average expected performance of each system:

1. Case 1: The first and second packets in the group are lost.
2. Case2: The second and the third packets in the group are lost.
3. Case3: The third and fourth packets in the group are lost.
4. Case 4: The fourth and fifth packets in the group are lost.

The last possibility, which we have not presented, is when the first and last packets are lost. In such case the burst loss will be two consecutive packets from two different groups.

The complete sets of results of the above cases are presented in Appendix B (Tables B-2). In the following four tables we summarize the performance of the three methods (the new algorithm, the last phase and the G.711-A). These summary tables include the average

scores of each of the previously mentioned methods when applied to each individual speaker.

First Male:

	Avg.
New	2.34
Last_Phase	2.18
Standard1	2.24

Second Male:

	Avg.
New	2.42
Last_Phase	2.3
Standard1	2.32

First Female

	Avg.
New	2.29
Last_Phase	2.25
Standard1	2.25

Second Female

	Avg.
New	2.14
Last_Phase	2.08
Standard1	2.07

We can see from the above tables and the complete set of results presented in Appendix B (Tables B-2) that the weighted addition step that we have added to the last phase actually helped a lot in boosting the score of the double packet loss case. This increase is more obvious in the male files. The last phase scores showed to be less than the ITU-T G.711-A concealment algorithm by 0.065 on average. In the final case, the situation turned to be the opposite and the score of the final phase is almost always better than the score achieved by implementing the standard algorithm. In the female cases, the final form of the algorithm has added an improvement in the performance of the double packet concealment. Eventually, we find that the final form of the algorithm is better than the ITU-T concealment standard in treating the double packet loss.

The following figure, Figure 7.4, also summarizes the final averages of each speaker for the three different concealment techniques, namely the New Algorithm, Last Phase and Standard1.

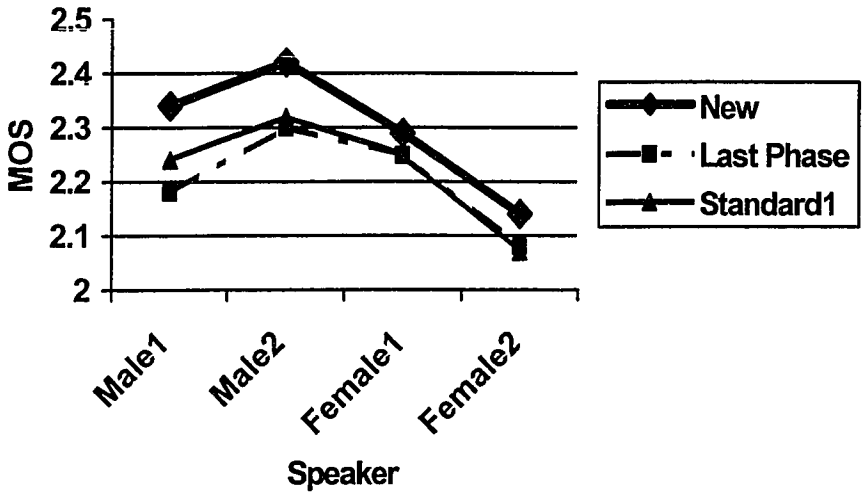


Figure 7.4: The Average Results of the Double Repeated packet loss

7.8.2 Modified Loss Pattern

As previously mentioned, this test is performed especially to suit the nature of the second standard (ANSI Annex B: T1-521-2000) source code. The test is limited to the single packet loss patterns. It is considered indicative to show the performance of each of the three concealment techniques. The following four tables present the final average scores obtained by applying the modified loss pattern on each of the four speakers. The complete sets of results are presented in the Tables B-3 in Appendix B.

First Male

	Avg.
New	3.11
Standard1	3.02
Standard2	2.81

Second Male

	Avg.
New	3.15
Standard1	2.98
Standard2	2.81

First Female

	Avg.
New	2.97
Standard1	2.81
Standard2	2.79

Second Female

	Avg.
New	2.83
Standard1	2.65
Standard2	2.61

From the above tables and the complete results in B-3 we can see that the resulting loss pattern revealed the high performance of the new proposed algorithm compared to both the ITU-T (G.711-A) and the ANSI (T1-521-200) concealment standards. The difference of the scores between the new algorithm and the ITU-T standard algorithm under the same loss conditions jumps in some cases to more than 0.2.

The following figure, Figure 7.5, also presents the final averages of each speaker for the three different concealment techniques, namely the New Algorithm, Standard1 and Standard2.

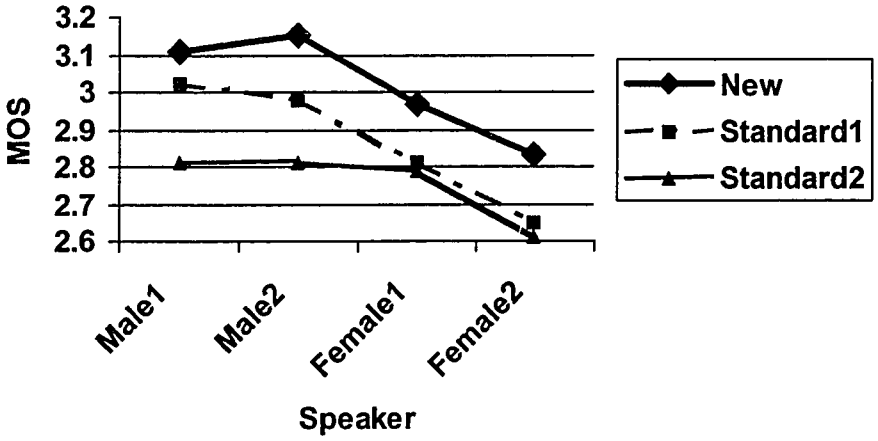


Figure 7.5: The Average Results of the Modified Single Repeated packet loss

7.8.3 Random Loss

The loss pattern in this case is formed randomly with no constraints on the burst length or the loss location. We have chosen four loss rates that vary between small and tolerable for most coders (5%) to average and realistic (10%), 25% or to a very high loss rate of 50%. Considering that we have 800 packets (80 samples each) in each of the wave files that we tested and that we have ten files per speaker, we can argue that the random loss pattern is gaussian. Thus, we performed only one test for each loss rate. The average score of each speaker is presented below. The detailed scores of each individual wave file are presented in Tables (B-4) in Appendix B.

7.8.3.1 Random Loss of Rate 5%

Below we present the average score obtained for each speaker. Table B-4-1 in Appendix B contains the complete set of results for this test.

First Male

	Avg.
New	3.66
Standard1	3.48

Second Male

	Avg.
New	3.51
Standard1	3.4

First Female

	Avg.
New	3.42
Standard1	3.33

Second Female

	Avg.
New	3.44
Standard1	3.29

We can see from the above average scores and the associated complete tables of scores in Appendix B (Tables B-4-1) that in the case of random loss of 5%, the new algorithm has always better performance than the ITU-T standard concealment algorithm. The difference in the score between the two algorithms is at least 0.1 and can actually reach 0.18.

The following figure, Figure 7.6, also presents the final averages of each speaker for the New Algorithm and Standard1.

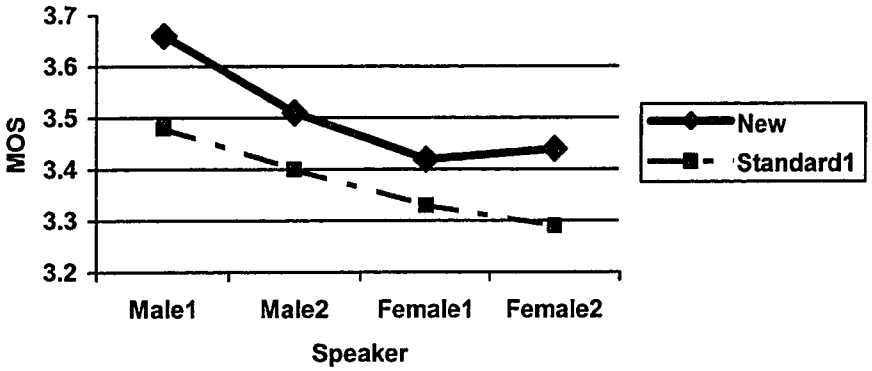


Figure 7.6: The Average Results for 5% Random Packet Loss

7.8.3.2 Random Loss of Rate 10%

The following four tables present briefly the average scores obtained from both new algorithm and the G.711-A concealment tool in the case of 10% packet loss. The complete detailed set of results containing the score for each of the forty wave files is presented in Appendix B (Tables B-4-2).

First Male

	Avg.
New	3.19
Standard1	3.08

Second Male

	Avg.
New	3.27
Standard1	3.13

First Female

	Avg.
New	3.05
Standard1	2.94

Second Female

	Avg.
New	3.01
Standard1	2.85

These scores indicate superior performance of the new algorithm compared to the existing standard of the ITU-T. The scores shown in the tables above show an average increase of 0.131 when the new algorithm is used to conceal the erasure. The 10% random loss is considered a realistic value of the expected erasure percentage in an IP network. This percentage can in some cases be as high as 25%. In the following figure, Figure 7.7, we also present the final averages of each speaker for the New Algorithm and Standard1 for 10% packet loss.

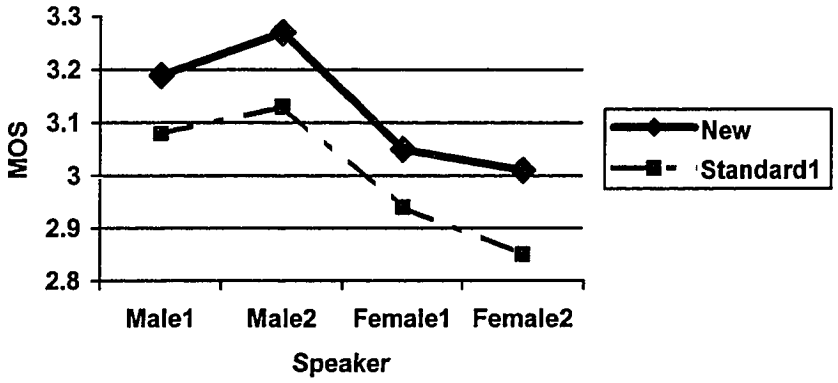


Figure 7.7: The Average Results for 10% Random Packet Loss

7.8.3.3 Random Loss of Rate 25%

The following four tables present a final average score of this test for each speaker. The complete set of results is presented in the table set B-4-3 in Appendix B.

First Male

	Avg.
New	2.74
Standard1	2.60

Second Male

	Avg.
New	2.79
Standard1	2.65

First Female

	Avg.
New	2.66
Standard1	2.56

Second Female

	Avg.
New	2.6
Standard1	2.45

Figure 7.8 also summarizes the final average scores expected from both techniques (the new algorithm and the ITU-T G.711-A) in the case of 25% packet loss.

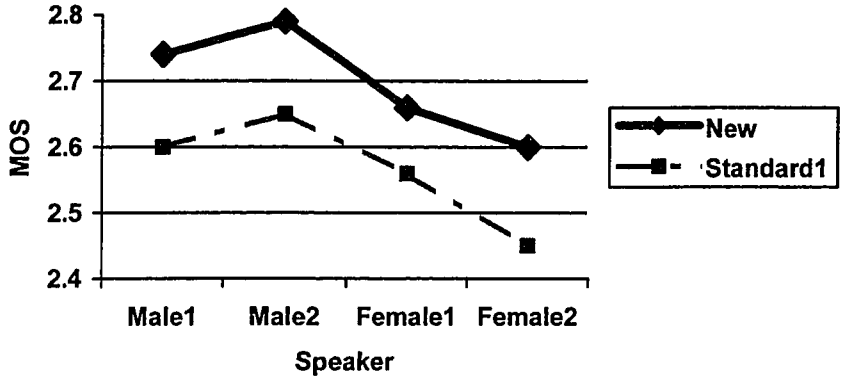


Figure 7.8: The Average Results for 25% Random Packet Loss

In this case we can see that the new algorithm still performs better than the standard concealment algorithm. We can actually consider the three different loss percentages that we presented, namely the 5%, 10% and 25% cases, to present a faithful proof of the excellent performance expected from the application of the new concealment algorithm in IP networks. However, in order to present a complete set of different erasure percentages, we proceed to a very high loss rate of 50%. The quality of the speech in this case is expected to deteriorate and may even lose intelligibility. The score in this case is presented for documentation purpose, as we do not expect either algorithm to perform well in such harsh conditions.

7.8.3.4 Random Loss of Rate 50%

In the following set of tables the results of a 50% loss pattern are shown as final average for each individual speaker. The detailed behaviour of each single wave file is indicated by a score presented in tables B-4-4 in Appendix B.

First Male

	Avg.
New	1.78
Standard1	1.71

Second Male

	Avg.
New	1.85
Standard1	1.88

First Female

	Avg.
New	1.76
Standard1	1.74

Second Female

	Avg.
New	1.54
Standard1	1.54

We can see that in this case both algorithms have weak performance and the results of them are comparable in both male and female cases. This is more obvious in the following figure, Figure 7.9.

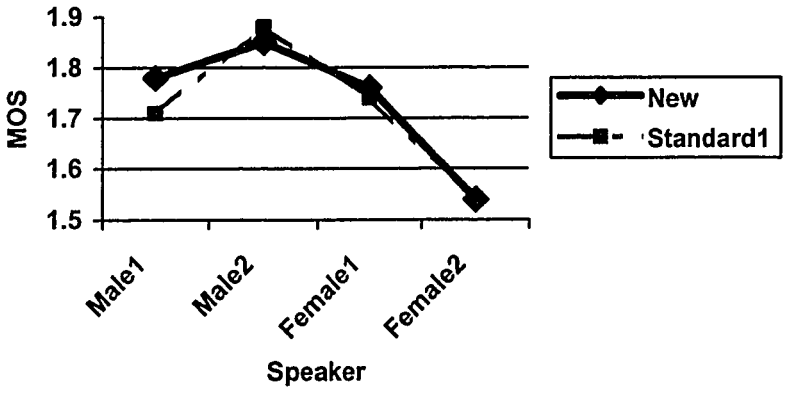


Figure 7.9: The Average Results for 50% Random Packet Loss

7.9 Summary

This chapter was dedicated to present the final form of the concealment algorithm that we propose. We started by a quick review of the principle of work implemented in the previous chapter. We started the research with the idea of implementing an LP- based prediction of the lost packet. We replaced the input excitation to the synthesis filter by a small fraction (typically 0.01) of the pitch- predicted replica of the missed speech frame. This pitch replica is actually extracted in the same manner as in the ITU-T standard concealment algorithm. The resulting predicted packet of the synthesis filter is then multiplied by a factor (0.7) and then added to the long- term prediction (again, the pitch replica estimated as in the ITU-T standard) multiplied by a factor of (1-0.7=0.3). This weighted sum proved to provide a better prediction of the concealed frame as we presented in the results section.

The new algorithm was presented in details. We showed how the new algorithm works in the case of one lost packet and in the case of burst erasures. We also provided a summary of the equations implemented in the algorithm and the corresponding complexity. The total complexity of the algorithm was estimated to be 1.74 MIPS. This complexity is considered affordable for most DSP processors. The G.711-A concealment standard proposed by the ITU-T has a complexity of 0.5 MIPS while the complexity of the ANSI concealment algorithm (T1-521-2000) for a 10 ms packet size is 2.3 MIPS.

We then proceeded to present the test setup and the test files. The test was performed on forty files belonging to four different speakers, two males and two females. The test patterns were implemented in both repeated loss and random loss at different rates. We mainly compared the results to the ITU-T standard concealment algorithm (G.711-A). We also provided a separate section of indicative results for tests performed on the new algorithm, the ITU-T standard tool and the ANSI (T1-521-2000) concealment tool. The new algorithm proved to be superior to the other standard tools in all the tests that we performed, except for the random loss test at rate 50% where the results were comparable to the ITU-T standard tool (G.711-A).

8. Conclusion and Future work

8.1 Contribution

This thesis addresses the implementation of a new concealment algorithm for the PCM ITU-T G.711 standard. The PCM speech compression standard is an eligible candidate to be the speech compression standard in IP telephony because of its simplicity and low delay. Results in [55] showed that the G.711 has significantly better performance and better tolerance to delay than other compression standards under normal operation conditions. The relatively large bandwidth required by PCM no longer stands as a barrier against a promising wide application of this compression algorithm in the new active and dynamic field of VoIP. New technology can support the 64 kb/s required by PCM with no difficulty as it now supports the video conferencing that demands much larger bandwidth for its image compression side. The problem that remained unsolved was the weak performance of the PCM coders in the periods of erasure. Packet loss problem is a well-known impairment in IP networks. It happens quite often and can reach high rates of 25%.

The thesis started with a review of the nature of speech production and perception. We also provided a comparative analysis of the different speech coding categories. Chapter 3 presented a detailed analysis of the different mathematical operations implemented in the Linear Prediction process. The most interesting point in this analysis is that it has a step-by-step complexity calculation of the different equations involved. In Chapter 4 we have presented a review of the different methods to evaluate the performance of a speech transmission system. In particular, we have presented the ITU-T P.862 PESQ speech assessment tool. Actually, this tool helped a lot in providing faithful unbiased scores for the different files and cases that we implemented. It played the main role in configuring the different parameters in our concealment model. Lastly, Chapter 5 provided an extensive review of the impairments existing in IP networks, namely delay, jitter and packet loss. This extensive survey collected a wide variety of creative suggestions for

dealing with the different problems facing a wider application of real-time speech transmission over IP networks.

The main contribution of the thesis is the proposal of a novel concealment algorithm to be added to the PCM coder. This concealment algorithm requires an affordable added complexity, which is well justified by considering the significant improvement it adds to the erasure concealment behavior of the coder.

A performance comparison was made between the new algorithm and the patented standard concealment algorithms provided by both the ITU-T and the ANSI. The results showed the superiority of the proposed technique.

The concealment model that we presented is actually an original implementation of the Linear Prediction parameters. The idea of the weighted addition between the long-term prediction (the RORPP prediction provided by the ITU-T G.711-A) and the output of the Linear Prediction (LP) model that we implemented provides the opportunity of getting the best of the two techniques; if one of the techniques works extremely better than the other in a certain period, the final combined form is affected by this superior performance, and if the scores are comparable the score of the final combined form is typically better than both the separate concealment algorithms.

We tried different values for the weighted addition coefficients α and β. We found that the best values are 0.7 and 0.3 respectively. Other values could work better in the case of male files. However, considering the performance of female files performance in the single lost packet test, we found that those values gave the smallest degradation from the previous phase scores and still guaranteed improvement in the performance of the male files in the case of single and double packet loss.

8.2 Future Work

While doing the research, several ideas appeared worth pursuing. These were mainly improvements to the proposed algorithm. The main ideas can be summarized as:

1. In the proposed algorithm we depended on the past speech packets to estimate the missing one. From the discussion of the jitter problem in Chapter 5, we recall that this problem is solved by implementing a jitter buffer at the receiver end to store and re-order packets before sending them to the receiver side. This buffer is usually of size 50 ms (five 10 ms packets). Our novel concealment algorithm can make use of this delay by involving future packets in the prediction of the current lost one. Such interpolated prediction can provide better prediction at the boundaries of voiced/ unvoiced speech segments. One possibility would be to transform the LP coefficients into Line Spectral Frequencies (LSF) and perform the interpolation between past and future samples in that domain and then transform it back into the LP domain [25]. The efficiency of this method and the added improvement in performance would have to be judged considering the added complexity and delay expected from this future interpolation step.

2. Adapting the values of the parameters of the novel concealment method to the nature of the speech segments seems promising. These parameters are namely, the gain G (the fraction of the long term contribution which replaces the input excitation to the LP-model) and the two parameters of the weighted addition. The values provided in the thesis are believed to be efficient and valid in general. However, permitting these values to adapt according to the nature of speech segment may result in better quality of the concealment. For voiced segments, repeated waveform segments the pitch prediction may be weighted higher than 0.3, and for fast changing speech and transitions a weight of higher than 0.7 can be given to the LP- prediction part. More information on the methods for adaptive filtering can be found in [3].

3. It would also be of interest to test the performance of the concealment algorithm in a global complex IP system that would include elements such as echo cancellers and echo suppressors. As reported in [67], the performance of the concealment algorithms can be affected by these elements.

Bibliography

[1] Wolfgang H., Pitch Determination of Speech Signals, *Algorithms and Devices,* Springer Series in Information Science, 1983.

[2] C. Montminy, *A study of Speech Compression Algorithms for Voice Over IP*, Master's thesis, Ottawa University, School of Information Technology and Engineering, 2000.

[3] S. Haykin, *Adaptive Filter Theory*. New Jersey: Prentice-Hall, 3rd ed., 1996.

[4] Douskalis B. *IP Telephony: The Integration of Robust VoIP Services*, Hewlette-Packard Professional Books, Prentice Hall, Upper Saddle River, New Jersey, 2000.

[5] J.D Markel, A.H. Gray, *Linear Prediction of Speech*, Springer Series in Information Science, 1976.

[6] Glefand S. A. Hearing: *An Introduction to Psychological and Physiological Acoustics*, Third Edition, Marcel Dekker Inc., New York, 1998.

[7] R.Goldberg, L. Reik: *A Practical Handbook of Speech coders*, CRC press, Boca Raton, 2000.

[8] *ITU-T G.729*: *Coding of speech at 8 kb/s using conjugate-structure algebric-code excited linear-prediction (CS-ACELP)*, International Telecommunication Union, March 1996.

[9] *ITU-T G.723.1*: *Dual Rate Speech Coder for Multimedia Communications Transmitting at 5.3 and 6.3 kb/s*, International Telecommunication Union, March 1996.

[10] *ITU-T G.729 Annex A*: *Reduced Complexity 8kb/s CS-ACELP Speech Codec*, International Telecommunication Union, November 1996.

[11] *ITU-T G.711: Pulse Code Modulation (PCM) of voice frequencies,* International Telecommunication Union, November 1988.

[12] *ITU-T G.711 Annex A*: *A High Quality Low-Complexity Algorithm for Packet Loss concealment with G.711*, International Telecommunication Union, November 2000.

[13] Haykins S. *Communication Systems:* Third Edition, John Willey & Sons, Toronto, 1994.

[14] Proakis J. G. and Manolakis D.G. *Digital Signal Processing: Principles, Algorithms, and Applications*, Third Edition, Prentice Hall, Upper Saddle River, New Jersey, 1996.

[15] Minoli D. and Minoli E.: *Delivering Voice over IP Networks*, Wiley Computer Publishing, Toronto, 1998.

[16] B.S. Atal et al. :*Speech and Audio Coding for Wireless and Network* Applications, Kluwer Academic Publishers, Boston, 1993.

[17] *ITU-T G.728*: *Coding of Speech at 16 kbit/s using low-delay Coded-Excited Linear Prediction*, International Telecommunication Union, February 1992.

[18] Benyassine A. et al. " ITU-T Recommendation G.729 Annex B: A Silence Compression Scheme for use with G.729 Optimized for V.70 Digital Simultaneous Voice and Data Applications", *IEEE Communication Magazine*, September 1997, pp. 64-73.

[*19*] *ITU-T G.723.1 Annex A*: *Silence Compression Scheme for Dual Rate Speech Coder for Multimedia Communications G.723.1,* International Telecommunication Union, November 1996.

[20] *ITU-T P Supplement 23*: *ITU-T Coded-Speech Database*, International Telecommunication Union, February 1998.

[21] Stallings W. *Data and Computer Communications*, Fifth Edition, Prentice Hall, Upper Saddle River, New Jersey, 1997.

[22] *ITU-T G.726*: *40, 32, 24, 16 kbit/s Adaptive Differential Pulse Code Modulation (ADPCM),* International Telecommunication Union, December 1990.

[23] Bradenburg K. "Perceptual Coding of High Quality Digital Audio", *Applications of Digital Signal Processing to Audio and Acoustics*, Kulwer Academic Press, 1998.

[24] *ITU-T G.727: 5-, 4-, 3- and 2-bit/sample Embedded Adaptive Differential Pulse Code Modulation (ADPCM),* International Telecommunication Union, December 1990.

[25] E. Mahfuz, *Packet Loss Concealment for Voice Over IP*, Master thesis, McGill University, Department of Electrical Engineering, 2001.

[26] Fielder L.D. et al. "AC-2 and AC-3: Low-Complexity Transform-Based Audio Coding", Papers on Digital Audio Bit Rate Reduction, 1997, pp. 54-72.

[27] B. Li et al.: QoS- Enabled Voice Support in the Next-Generation Internet: Issues, Existing Approaches and Challenges, *IEEE Communication Magazine*, April 2000, pp.54–61.

[28] Yensen T. et al. " Determining Acoustic Round Trip Delay for VoIP Conferences", *1998 IEEE Second Workshop on Multimedia Signal Processing*, pp.161-166.

[29] Ramjee R. et al. " Adaptive Playout Mechanisms for Packetized Audio Applications in Wide-Area Networks ", *IEEE INFOCOM 1994*, vol.2, pp.680-688.

[30] *ITU-T H.323: Packet- Based Multimedia Communication Systems*, International Telecommunication Union, March 1999.

[31] Noll P. "MPEG Digital Audio Coding" *IEEE Signal Processing Magazine*, September 1997, pp.59-81.

[32] Kondoz A.M., Digital Speech: *Coding for Low Bit Rate Communication System*, John Wiley & Sons, Toronto, 1994.

[33] P.B Denes et al. The Speech Chain: *The Physics and Biology of Spoken Language*. Waverly Press, Baltimore 1963.

[34] A.W. Rix et al. Perceptual Evaluation of Speech Quality (PESQ)- A New Method for Speech Quality Assessment of Telephone Networks and Codecs, *ICASSP*, 2001.

[35] Black U. *Advanced Internet Technologies* , Second Edition, Prentice Hall, Upper Saddle River, New Jersey, 2000.

[36] G. Thomsen et al. Internet Technology: Going Like Crazy, *IEEE Spectrum*, May 2000, pp. 52-58.

[37] Alwyn L. and Broom S. " Feasibility of Non-Intrusive Monitoring of Voice Call Quality in Internet Telephony", *British Telecommunication Engineering*, August 1999, pp.15-20.

[38] Bernards J. and Stemerdink J. " A Perceptual Speech quality Measure Based on A Psychacoustic Sound Representation", *Journal of Audio Engineering Society*, March 1994, pp. 115-123.

[39] *ITU-T P.800: Methods for Objective and Subjective Assessment of Quality*, International Telecommunication Union, August 1996.

[40] *ITU-T P.830: Subjective Performance Assessment of Telephone–band and Wide-Band*, International Telecommunication Union, February 1996.

[41] *ITU-T P.861: Objective Quality Measurement of Telephone-Band (300-3400 Hz) Speech Codecs,* International Telecommunication Union, February 1998.

[42] *ITU-T P.862: Perceptual evaluation of speech quality (PESQ), an objective method for end-to-end speech quality assessment of narrowband telephone network and speech codecs*. International Telecommunication Union, May 2000.

[43] A. Barberis et al.: A Simulation Study of Adaptive Voice Communications on IP Networks, *IEEE Computer Communication Magazine*, vol. 24, May 2001, pp-757-767.

[44] Hassan M. and Nayandoro A. " Internet Telephony: Services, Technical Challenges, and Products", *IEEE Communication Magazine*, April 2000, pp. 96-103.

[45] Alcatel Networks Corporation, *Voice over DSL Quality Study*, Technical paper. *http://www.alcatel.ca/doctypes/opgrelatedinformation/child1/VODSLqos.pdf*

[46] Nortel Networks Corporation, *Packet Loss and Packet Loss Concealment* Technical Brief. *http://www.nortelnetworks.com/products/01/succession/es/collateral/tb_pktloss.pdf*

[47] *ITU-T G.107: The E-model, a computational model for use in transmission planning*, International Telecommunication Union, December 1998.

[48] ANSI, *Packet Loss Concealment algorithm for use with ITU-T Recommendation G.711*, December 1999. ANSI Recommendation T1.521-1999 (Annex A).

[49] *ITU-T G.113: Transmission System and Media: Transmission impairments*, International Telecommunication Union, February 1996.

[50] *ITU-T G.114: One Way Transmission Time,* International Telecommunication Union, February 1996.

[51] M. Rowe, "Measure VoIP Networks for Jitter and Loss", Test & Measurement World, December 1999. *http://www.tmworld.com/articles/12_1999_VoIP.htm*

[52] S.B.Azami et al. "Channel Loss and Queuing Loss Tradeoffs in Voice Transmission over Packet Switching Systems", ICC conference 2002.

[53] Perkins M.et al. "Characterizing the Subjective Performance of the ITU-T 8kb/s Speech Coding Algorithm- ITU-T G.729", IEEE Communication Magazine, September 1997, pp. 74-81.

[54] ANSI, *Packet Loss Concealment algorithm for use with ITU-T Recommendation G.711*, July 2000. ANSI Recommendation T1.521-2000 (Annex B).

[55] D. De Vleeschaur et al. " Quality bounds for Packetized Voice Transport", Alcatel Telecommunication review, 1st Quarter 2000.

[56] Katsuyoshi L. et al. " *Performance Evaluation of the Architecture for End-to-End Quality-of-Service Provisioning*", IEEE Commuincation Magazine, April 2000, pp.76-81.

[57] S. Casner and V. Jacobson, "Compressing IP/ UDP/ RTP Headers for Low-Speed Serial Links", *IETF RFC 2508*, February 1999.

[58] T. Kostas et al., " Real-Time Voice over Packet- Switched Networks, " *IEEE Network*, January/February 1998, pp. 18-27.

[59] Goodman B., "Internet Telephony and Modem Delay", IEEE Network, May/ June 999, pp. 8-16.

[60] Perkins C., Hodson O. and Hardman V., " A Survey of Packet Loss Recovery Techniques for Streaming Audio", IEEE Networks, September/ October 1998, pp. 40-48.

[61] Cisco Systems, " Echo Analysis for Voice over IP", Cisco Documentation, *http://www.cisco.com/univercd/cc/td/doc/cisintwk/intsolns/voipsol/ea_isd.htm*

[62] *ITU-T G.131: Control of Talker Echo*, International Telecommunication Union, August 1996.

[63] *ITU-T G.165: Echo Cancellers*, International Telecommunication Union, March 1993.

[64] T. Nguyen, et al., "Voice over IP Service and Performnace in Satellite Networks", *IEEE Communication Magazine*, March 2001, pp. 164-171.

[65] Nortel Networks Corporation, *Voice over packet: An assessment of voice performance on packet networks.* White Paper. *http://www.nortelnetworks.com/products/01/succession/es/doclib.html*

[66] C. Perkins et *al.*, " RTP Payload for redundant Audio Data," *IETF RFC 2198*, Sept. 1997.

[67] Y. Huang, *Effects of Vocoder Distortion and Packet Loss on Network Echo Cancellation*, Master's thesis, Carleton University, Faculty of Engineering, Department of Systems and Computer Engineering, 2000.

Appendix A: Design History Results

1.Results Setup:

The results are presented in four tables for each speaker. The first three present the scores for a certain loss location for 10 different sentences presented in column form from one to ten. The last table, titled average, represents the average of the score of each sentence (column) for the 3 different loss locations.

A.1 Test pattern (Case of 160 Autocorrelation window)

In the following pages the results of the test pattern described before (G G B G G B) are shown for 4 different speakers, two males and two females. Forty different wave files, 10 for each speaker, have been tested with the single packet loss of 33% by 3 different concealment methods. The 1st method referred to as prediction is the new method (Phase 2 model), while the 2nd row referred to as standard shows the scores obtained by applying the ITU-T standard (the long- term prediction) and the third row shows the results when employing the commercially used previous packet repetition (without any modification in its shape or gain).

The test pattern has been repeated for the three different possible locations of the packet loss within the group of 3 packets. This is referred to above each table by rem(I,3)=0 or 1 or 2 where rem stands for the remainder and I is the index of the current packet inside the tested wave file. So we have 3 different scores for each wave file, for each method respectively. As expected the same pattern of loss is applied identically on the 3 concealment methods.
In case of 240- samples autocorrelation window the loss rate is 25% and the repetition size of the pattern is 4:

- rem (I,4)=0
- rem (I,4)=1

- rem(I,4)=2
- rem(I,4)=3

Where I is the number of the packet in the repetition pattern ($0 \leq i \leq 3$)

A-1: Results of 2nd Phase

Male1:

Rem(I,3)=0

file	1	2	3	4	5	6	7	8	9	10	**Avg.**
Prediction	2.727	2.631	2.427	2.573	2.626	2.547	2.641	2.635	2.441	2.646	**2.59**
Standard	2.796	2.849	2.459	2.533	2.656	2.652	2.804	2.647	2.499	2.623	**2.65**
Repetition	2.264	1.988	1.993	2.210	2.195	2.135	2.166	2.336	2.304	2.383	**2.2**

Rem(I,3)=1

file	1	2	3	4	5	6	7	8	9	10	**Avg.**
Prediction	2.711	2.468	2.541	2.589	2.792	2.478	2.584	2.739	2.671	2.720	**2.63**
Standard	2.864	2.762	2.550	2.687	2.782	2.567	2.669	2.705	2.774	2.807	**2.72**
Repetition	2.232	2.065	2.027	2.131	2.340	2.152	2.294	2.370	2.190	2.280	**2.21**

Rem(I,3)=2

file	1	2	3	4	5	6	7	8	9	10	**Avg.**
Prediction	2.707	2.642	2.486	2.694	2.649	2.601	2.579	2.694	2.655	2.822	**2.65**
Standard	2.894	2.816	2.731	2.706	2.564	2.660	2.619	2.713	2.641	2.814	**2.72**
Repetition	2.261	1.957	1.969	2.293	2.361	2.131	2.210	2.388	2.269	2.318	**1.98**

Average:

file	1	2	3	4	5	6	7	8	9	10	**Avg.**
Prediction	2.715	2.58	2.485	2.619	2.689	2.542	2.601	2.689	2.589	2.729	**2.62**
Standard	2.851	2.809	2.58	2.642	2.667	2.626	2.697	2.688	2.638	2.748	**2.69**
Repetition	2.252	2	1.996	2.211	2.299	2.139	2.223	2.254	2.254	2.327	**2.13**

Male2:

Rem(I,3)=0

file	1	2	3	4	5	6	7	8	9	10	Avg.
Prediction	2.561	2.620	2.594	2.598	2.753	2.499	2.707	2.723	2.644	2.593	**2.63**
Standard	2.755	2.940	2.779	2.791	2.690	2.486	2.914	2.836	2.763	2.841	**2.78**
Repetition	2.101	2.216	2.095	1.925	2.294	1.873	2.086	2.228	2.178	2.001	**2.1**

Rem(I,3)=1

file	1	2	3	4	5	6	7	8	9	10	Avg.
Prediction	2.545	2.600	2.671	2.630	2.604	2.515	2.646	2.733	2.633	2.552	**2.61**
Standard	2.636	2.840	2.806	2.871	2.667	2.732	2.844	2.827	2.826	2.912	**2.8**
Repetition	2.225	2.234	2.064	1.977	2.323	1.874	2.173	2.259	2.174	2.012	**2.13**

Rem(I,3)=2

file	1	2	3	4	5	6	7	8	9	10	Avg.
Prediction	2.567	2.680	2.680	2.518	2.680	2.427	2.521	2.610	2.516	2.603	**2.58**
Standard	2.523	2.841	2.959	2.821	2.703	2.699	2.795	2.623	2.782	2.811	**2.76**
Repetition	2.164	2.176	2.037	1.938	2.282	1.903	2.224	2.269	2.129	2.005	**2.11**

Average:

file	1	2	3	4	5	6	7	8	9	10	Avg.
Prediction	2.558	2.633	2.648	2.582	2.679	2.480	2.625	2.689	2.598	2.583	**2.61**
Standard	2.638	2.874	2.848	2.828	2.69	2.639	2.851	2.762	2.790	2.855	**2.78**
Repetition	2.163	2.209	2.065	1.95	2.185	1.883	2.161	2.252	2.163	2.006	**2.1**

Female1:

Rem(I,3)=0

file	1	2	3	4	5	6	7	8	9	10	**Avg.**
Prediction	2.650	2.495	2.519	2.698	2.759	2.637	2.756	2.587	2.819	2.746	**2.67**
Standard	2.656	2.439	2.424	2.641	2.708	2.622	2.690	2.438	2.669	2.659	**2.59**
Repetition	2.022	1.949	1.891	1.984	1.931	1.919	2.019	2.115	2.045	1.924	**1.98**

Rem(I,3)=1

file	1	2	3	4	5	6	7	8	9	10	**Avg.**
Prediction	2.638	2.697	2.685	2.663	2.649	2.553	2.663	2.676	2.751	2.592	**2.66**
Standard	2.521	2.635	2.603	2.515	2.660	2.542	2.630	2.528	2.756	2.583	**2.6**
Repetition	2.029	1.893	1.880	2.032	1.899	1.943	2.037	2.041	2.031	1.895	**1.97**

Rem(I,3)=2

file	1	2	3	4	5	6	7	8	9	10	**Avg.**
Prediction	2.601	2.632	2.543	2.682	2.734	2.526	2.691	2.660	2.795	2.731	**2.66**
Standard	2.482	2.593	2.473	2.533	2.701	2.544	2.648	2.549	2.682	2.646	**2.59**
Repetition	2.017	1.936	1.883	1.983	1.845	1.967	1.984	2.052	2.024	1.851	**1.95**

Average:

file	1	2	3	4	5	6	7	8	9	10	**Avg.**
Prediction	2.63	2.608	2.582	2.681	2.714	2.572	2.703	2.641	2.788	2.69	**2.66**
Standard	2.553	2.556	2.5	2.563	2.690	2.569	2.656	2.505	2.702	2.629	**2.59**
Repetition	2.023	1.926	1.885	2	1.892	1.943	2.013	2.069	2.033	1.89	**1.97**

Female2:

Rem(I,3)=0

file	1	2	3	4	5	6	7	8	9	10	**Avg.**
Prediction	2.562	2.667	2.660	2.660	2.810	2.638	2.496	2.504	2.591	2.618	**2.62**
Standard	2.518	2.644	2.633	2.562	2.748	2.506	2.437	2.405	2.510	2.562	**2.55**
Repetition	1.692	1.828	1.939	1.955	2.015	1.827	1.485	1.831	1.750	1.920	**1.82**

Rem(I,3)=1

file	1	2	3	4	5	6	7	8	9	10	**Avg.**
Prediction	2.486	2.692	2.553	2.585	2.703	2.492	2.418	2.579	2.726	2.569	**2.58**
Standard	2.531	2.617	2.421	2.527	2.572	2.455	2.247	2.477	2.679	2.390	**2.49**
Repetition	1.739	1.939	1.965	1.991	2.075	1.939	1.673	1.824	1.702	1.930	**1.88**

Rem(I,3)=2

file	1	2	3	4	5	6	7	8	9	10	**Avg.**
Prediction	2.391	2.567	2.675	2.656	2.780	2.377	2.343	2.468	2.585	2.614	**2.55**
Standard	2.431	2.474	2.481	2.526	2.629	2.315	2.174	2.437	2.477	2.483	**2.44**
Repetition	1.724	1.914	1.973	1.924	2.063	1.815	1.570	1.925	1.821	1.927	**1.87**

Average:

file	1	2	3	4	5	6	7	8	9	10	**Avg.**
Prediction	2.48	2.624	2.629	2.634	2.764	2.502	2.419	2.517	2.634	2.6	**2.58**
Standard	2.49	2.578	2.512	2.538	2.650	2.425	2.286	2.439	2.555	2.478	**2.49**
Repetition	1.718	1.894	1.959	1.96	2.051	1.86	1.567	1.89	1.758	1.926	**1.86**

A-2: Results for 3rd Phase

Male1:

Rem(I,3)=0

file	1	2	3	4	5	6	7	8	9	10	**Avg.**
With	2.970	2.898	2.651	2.654	2.923	2.774	2.901	2.924	2.644	2.768	**2.81**
Without	2.727	2.631	2.427	2.573	2.626	2.547	2.641	2.635	2.441	2.646	**2.59**
Standard	2.796	2.849	2.459	2.533	2.656	2.652	2.804	2.647	2.499	2.623	**2.652**
Repetition	2.264	1.988	1.993	2.210	2.195	2.135	2.166	2.336	2.304	2.383	**2.2**

Rem(I,3)=1

file	1	2	3	4	5	6	7	8	9	10	**Avg.**
With	2.946	2.752	2.785	2.710	2.970	2.737	2.809	2.929	2.897	2.938	**2.85**
Without	2.711	2.468	2.541	2.589	2.792	2.478	2.584	2.739	2.671	2.720	**2.63**
Standard	2.864	2.762	2.550	2.687	2.782	2.567	2.669	2.705	2.774	2.807	**2.72**
Repetition	2.232	2.065	2.027	2.131	2.340	2.152	2.294	2.370	2.190	2.280	**2.21**

Rem(I,3)=2

file	1	2	3	4	5	6	7	8	9	10	**Avg.**
With	2.877	3.019	2.850	2.863	2.725	2.887	2.726	2.893	2.865	2.983	**2.87**
Without	2.707	2.642	2.486	2.694	2.649	2.601	2.579	2.694	2.655	2.822	**2.65**
Standard	2.849	2.678	2.746	2.756	2.741	2.570	2.716	2.646	2.774	2.712	**2.72**
Repetition	2.261	1.957	1.969	2.293	2.361	2.131	2.210	2.388	2.269	2.318	**1.98**

Average:

file	1	2	3	4	5	6	7	8	9	10	**Avg.**
With	2.931	2.764	2.762	2.742	2.873	2.796	2.812	2.915	2.802	2.896	**2.83**
Without	2.715	2.58	2.485	2.619	2.689	2.542	2.601	2.689	2.589	2.729	**2.62**
Standard	2.85	2.781	2.586	2.67	2.748	2.691	2.73	2.643	2.71	2.673	**2.7**
Repetition	2.252	2	1.996	2.211	2.299	2.139	2.223	2.254	2.254	2.327	**2.13**

Male2:

Rem(I,3)=0

file	1	2	3	4	5	6	7	8	9	10	**Avg.**
With	2.743	2.851	2.803	2.881	2.934	2.751	3.017	2.910	2.894	2.823	**2.86**
Without	2.561	2.620	2.594	2.598	2.753	2.499	2.707	2.723	2.644	2.593	**2.63**
Standard	2.718	2.778	2.824	2.778	2.635	2.698	2.803	2.819	2.764	2.853	**2.77**
Repetition	2.101	2.216	2.095	1.925	2.294	1.873	2.086	2.228	2.178	2.001	**2.1**

Rem(I,3)=1

file	1	2	3	4	5	6	7	8	9	10	**Avg.**
With	2.762	2.832	2.868	2.922	2.817	2.817	2.992	2.886	2.837	2.877	**2.86**
Without	2.545	2.600	2.671	2.630	2.604	2.515	2.646	2.733	2.633	2.552	**2.61**
Standard	2.750	2.856	2.876	2.928	2.736	2.783	2.822	2.729	2.886	2.947	**2.83**
Repetition	2.225	2.234	2.064	1.977	2.323	1.874	2.173	2.259	2.174	2.012	**2.13**

Rem(I,3)=2

file	1	2	3	4	5	6	7	8	9	10	**Avg.**
With	2.744	2.874	2.918	2.845	2.861	2.712	2.878	2.798	2.835	2.794	**2.83**
Without	2.567	2.680	2.680	2.518	2.680	2.427	2.521	2.610	2.516	2.603	**2.58**
Standard	2.565	2.831	2.894	2.899	2.719	2.642	2.796	2.644	2.794	2.865	**2.76**
Repetition	2.164	2.176	2.037	1.938	2.282	1.903	2.224	2.269	2.129	2.005	**2.11**

Average:

file	1	2	3	4	5	6	7	8	9	10	**Avg.**
Prediction	2.75	2.852	2.863	2.883	2.871	2.76	2.963	2.865	2.855	2.831	**2.85**
Standard	2.678	2.822	2.865	2.865	2.7	2.708	2.807	2.807	2.815	2.89	**2.8**
Repetition	2.163	2.209	2.065	1.95	2.185	1.883	2.161	2.252	2.163	2.006	**2.1**

Female1:

Rem(I,3)=0

file	1	2	3	4	5	6	7	8	9	10	**Avg.**
With	2.775	2.616	2.635	2.790	2.850	2.745	2.869	2.683	2.912	2.746	**2.76**
Without	2.650	2.495	2.519	2.698	2.759	2.637	2.756	2.587	2.819	2.746	**2.67**
Standard	2.517	2.502	2.435	2.601	2.723	2.632	2.667	2.521	2.669	2.688	**2.6**
Repetition	2.022	1.949	1.891	1.984	1.931	1.919	2.019	2.115	2.045	1.924	**1.98**

Rem(I,3)=1

file	1	2	3	4	5	6	7	8	9	10	**Avg.**
With	2.728	2.813	2.785	2.747	2.726	2.679	2.773	2.818	2.841	2.712	**2.76**
Without	2.638	2.697	2.685	2.663	2.649	2.553	2.663	2.676	2.751	2.592	**2.66**
Standard	2.591	2.606	2.447	2.594	2.765	2.584	2.671	2.516	2.594	2.561	**2.59**
Repetition	2.029	1.893	1.880	2.032	1.899	1.943	2.037	2.041	2.031	1.895	**1.97**

Rem(I,3)=2

file	1	2	3	4	5	6	7	8	9	10	**Avg.**
With	2.661	2.698	2.648	2.771	2.866	2.642	2.821	2.782	2.897	2.831	**2.76**
Without	2.601	2.632	2.543	2.682	2.734	2.526	2.691	2.660	2.795	2.731	**2.66**
Standard	2.410	2.535	2.575	2.580	2.724	2.668	2.663	2.518	2.651	2.659	**2.6**
Repetition	2.017	1.936	1.883	1.983	1.845	1.967	1.984	2.052	2.024	1.851	**1.95**

Average:

file	1	2	3	4	5	6	7	8	9	10	**Avg.**
Prediction	2.721	2.709	2.689	2.769	2.814	2.689	2.821	2.761	2.883	2.763	**2.76**
Standard	2.506	2.548	2.486	2.592	2.737	2.628	2.667	2.518	2.638	2.636	**2.6**
Repetition	2.023	1.926	1.885	2	1.892	1.943	2.013	2.069	2.033	1.89	**1.97**

Female2:

Rem(I,3)=0

file	1	2	3	4	5	6	7	8	9	10	**Avg.**
With	2.669	2.825	2.767	2.702	2.875	2.760	2.570	2.605	2.679	2.639	**2.71**
Without	2.562	2.667	2.660	2.660	2.810	2.638	2.496	2.504	2.591	2.618	**2.62**
Standard	2.602	2.578	2.549	2.597	2.664	2.332	2.339	2.332	2.543	2.590	**2.51**
Repetition	1.692	1.828	1.939	1.955	2.015	1.827	1.485	1.831	1.750	1.920	**1.82**

Rem(I,3)=1

file	1	2	3	4	5	6	7	8	9	10	**Avg.**
With	2.596	2.712	2.614	2.644	2.747	2.631	2.511	2.639	2.828	2.608	**2.65**
Without	2.486	2.692	2.553	2.585	2.703	2.492	2.418	2.579	2.726	2.569	**2.58**
Standard	2.509	2.644	2.493	2.553	2.598	2.559	2.294	2.353	2.719	2.541	**2.53**
Repetition	1.739	1.939	1.965	1.991	2.075	1.939	1.673	1.824	1.702	1.930	**1.88**

Rem(I,3)=2

file	1	2	3	4	5	6	7	8	9	10	**Avg.**
With	2.512	2.609	2.653	2.696	2.883	2.459	2.471	2.535	2.726	2.651	**2.62**
Without	2.391	2.567	2.675	2.656	2.780	2.377	2.343	2.468	2.585	2.614	**2.55**
Standard	2.503	2.496	2.379	2.507	2.744	2.479	2.275	2.497	2.583	2.470	**2.49**
Repetition	1.724	1.914	1.973	1.924	2.063	1.815	1.570	1.925	1.821	1.927	**1.87**

Average:

file	1	2	3	4	5	6	7	8	9	10	**Avg.**
Prediction	2.592	2.715	2.678	2.681	2.853	2.617	2.517	2.592	2.744	2.633	**2.66**
Standard	2.55	2.538	2.573	2.474	2.552	2.669	2.46	2.303	2.394	2.615	**2.53**
Repetition	1.72	1.718	1.894	1.959	1.96	2.051	1.86	1.567	1.89	1.758	**1.93**

A-3: Results For 4th Phase

Male1:

Rem(I,3)=0

file	1	2	3	4	5	6	7	8	9	10	**Avg.**
Prediction	3.020	2.932	2.706	2.696	2.937	2.843	2.987	2.980	2.606	2.805	**2.85**
Standard	2.796	2.849	2.459	2.533	2.656	2.652	2.804	2.647	2.499	2.623	**2.65**
Repetition	2.264	1.988	1.993	2.210	2.195	2.135	2.166	2.336	2.304	2.383	**2.2**

Rem(I,3)=1

file	1	2	3	4	5	6	7	8	9	10	**Avg.**
Prediction	3.022	2.785	2.795	2.765	3.007	2.788	2.831	2.957	2.899	3.027	**2.89**
Standard	2.864	2.762	2.550	2.687	2.782	2.567	2.669	2.705	2.774	2.807	**2.72**
Repetition	2.232	2.065	2.027	2.131	2.340	2.152	2.294	2.370	2.190	2.280	**2.21**

Rem(I,3)=2

file	1	2	3	4	5	6	7	8	9	10	**Avg.**
Prediction	2.960	3.087	2.897	2.868	2.748	2.922	2.775	2.936	2.940	3.068	**2.92**
Standard	2.849	2.678	2.746	2.756	2.741	2.570	2.716	2.646	2.774	2.712	**2.72**
Repetition	2.261	1.957	1.969	2.293	2.361	2.131	2.210	2.388	2.269	2.318	**1.98**

Average:

file	1	2	3	4	5	6	7	8	9	10	**Avg**
Prediction	3.001	2.935	2.799	2.776	2.898	2.851	2.864	2.958	2.815	2.97	**2.89**
Standard	2.85	2.781	2.586	2.67	2.748	2.691	2.73	2.643	2.71	2.673	**2.7**
Repetition	2.252	2	1.996	2.211	2.299	2.139	2.223	2.254	2.254	2.327	**2.13**

Male2:

Rem(I,3)=0

file	1	2	3	4	5	6	7	8	9	10	**Avg.**
Prediction	2.780	2.903	2.864	2.928	2.969	2.800	3.033	2.943	2.908	2.880	**2.9**
Standard	2.718	2.778	2.824	2.778	2.635	2.698	2.803	2.819	2.764	2.853	**2.77**
Repetition	2.101	2.216	2.095	1.925	2.294	1.873	2.086	2.228	2.178	2.001	**2.1**

Rem(I,3)=1

file	1	2	3	4	5	6	7	8	9	10	**Avg.**
Prediction	2.802	2.907	2.904	2.996	2.847	2.869	3.023	2.942	2.882	2.971	**2.91**
Standard	2.750	2.856	2.876	2.928	2.736	2.783	2.822	2.729	2.886	2.947	**2.83**
Repetition	2.225	2.234	2.064	1.977	2.323	1.874	2.173	2.259	2.174	2.012	**2.13**

Rem(I,3)=2

file	1	2	3	4	5	6	7	8	9	10	**Avg.**
Prediction	2.776	2.961	2.993	2.878	2.886	2.765	2.909	2.843	2.913	2.857	**2.88**
Standard	2.565	2.831	2.894	2.899	2.719	2.642	2.796	2.644	2.794	2.865	**2.76**
Repetition	2.164	2.176	2.037	1.938	2.282	1.903	2.224	2.269	2.129	2.005	**2.11**

Average:

file	1	2	3	4	5	6	7	8	9	10	**Avg.**
Prediction	2.786	2.924	2.92	2.934	2.901	2.812	2.988	2.909	2.901	2.903	**2.9**
Standard	2.678	2.822	2.865	2.865	2.7	2.708	2.807	2.807	2.815	2.89	**2.8**
Repetition	2.163	2.209	2.065	1.95	2.185	1.883	2.161	2.252	2.163	2.006	**2.11**

Female1:

Rem(I,3)=0

file	1	2	3	4	5	6	7	8	9	10	**Avg.**
Prediction	2.785	2.622	2.631	2.810	2.850	2.753	2.902	2.691	2.917	2.760	**2.77**
Standard	2.517	2.502	2.435	2.601	2.723	2.632	2.667	2.521	2.669	2.688	**2.56**
Repetition	2.022	1.949	1.891	1.984	1.931	1.919	2.019	2.115	2.045	1.924	**1.98**

Rem(I,3)=1

file	1	2	3	4	5	6	7	8	9	10	**Avg.**
Prediction	2.740	2.807	2.805	2.750	2.747	2.687	2.780	2.830	2.854	2.716	**2.77**
Standard	2.591	2.606	2.447	2.594	2.765	2.584	2.671	2.516	2.594	2.561	**2.59**
Repetition	2.029	1.893	1.880	2.032	1.899	1.943	2.037	2.041	2.031	1.895	**1.97**

Rem(I,3)=2

file	1	2	3	4	5	6	7	8	9	10	**Avg.**
Prediction	2.663	2.698	2.657	2.776	2.870	2.656	2.835	2.794	2.907	2.841	**2.77**
Standard	2.410	2.535	2.575	2.580	2.724	2.668	2.663	2.518	2.651	2.659	**2.6**
Repetition	2.017	1.936	1.883	1.983	1.845	1.967	1.984	2.052	2.024	1.851	**1.95**

Average:

file	1	2	3	4	5	6	7	8	9	10	**Avg**
Prediction	2.729	2.709	2.698	2.779	2.822	2.699	2.839	2.772	2.893	2.773	**2.77**
Standard	2.506	2.548	2.486	2.592	2.737	2.628	2.667	2.518	2.638	2.636	**2.6**
Repetition	2.023	1.926	1.885	2	1.892	1.943	2.013	2.069	2.033	1.89	**2**

Female2:

Rem(I,3)=0

file	1	2	3	4	5	6	7	8	9	10	**Avg.**
Prediction	2.676	2.835	2.772	2.711	2.891	2.775	2.589	2.592	2.707	2.707	**2.73**
Standard	2.602	2.578	2.549	2.597	2.664	2.332	2.339	2.332	2.543	2.590	**2.51**
Repetition	1.692	1.828	1.939	1.955	2.015	1.827	1.485	1.831	1.750	1.920	**1.82**

Rem(I,3)=1

file	1	2	3	4	5	6	7	8	9	10	**Avg.**
Prediction	2.590	2.737	2.605	2.648	2.744	2.640	2.516	2.649	2.842	2.617	**2.66**
Standard	2.509	2.644	2.493	2.553	2.598	2.559	2.294	2.353	2.719	2.541	**2.53**
Repetition	1.739	1.939	1.965	1.991	2.075	1.939	1.673	1.824	1.702	1.930	**1.88**

Rem(I,3)=2

file	1	2	3	4	5	6	7	8	9	10	**Avg.**
Prediction	2.517	2.631	2.678	2.710	2.894	2.478	2.491	2.547	2.737	2.641	**2.63**
Standard	2.503	2.496	2.379	2.507	2.744	2.479	2.275	2.497	2.583	2.470	**2.49**
Repetition	1.724	1.914	1.973	1.924	2.063	1.815	1.570	1.925	1.821	1.927	**1.87**

Average:

file	1	2	3	4	5	6	7	8	9	10	**Avg.**
Prediction	2.594	2.734	2.685	2.69	2.843	2.631	2.532	2.596	2.762	2.655	**2.67**
Standard	2.538	2.573	2.474	2.552	2.669	2.46	2.303	2.394	2.615	2.534	**2.51**
Repetition	1.718	1.894	1.959	1.96	2.051	1.86	1.567	1.89	1.758	1.926	**1.86**

A-4: Results of 5th Phase

A-4-1: 50 160 Samples Autocorrelation Window

Male1:

rem(I,4)=0

	1	2	3	4	5	6	7	8	9	10	**Avg.**
Short	3.092	2.971	2.851	2.954	2.930	2.796	2.971	2.871	3.049	3.118	**2.96**
G.711.A	2.999	2.918	2.754	2.817	2.835	2.506	2.815	2.803	2.863	2.894	**2.82**

Rem(I,4)=1

	1	2	3	4	5	6	7	8	9	10	**Avg.**
Short	2.937	3.097	2.814	2.875	3.003	2.997	2.926	2.994	2.916	3.125	**2.97**
G.711.A	2.875	2.957	2.691	2.746	2.770	2.846	2.769	2.832	2.804	2.028	**2.73**

Rem(I,4)=2

	1	2	3	4	5	6	7	8	9	10	**Avg**
Short	2.981	2.929	2.868	2.706	2.903	2.858	2.937	3.118	2.842	2.827	**2.9**
G.711.A	2.972	2.879	2.682	2.741	2.683	2.740	2.933	2.940	2.702	2.725	**2.8**

Rem(I,4)=3

	1	2	3	4	5	6	7	8	9	10	**Avg.**
Short	3.107	3.016	2.919	2.933	3.172	2.943	2.885	3.165	2.948	3.018	**3.01**
G.711.A	2.917	3.104	2.702	2.775	3.001	2.822	2.833	2.977	2.724	2.804	**2.87**

Average:

	1	2	3	4	5	6	7	8	9	10	**Avg.**
Short	3.029	3.003	2.863	2.867	3.002	2.9	2.93	3.037	2.939	3.022	**2.96**
G.711.A	2.941	2.965	2.707	2.77	2.822	2.729	2.838	2.888	2.774	2.863	**2.83**

Male2:

rem(I,4)=0

	1	2	3	4	5	6	7	8	9	10	**Avg.**
Short	2.943	2.970	3.016	3.082	3.016	2.985	3.026	2.945	3.020	2.966	**3**
G.711A	2.807	2.966	2.9	2.969	2.845	2.931	2.909	2.801	2.831	2.961	**2.89**

Rem(I,4)=1

	1	2	3	4	5	6	7	8	9	10	**Avg.**
Short	3.034	2.935	2.894	2.876	2.954	2.777	3.035	2.937	2.909	2.922	**2.93**
G.711A	2.825	2.892	2.893	2.803	2.765	2.698	2.915	2.848	2.793	2.870	**2.83**

Rem(I,4)=2

	1	2	3	4	5	6	7	8	9	10	**Avg.**
Short	2.678	2.973	2.961	3.067	2.903	2.847	3.126	2.946	2.983	2.818	**2.93**
G.711A	2.523	2.906	2.863	2.979	2.702	2.876	2.945	2.763	2.965	2.794	**2.83**

Rem(I,4)=3

	1	2	3	4	5	6	7	8	9	10	**Avg.**
Short	2.850	2.896	2.949	2.913	3.034	2.800	3.071	3.072	2.924	3.025	**2.95**
G.711A	2.793	2.857	2.913	2.833	2.909	2.750	2.920	2.889	2.783	3.018	**2.87**

Average:

	1	2	3	4	5	6	7	8	9	10	**Avg.**
Short	2.876	2.944	2.955	2.985	2.977	2.852	3.065	2.975	2.959	2.933	**2.95**
G.711A	2.737	2.905	2.892	2.896	2.805	2.814	2.922	2.825	2.843	2.911	**2.86**

Female1:

rem(I,4)=0

	1	2	3	4	5	6	7	8	9	10	**Avg.**
Short	2.819	2.825	2.720	2.776	2.926	2.743	2.910	2.841	2.991	2.868	**2.81**
G.711A	2.649	2.647	2.548	2.560	2.695	2.503	2.739	2.527	2.704	2.645	**2.65**

Rem(I,4)=1

	1	2	3	4	5	6	7	8	9	10	**Avg.**
Short	2.849	2.797	2.804	2.884	2.920	2.799	2.919	2.867	3.027	2.820	**2.83**
G.711A	2.583	2.585	2.598	2.654	2.793	2.639	2.758	2.561	2.864	2.664	**2.65**

Rem(I,4)=2

	1	2	3	4	5	6	7	8	9	10	**Avg.**
Short	2.872	2.793	2.909	2.963	2.811	2.843	2.943	2.914	2.973	2.827	**2.84**
G.711A	2.633	2.611	2.563	2.730	2.660	2.632	2.809	2.69	2.748	2.719	**2.7**

Rem(I,4)=3

	1	2	3	4	5	6	7	8	9	10	**Avg.**
Short	2.789	2.764	2.753	2.871	3.001	2.748	2.877	2.818	2.897	2.874	**2.81**
G.711 A	2.5	2.583	2.615	2.602	2.837	2.644	2.664	2.577	2.627	2.682	**2.64**

Average:

	1	2	3	4	5	6	7	8	9	10	**Avg.**
Short	2.832	2.795	2.797	2.874	2.915	2.783	2.912	2.86	2.972	2.847	**2.86**
G.711 A	2.591	2.607	2.581	2.637	2.747	2.605	2.743	2.589	2.736	2.678	**2.65**

Female2:

rem(I,4)=0

	1	2	3	4	5	6	7	8	9	10	**Avg.**
Short	2.705	2.793	2.837	2.837	2.792	2.639	2.816	2.817	2.847	2.744	**2.78**
G.711A	2.549	2.590	2.636	2.563	2.584	2.407	2.641	2.572	2.789	2.514	**2.58**

Rem(I,4)=1

	1	2	3	4	5	6	7	8	9	10	**Avg.**
Short	2.818	2.782	2.783	2.796	2.968	2.843	2.584	2.586	2.835	2.918	**2.79**
G.711 A	2.655	2.689	2.609	2.597	2.782	2.609	2.346	2.350	2.708	2.697	**2.6**

Rem(I,4)=2

	1	2	3	4	5	6	7	8	9	10	**Avg.**
Short	2.773	2.738	2.720	2.705	3.030	2.752	2.576	2.669	2.830	2.697	**2.75**
G.711 A	2.708	2.625	2.603	2.561	2.835	2.609	2.261	2.431	2.598	2.636	**2.59**

Rem(I,4)=3

	1	2	3	4	5	6	7	8	9	10	**Avg.**
Short	2.584	2.916	2.773	2.804	2.896	2.699	2.609	2.704	2.786	2.659	**2.74**
G.711 A	2.358	2.760	2.604	2.556	2.630	2.481	2.290	2.452	2.666	2.485	**2.53**

Average:

	1	2	3	4	5	6	7	8	9	10	**Avg.**
Short	2.72	2.807	2.779	2.786	2.922	2.734	2.646	2.694	2.825	2.755	**2.77**
G.711 A	2.568	2.666	2.613	2.569	2.708	2.527	2.385	2.451	2.69	2.589	**2.6**

A-4-2: 240 Samples Autocorrelation Window

Score for 25 % loss (GGG B GGG B)

Male1:

rem(I,4)=0

	1	2	3	4	5	6	7	8	9	10	**Avg.**
Short	3.153	3.00	2.905	2.967	2.959	2.809	2.973	2.910	3.072	3.151	**2.99**
G.711A	2.999	2.918	2.754	2.817	2.835	2.506	2.815	2.803	2.863	2.894	**2.82**

Rem(I,4)=1

	1	2	3	4	5	6	7	8	9	10	**Avg.**
Short	2.952	3.100	2.824	2.876	3.011	3.008	2.941	3.029	2.856	3.160	**2.98**
G.711A	2.875	2.957	2.691	2.746	2.770	2.846	2.769	2.832	2.804	2.028	**2.73**

Rem(I,4)=2

	1	2	3	4	5	6	7	8	9	10	**Avg.**
Short	2.991	2.939	2.920	2.757	2.924	2.846	2.937	3.092	2.851	2.851	**2.91**
G.711 A	2.972	2.879	2.682	2.741	2.683	2.740	2.933	2.940	2.702	2.725	**2.8**

Rem(I,4)=3

	1	2	3	4	5	6	7	8	9	10	**Avg.**
Short	3.113	3.090	2.978	2.968	3.138	2.928	2.888	3.183	2.962	3.027	**3.03**
G.711A	2.917	3.104	2.702	2.775	3.001	2.822	2.833	2.977	2.724	2.804	**2.87**

Average:

	1	2	3	4	5	6	7	8	9	10	**Avg.**
Short	3.052	3.032	2.907	2.892	3.008	2.898	2.935	3.054	2.936	3.047	**2.98**
G.711 A	2.941	2.965	2.707	2.77	2.822	2.729	2.838	2.888	2.773	2.862	**2.83**

Score for 25 % loss (GGG B GGG B), Second set

Male2:

rem(I,4)=0

	1	2	3	4	5	6	7	8	9	10	**Avg.**
Short	2.964	3.004	3.035	3.076	3.003	3.001	3.030	2.994	3.007	2.977	**3.01**
G.711A	2.807	2.966	2.9	2.969	2.845	2.931	2.909	2.801	2.831	2.961	**2.89**

Rem(I,4)=1

	1	2	3	4	5	6	7	8	9	10	**Avg.**
Short	3.045	2.952	2.985	2.920	2.962	2.761	3.039	2.934	2.912	2.916	**2.94**
G.711A	2.825	2.892	2.893	2.803	2.765	2.698	2.915	2.848	2.793	2.870	**2.83**

Rem(I,4)=2

	1	2	3	4	5	6	7	8	9	10	**Avg.**
Short	2.667	2.947	2.968	3.074	2.905	2.865	3.160	2.976	2.945	2.820	**2.93**
G.711A	2.523	2.906	2.863	2.979	2.702	2.876	2.945	2.763	2.965	2.794	**2.83**

Rem(I,4)=3

	1	2	3	4	5	6	7	8	9	10	**Avg.**
Short	2.883	2.902	3.012	2.905	3.045	2.791	3.051	3.049	2.927	3.065	**2.96**
G.711 A	2.793	2.857	2.913	2.833	2.909	2.750	2.920	2.889	2.783	3.018	**2.87**

Average:

	1	2	3	4	5	6	7	8	9	10	**Avg.**
Short	2.89	2.951	3.00	2.994	2.979	2.855	3.07	2.988	2.948	2.945	**2.96**
G.711A	2.737	2.905	2.892	2.896	2.805	2.814	2.922	2.825	2.843	2.911	**2.86**

Female: 1st set

Rem(I,4)=0

	1	2	3	4	5	6	7	8	9	10	**Avg.**
Short	2.812	2.804	2.719	2.795	2.906	2.668	2.857	2.777	2.936	2.835	**2.81**
G.711A	2.649	2.647	2.548	2.560	2.695	2.503	2.739	2.527	2.704	2.645	**2.65**

Rem(I,4)=1

	1	2	3	4	5	6	7	8	9	10	**Avg.**
Short	2.762	2.750	2.818	2.835	2.967	2.765	2.887	2.795	2.976	2.783	**2.83**
G.711A	2.583	2.585	2.598	2.654	2.793	2.639	2.758	2.561	2.864	2.664	**2.65**

Rem(I,4)=2

	1	2	3	4	5	6	7	8	9	10	**Avg.**
Short	2.798	2.744	2.850	2.964	2.819	2.780	2.925	2.833	2.930	2.789	**2.84**
G.711 A	2.633	2.611	2.563	2.730	2.660	2.632	2.809	2.69	2.748	2.719	**2.7**

Rem(I,4)=3

	1	2	3	4	5	6	7	8	9	10	**Avg.**
Short	2.730	2.750	2.750	2.835	2.999	2.732	2.841	2.752	2.843	2.851	**2.81**
G.711A	2.5	2.583	2.615	2.602	2.837	2.644	2.664	2.577	2.627	2.682	**2.64**

Average:

	1	2	3	4	5	6	7	8	9	10	**Avg.**
Short	2.776	2.762	2.784	2.857	2.923	2.736	2.878	2.789	2.921	2.815	**2.82**
G.711 A	2.591	2.607	2.581	2.637	2.746	2.605	2.743	2.589	2.736	2.678	**2.65**

Female: 2nd set

rem(I,4)=0

	1	2	3	4	5	6	7	8	9	10	**Avg.**
Short	2.762	2.79	2.848	2.802	2.74	2.622	2.836	2.830	2.836	2.684	**2.78**
G.711A	2.549	2.590	2.636	2.563	2.584	2.407	2.641	2.572	2.789	2.514	**2.58**

Rem(I,4)=1

	1	2	3	4	5	6	7	8	9	10	**Avg.**
Short	2.818	2.842	2.764	2.783	2.929	2.857	2.575	2.547	2.841	2.904	**2.79**
G.711A	2.655	2.689	2.609	2.597	2.782	2.609	2.346	2.350	2.708	2.697	**2.6**

Rem(I,4)=2

	1	2	3	4	5	6	7	8	9	10	**Avg.**
Short	2.823	2.769	2.695	2.683	3	2.756	2.569	2.622	2.812	2.653	**2.74**
G.711A	2.708	2.625	2.603	2.561	2.835	2.609	2.261	2.431	2.598	2.636	**2.59**

Rem(I,4)=3

	1	2	3	4	5	6	7	8	9	10	**Avg.**
Short	2.578	2.903	2.721	2.756	2.889	2.662	2.615	2.656	2.789	2.657	**2.72**
G.711A	2.358	2.760	2.604	2.556	2.630	2.481	2.290	2.452	2.666	2.485	**2.53**

Average:

	1	2	3	4	5	6	7	8	9	10	**Avg.**
Short	2.745	2.826	2.757	2.756	2.89	2.724	2.649	2.664	2.82	2.745	**2.76**
G.711A	2.568	2.666	2.613	2.569	2.708	2.527	2.385	2.451	2.69	2.589	**2.58**

Appendix B: The Results of The New Algorithm

(Final Phase)

B-1: Results of The Single Repeated Packet Loss:

- **First Male:**

Case1:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	3.141	3.014	2.894	3.008	3.017	2.772	3.014	2.983	3.017	3.119	3.0
Last_Phase	3.132	2.975	2.925	2.972	3.005	2.794	3.021	2.944	3.099	3.176	3.0
Standard1	2.999	2.918	2.754	2.817	2.835	2.506	2.815	2.803	2.863	2.894	2.82

Case2:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	3.048	3.101	2.768	2.891	3.046	3.051	2.985	3.051	2.915	3.198	3.01
Last_Phase	2.967	3.138	2.778	2.898	3.038	3.032	2.989	3.059	2.881	3.171	2.99
Standard1	2.875	2.957	2.691	2.746	2.770	2.846	2.769	2.832	2.804	3.028	2.83

Case3:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	3.148	2.961	2.883	2.791	2.877	2.959	3.087	3.159	2.813	2.930	2.96
Last_Phase	2.995	2.939	2.913	2.780	2.914	2.879	2.963	3.125	2.863	2.879	2.93
Standard1	2.972	2.879	2.682	2.741	2.683	2.740	2.933	2.940	2.702	2.752	2.80

Case4:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	3.133	3.170	2.884	3.002	3.255	2.919	2.984	3.207	2.949	3.033	3.05
Last_Phase	3.112	3.134	2.955	2.994	3.186	2.908	2.914	3.196	2.970	3.042	3.04
Standard1	2.917	3.104	2.702	2.775	3.001	2.822	2.833	2.977	2.724	2.804	2.87

Average:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	3.12	3.062	2.857	2.923	3.049	2.925	3.018	3.1	2.9215	3.07	3.01
Last_Phase	3.052	3.047	2.892	2.911	3.036	2.903	2.972	3.081	2.953	3.057	2.99
Standard1	2.941	2.965	2.707	2.77	2.822	2.728	2.838	2.888	2.773	2.87	2.83

- **Second Male:**

Case1:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	3.058	3.039	3.103	3.103	3.038	3.064	3.083	2.949	3.017	3.029	3.05
Last_Phase	2.987	3.020	3.065	3.090	3.005	3.030	3.051	2.989	3.059	2.989	3.03
Standard1	2.807	2.966	2.9	2.969	2.845	2.931	2.909	2.801	2.831	2.961	2.89

Case2:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	3.116	3.030	3.012	2.920	2.938	2.833	3.102	3.013	2.937	2.958	2.99
Last_Phase	3.087	2.979	3.008	2.940	2.969	2.784	3.076	2.962	2.924	2.941	2.97
Standard1	2.825	2.892	2.893	2.803	2.765	2.698	2.915	2.848	2.793	2.870	2.83

Case3:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	2.735	3.993	2.985	3.110	2.875	2.933	3.133	2.999	3.040	2.854	3.07
Last_Phase	2.692	2.955	2.943	3.106	2.931	2.869	3.147	2.981	2.976	2.844	2.94
Standard1	2.523	2.907	2.863	2.979	2.702	2.876	2.945	2.763	2.965	2.794	2.83

Case4:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	2.924	2.959	3.079	2.923	3.108	2.841	3.063	3.090	2.952	3.078	3.00
Last_Phase	2.906	2.910	3.038	2.933	3.002	2.817	3.082	3.112	2.946	3.061	2.98
Standrard1	2.793	2.857	2.913	2.833	2.909	2.750	2.920	2.889	2.783	3.018	2.86

Average:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	2.958	3.005	3.045	3.014	2.918	3.095	3.013	2.987	2.980	3.078	3.01
Last_Phase	2.918	2.966	3.014	3.017	2.875	3.089	3.011	2.976	2.960	3.061	2.99
Standrard1	2.737	2.906	2.892	2.896	2.813	2.922	2.920	2.843	2.918	3.018	2.89

- **First Female:**

Case1:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	2.819	2.811	2.717	2.786	2.871	2.641	2.867	2.758	2.877	2.825	2.8
Last_Phase	2.818	2.824	2.759	2.816	2.915	2.640	2.870	2.801	2.932	2.865	2.82
Standard1	2.649	2.647	2.548	2.560	2.695	2.503	2.739	2.527	2.704	2.645	2.65

Case2:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	2.729	2.711	2.740	2.824	2.941	2.771	2.878	2.772	3.039	2.796	2.82
Last_Phase	2.768	2.718	2.809	2.854	2.951	2.774	2.885	2.820	3.013	2.815	2.84
Standard1	2.583	2.585	2.598	2.654	2.793	2.639	2.758	2.561	2.864	2.664	2.65

Case3:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	2.780	2.766	2.816	2.966	2.772	2.763	2.935	2.836	2.901	2.806	2.83
Last_Phase	2.815	2.779	2.854	2.973	2.814	2.793	2.928	2.857	2.930	2.801	2.85
Standard1	2.633	2.611	2.563	2.730	2.660	2.632	2.809	2.69	2.748	2.719	2.7

Case4:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	2.717	2.758	2.745	2.790	2.981	2.722	2.832	2.730	2.833	2.863	2.8
Last_Phase	2.752	2.777	2.773	2.848	3.023	2.741	2.853	2.782	2.862	2.882	2.83
Standard1	2.5	2.583	2.615	2.602	2.837	2.644	2.664	2.577	2.627	2.682	2.63

Average:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	2.761	2.762	2.755	2.742	2.891	2.724	2.878	2.774	2.913	2.822	2.8
Last_Phase	2.78	2.775	2.799	2.873	2.925	2.737	2.884	2.814	2.934	2.840	2.83
Standard1	2.591	2.607	2.581	2.637	2.746	2.605	2.743	2.589	2.736	2.678	2.65

- **Second Female**:

Case1:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	2.718	2.787	2.848	2.771	2.722	2.590	2.852	2.788	2.864	2.684	2.76
Last_Phase	2.718	2.774	2.879	2.816	2.711	2.630	2.837	2.846	2.846	2.693	2.78
Standard1	2.549	2.590	2.636	2.563	2.584	2.407	2.641	2.572	2.789	2.514	2.58

Case2:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	2.801	2.850	2.789	2.809	2.876	2.825	2.558	2.558	2.836	2.906	2.78
Last_Phase	2.801	2.811	2.806	2.806	2.860	2.851	2.587	2.568	2.830	2.910	2.78
Standard1	2.655	2.689	2.609	2.597	2.782	2.609	2.346	2.350	2.708	2.697	2.6

Case3:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	2.817	2.784	2.735	2.692	2.970	2.787	2.558	2.576	2.814	2.723	2.75
Last_Phase	2.810	2.765	2.723	2.697	2.967	2.759	2.590	2.653	2.803	2.656	2.74
Standard1	2.708	2.625	2.603	2.561	2.835	2.609	2.261	2.431	2.598	2.636	2.59

Case4:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	2.582	2.874	2.785	2.730	2.843	2.644	2.589	2.655	2.804	2.652	2.72
Last_Phase	2.570	2.886	2.756	2.777	2.858	2.676	2.642	2.658	2.763	2.661	2.72
Standard1	2.358	2.760	2.604	2.556	2.630	2.481	2.290	2.452	2.666	2.485	2.53

Average:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	2.730	2.824	2.789	2.751	2.853	2.712	2.639	2.644	2.830	2.741	2.75
Last_Phase	2.720	2.809	2.791	2.774	2.849	2.729	2.664	2.681	2.811	2.73	2.75
Standard1	2.568	2.666	2.613	2.57	2.708	2.527	2.385	2.451	2.69	2.583	2.58

B-2: Results for Double Repeated Packet Loss

- **First Male:**

Case1:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	2.559	2.420	2.128	2.225	2.242	2.248	2.379	2.651	2.459	2.259	2.36
Last_Phase	2.388	2.300	1.993	2.066	2.058	2.046	2.160	2.382	2.353	2.191	2.19
Standard1	2.516	2.466	2.018	2.128	2.374	2.066	2.159	2.365	2.142	2.149	2.24

Case2:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	2.557	2.486	2.371	2.199	2.276	2.246	2.146	2.241	2.150	2.384	2.31
Last_Phase	2.366	2.386	2.188	2.048	2.152	2.078	1.946	2.171	2.047	2.282	2.17
Standard1	2.443	2.448	2.239	2.081	2.183	2.203	2.093	2.188	2.221	2.221	2.23

Case3:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	2.349	2.458	2.240	2.311	2.591	2.216	2.259	2.305	2.040	2.412	2.32
Last_Phase	2.231	2.279	2.044	2.153	2.477	2.117	2.110	2.124	1.807	2.224	2.16
Standard1	2.197	2.415	2.240	2.228	2.180	2.255	2.262	2.209	2.031	2.301	2.23

Case4:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	2.470	2.416	2.522	2.400	2.417	2.213	2.370	2.282	2.239	2.517	2.38
Last_Phase	2.275	2.218	2.291	2.207	2.314	2.037	2.179	2.126	2.105	2.258	2.20
Standard1	2.317	2.466	2.187	2.348	2.315	2.164	2.168	2.179	2.298	2.333	2.28

Average:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	2.484	2.445	2.315	2.284	2.382	2.231	2.289	2.37	2.222	2.394	2.34
Last_Phase	2.315	2.296	2.129	2.119	2.251	2.07	2.1	2.201	2.078	2.239	2.18
Standard1	2.368	2.393	2.171	2.196	2.263	2.172	2.171	2.235	2.173	2.251	2.24

- **Second Male**

Case1:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	1.863	2433	2.541	2.497	2.530	2.125	2.242	2.182	2.462	2.581	2.35
Last_Phase	1.781	2.268	2.420	2.407	2.322	2.084	2.144	2.141	2.270	2.468	2.23
Standard1	1.845	2.206	2.368	2.258	2.394	2.106	2.190	2.029	2.287	2.622	2.23

Case2:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	2.427	2.438	2.532	2.424	2.366	2.437	2.447	2.485	2.511	2.342	2.44
Last_Phase	2.256	2.349	2.369	2.375	2.279	2.357	2.350	2.305	2.418	2.241	2.33
Standard1	2.310	2.246	2.399	2.375	2.156	2.380	2.326	2.263	2.425	2.414	2.33

Case3:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	2.387	2.502	2.428	2.525	2.225	2.528	2.530	2.415	2.545	2.374	2.45
Last_Phase	2.307	2.340	2.376	2.446	2.133	2.444	2.247	2.203	2.418	2.249	2.32
Standard1	2.207	2.366	2.491	2.458	2.291	2.448	2.367	2.306	2.451	2.310	2.37

Case4:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	2.171	2.533	2.568	2.437	2.332	2.380	2.603	2.531	2.215	2.545	2.43
Last_Phase	2.028	2.435	2.377	2.366	2.216	2.260	2.437	2.410	2.143	2.401	2.31
Standard1	2.182	2.410	2.494	2.405	2.166	2.443	2.507	2.419	2.121	2.484	2.36

Average:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	2.212	2.477	2.517	2.471	2.363	2.368	2.456	2.403	2.433	2.461	2.42
Last_Phase	2.093	2.348	2.386	2.399	2.238	2.286	2.295	2.265	2.312	2.340	2.3
Standard1	2.136	2.307	2.438	2.374	2.252	2.344	2.348	2.254	2.321	2.458	2.32

- **First Female**

Case1:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	2.190	2.162	2.185	2.326	2.281	2.195	2.434	2.201	2.429	2.315	2.27
Last_Phase	2.153	2.011	2.117	2.428	2.242	2.167	2.374	2.187	2.403	2.280	2.24
Standard1	2.123	2.024	2.187	2.409	2.121	2.136	2.383	2.096	2.363	2.189	2.20

Case2:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	2.289	2.203	2.372	2.152	2.320	2.165	2.343	2.107	2.428	2.245	2.26
Last_Phase	2.239	2.193	2.340	2.125	2.261	2.112	2.211	2.083	2.400	2.182	2.21
Standard1	2.263	2.207	2.218	2.242	2.357	2.115	2.387	2.114	2.400	2.210	2.25

Case3:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	2.329	2.304	2.267	2.358	2.342	2.185	2.432	2.217	2.288	2.230	2.3
Last_Phase	2.272	2.260	2.156	2.345	2.326	2.182	2.385	2.228	2.223	2.142	2.25
Standard1	2.319	2.220	2.295	2.223	2.378	2.198	2.348	2.110	2.385	2.204	2.27

Case4:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	2.290	2.407	2.143	2.241	2.577	2.282	2.421	2.291	2.366	2.319	2.33
Last_Phase	2.280	2.372	2.109	2.264	2.551	2.230	2.392	2.251	2.311	2.286	2.30
Standard1	2.182	2.293	2.201	2.154	2.472	2.205	2.282	2.242	2.399	2.269	2.27

Average:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	2.275	2.269	2.242	2.269	2.38	2.207	2.408	2.204	2.378	2.278	2.29
Last_Phase	2.236	2.209	2.181	2.291	2.345	2.173	2.341	2.187	2.334	2.223	2.25
Standard1	2.222	2.186	2.225	2.257	2.332	2.164	2.35	2.141	2.387	2.218	2.25

- **Second Female**

Case1:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	2.083	2.242	2.092	2.280	2.318	2.020	1.850	2.138	2.309	1.959	2.13
Last_Phase	2.058	2.126	2.055	2.304	2.253	2.044	1.784	2.057	2.222	1.899	2.08
Standard1	2.002	2.264	2.012	2.168	2.171	2.096	1.681	2.048	2.238	1.897	2.06

Case2:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	2.025	2.227	2.037	2.065	2.366	2.271	1.804	1.973	2.283	2.253	2.13
Last_Phase	2.016	2.118	1.950	2.032	2.256	2.082	1.710	2.006	2.170	2.282	2.06
Standard1	1.899	2.033	1.927	2.028	2.366	2.149	1.783	2.064	2.274	2.058	2.06

Case3:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	1.962	2.168	2.120	2.222	2.304	1.949	1.921	2.222	2.353	2.054	2.13
Last_Phase	1.886	2.143	1.973	2.152	2.181	1.905	1.885	2.202	2.314	2.045	2.07
Standard1	1.886	2.062	2.094	2.240	2.194	1.938	1.887	2.018	2.285	2.109	2.07

Case4:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	2.085	2.128	2.212	2.128	2.180	2.110	2.042	2.207	2.430	2.185	2.17
Last_Phase	1.985	2.111	2.110	2.044	2.084	2.029	2.035	2.165	2.355	2.138	2.11
Standard1	2.017	2.058	2.119	2.078	2.163	2.042	1.881	2.133	2.413	2.008	2.09

Average:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	2.039	2.191	2.115	2.174	2.292	2.088	1.904	2.135	2.344	2.113	2.14
Last_Phase	1.986	2.125	2.022	2.133	2.194	2.015	1.854	2.108	2.265	2.091	2.08
Standard1	1.951	2.104	2.038	2.129	2.224	2.056	1.808	2.066	2.303	2.018	2.07

B-3: Results for Modified Loss Pattern

- **First Male**

Case1_modified:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	3.193	3.243	2.987	3.121	3.150	2.856	3.091	3.032	3.212	3.217	3.11
Standard1	3.055	3.097	2.813	2.916	2.953	2.561	2.952	2.843	3.020	3.026	2.92
Standard2	2.852	2.786	2.746	2.903	2.886	2.625	2.797	2.802	2.872	2.947	2.82

Case2_modified:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	3.232	3.184	2.954	2.982	3.231	3.097	3.087	3.079	3.038	3.328	3.12
Standard1	3.028	3.042	2.792	2.868	2.922	2.889	2.814	2.864	2.868	3.216	2.93
Standard2	2.827	2.861	2.687	2.761	2.785	2.818	2.731	2.864	2.672	2.946	2.8

Case3_modified:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	3.262	3.033	2.959	2.883	2.976	3.085	3.147	3.219	2.998	2.993	3.06
Standard1	3.089	2.908	2.728	2.833	2.754	2.881	2.937	3.007	2.871	2.765	2.88
Standard2	2.891	2.780	2.838	2.705	2.751	2.738	2.831	2.874	2.583	2.770	2.78

Case4_modified:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	3.231	3.221	3.066	3.094	3.306	3.022	3.037	3.266	3.144	3.123	3.15
Standard1	3.00	3.164	2.789	2.885	3.073	2.948	2.882	3.043	2.881	2.879	2.95
Standard2	2.804	2.868	2.760	2.617	3.050	2.747	2.726	2.849	2.816	2.839	2.81

Average:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	3.23	3.17	2.992	3.02	3.166	3.015	3.091	3.149	3.098	3.165	3.11
Standard1	3.043	3.053	2.781	2.876	2.926	2.820	2.896	3.94	2.91	2.97	3.02
Standard2	2.85	2.824	2.758	2.75	2.868	2.732	2.77	2.85	2.736	2.876	2.81

- **Second Male**

Case1_modified:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	3.184	3.216	3.225	3.253	3.233	3.141	3.249	3.130	3.151	3.149	3.19
Standard1	2.914	3.081	3.002	3.087	3.038	2.977	3.027	2.972	2.907	3.038	3
Standard2	2.791	2.906	2.842	2.863	2.766	2.754	2.792	2.869	2.928	2.667	2.82

Case2_modified:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	3.172	3.297	3.174	3.056	3.239	2.894	3.239	3.251	3.133	3.073	3.15
Standard1	2.882	3.080	2.976	2.923	3.045	2.743	2.982	3.049	2.961	2.983	2.96
Standard2	2.708	2.849	2.837	2.879	2.899	2.523	2.895	2.825	2.876	2.703	2.8

Case3_modified:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	2.836	3.206	3.088	3.174	3.057	3.007	3.450	3.083	3.155	2.953	3.1
Standard1	2.605	3.107	2.956	3.053	2.879	2.957	3.184	2.829	3.039	2.839	2.94
Standard2	2.485	2.917	2.756	2.867	2.797	2.686	2.923	2.847	2.769	2.694	2.78

Case4_modified:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	2.968	3.125	3.165	3.039	3.408	2.959	3.213	3.244	3.092	3.244	3.15
Standard1	2.815	2.972	2.977	2.948	3.143	2.840	3.056	3.033	2.943	3.131	2.99
Standard2	2.785	2.795	2.912	2.826	2.914	2.794	2.942	2.873	2.677	2.874	2.84

Average

	1	2	3	4	5	6	7	8	9	10	Avg.
New	3.04	3.211	3.163	3.131	3.23	3.00	3.288	3.177	3.133	3.105	3.15
Standard1	2.804	3.06	2.978	3.00	3.03	2.88	3.06	2.971	2.963	3.00	2.98
Standard2	2.693	2.87	2.837	2.860	2.844	2.69	2.888	2.854	2.813	2.74	2.81

- **First Female**

Case1_modified:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	3.012	2.854	2.914	2.949	3.033	2.822	3.076	3.022	3.113	2.996	2.98
Standard1	2.810	2.707	2.737	2.695	2.849	2.663	2.945	2.782	2.983	2.835	2.80
Standard2	2.747	2.743	2.661	2.737	2.882	2.716	2.929	2.833	2.979	2.888	2.81

Case2_modified:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	2.927	2.895	2.999	2.910	3.001	2.916	3.007	3.072	3.309	2.887	2.99
Standard1	2.766	2.728	2.807	2.742	2.859	2.771	2.887	2.834	3.118	2.770	2.83
Standard2	2.762	2.704	2.806	2.784	2.790	2.724	2.879	2.709	3.063	2.682	2.79

Case3_modified:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	2.971	2.947	2.969	3.094	2.842	2.889	3.072	3.042	3.080	2.863	2.98
Standard1	2.812	2.840	2.693	2.839	2.731	2.739	2.948	2.892	2.935	2.788	2.82
Standard2	2.824	2.730	2.815	2.819	2.719	2.791	2.827	2.836	2.939	2.739	2.80

Case4_modified:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	2.865	2.847	2.861	2.932	3.119	2.810	3.005	2.901	3.077	3.012	2.94
Standard1	2.695	2.660	2.740	2.755	2.972	2.733	2.841	2.746	2.879	2.883	2.79
Standard2	2.593	2.691	2.646	2.774	2.816	2.682	2.811	2.771	2.763	2.860	2.74

Average:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	2.944	2.884	2.936	2.971	2.999	2.859	3.04	3.009	3.145	2.94	2.97
Standard1	2.771	2.734	2.744	2.758	2.853	2.727	2.905	2.814	2.979	2.819	2.81
Standard2	2.732	2.717	2.732	2.779	2.802	2.728	2.862	2.787	2.937	2.8	2.79

- **Second Female**

Case1_modified:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	2.752	2.794	2.909	2.995	2.873	2.756	2.863	2.881	2.920	2.829	2.86
Standard1	2.563	2.595	2.685	2.787	2.746	2.567	2.653	2.655	2.852	2.640	2.67
Standard2	2.532	2.694	2.689	2.822	2.647	2.571	2.629	2.614	2.840	2.607	2.66

Case2_modified:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	2.841	2.885	2.897	2.906	2.973	2.916	2.599	2.601	2.981	2.993	2.86
Standard1	2.666	2.704	2.717	2.689	2.798	2.752	2.402	2.387	2.827	2.762	2.67
Standard2	2.603	2.610	2.733	2.783	2.805	2.682	2.283	2.328	2.759	2.677	2.63

Case3_modified:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	2.832	2.800	2.956	2.794	3.096	2.875	2.590	2.718	2.884	2.809	2.84
Standard1	2.719	2.656	2.832	2.629	2.913	2.685	2.296	2.576	2.621	2.736	2.67
Standard2	2.464	2.662	2.798	2.583	2.892	2.447	2.255	2.517	2.685	2.549	2.59

Case4_modified:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	2.587	2.898	2.819	2.927	2.916	2.780	2.668	2.669	2.875	2.735	2.79
Standard1	2.358	2.818	2.643	2.736	2.697	2.607	2.383	2.456	2.737	2.551	2.6
Standard2	2.328	2.861	2.718	2.762	2.745	2.443	2.452	2.408	2.573	2.538	2.59

Average:

	1	2	3	4	5	6	7	8	9	10	Avg.
New	2.753	2.844	2.895	2.906	2.965	2.832	2.68	2.717	2.915	2.842	2.83
Standard1	2.577	2.693	2.719	2.71	2.789	2.653	2.434	2.519	2.759	2.672	2.65
Standard2	2.482	2.707	2.735	2.738	2.772	2.536	2.405	2.467	2.714	2.593	2.61

B-4: Results for Random Loss Tests

B-4-1 Results for 5% packet loss

- **First Male**

	1	2	3	4	5	6	7	8	9	10	Avg.
New	3.675	3.638	3.619	3.715	3.786	3.617	3.723	3.665	3.471	3.692	3.66
Standard1	3.488	3.675	3.345	3.475	3.542	3.541	3.719	3.394	3.380	3.289	3.48

- **Second Male**

	1	2	3	4	5	6	7	8	9	10	Avg.
New	3.424	3.516	3.620	3.518	3.379	3.326	3.757	3.618	3.550	3.380	3.51
Standard1	3.390	3.481	3.569	3.409	3.277	3.333	3.567	3.300	3.412	3.282	3.4

- **First Female**

	1	2	3	4	5	6	7	8	9	10	Avg.
New	3.604	3.487	3.307	3.261	3.238	3.247	3.519	3.509	3.620	3.453	3.42
Standard1	3.436	3.449	3.173	3.219	3.186	3.117	3.447	3.404	3.468	3.371	3.33

- **Second Female**

	1	2	3	4	5	6	7	8	9	10	Avg.
New	3.612	3.519	3.347	3.372	3.714	3.182	3.414	3.254	3.550	3.452	3.44
Standard1	3.391	3.324	3.156	3.276	3.539	2.945	3.254	3.244	3.417	3.394	3.29

B-4-2: Results for 10% packet loss

- **First Male**

	1	2	3	4	5	6	7	8	9	10	Avg.
New	3.255	3.222	3.125	3.247	3.101	3.112	3.285	3.290	3.379	2.917	3.19
Standard1	3.079	3.076	3.049	3.086	3.033	3.071	3.206	3.148	3.081	2.947	3.08

- **Second Male**

	1	2	3	4	5	6	7	8	9	10	Avg.
New	3.429	3.343	3.286	3.323	3.119	3.187	3.462	3.218	3.160	3.171	3.27
Standard1	3.270	3.128	3.169	3.214	2.983	3.155	3.247	2.998	3.064	3.088	3.13

- **First Female**

	1	2	3	4	5	6	7	8	9	10	Avg.
New	3.136	3.089	2.853	2.940	3.248	2.956	3.072	3.136	3.186	2.910	3.05
Standard1	2.977	2.944	2.822	2.881	3.173	2.839	2.977	2.960	3.031	2.778	2.94

- **Second Female**

	1	2	3	4	5	6	7	8	9	10	Avg.
New	2.974	3.163	3.082	2.839	3.040	3.026	2.967	2.891	3.111	3.036	3.01
Standard1	2.803	2.959	2.966	2.643	2.891	2.866	2.769	2.792	3.055	2.810	2.85

B-4-3: Results for 25 % Random Packet Loss

- **First Male**

	1	2	3	4	5	6	7	8	9	10	Avg.
New	2.736	2.806	2.517	2.623	2.996	2.711	2.839	2.784	2.773	2.636	2.74
Standard1	2.641	2.734	2.524	2.500	2.698	2.644	2.549	2.779	2.579	2.399	2.60

- **Second Male**

	1	2	3	4	5	6	7	8	9	10	Avg.
New	2.668	2.788	2.935	2.808	3.012	2.766	2.693	2.736	2.725	2.749	2.79
Standard1	2.399	2.665	2.793	2.721	2.804	2.701	2.615	2.526	2.621	2.648	2.65

- **First Female**

	1	2	3	4	5	6	7	8	9	10	Avg.
New	2.616	2.479	2.569	2.700	2.780	2.676	2.696	2.535	2.844	2.704	2.66
Standard1	2.461	2.401	2.520	2.559	2.695	2.581	2.622	2.494	2.734	2.529	2.56

- **Second Female**

	1	2	3	4	5	6	7	8	9	10	Avg.
New	2.479	2.678	2.555	2.259	2.846	2.643	2.396	2.600	2.899	2.612	2.6
Standard1	2.349	2.516	2.502	2.161	2.687	2.454	2.200	2.470	2.787	2.382	2.45

B-4-6 Results for 50% Random Packet Loss

- **First Male**

	1	2	3	4	5	6	7	8	9	10	Avg.
New	1.774	1.629	1.591	1.893	1.825	1.581	1.761	2.079	1.930	1.692	1.78
Standard1	1.798	1.551	1.616	1.831	1.730	1.421	1.767	1.990	1.691	1.751	1.71

- **Second Male**

	1	2	3	4	5	6	7	8	9	10	Avg.
New	1.691	1.817	2.058	1.904	1.730	1.843	1.846	1.777	1.777	2.047	1.85
Standard1	1.674	1.973	2.114	1.989	1.656	1.838	1.952	1.853	1.800	1.958	1.88

- **First Female**

	1	2	3	4	5	6	7	8	9	10	Avg.
New	1.700	1.445	1.826	1.564	1.982	1.783	1.901	1.766	1.834	1.772	1.76
Standard1	1.792	1.686	1.772	1.455	1.988	1.703	1.834	1.712	1.766	1.649	1.74

- **Second Female**

	1	2	3	4	5	6	7	8	9	10	Avg.
New	1.293	1.470	1.473	1.657	1.728	1.529	1.500	1.373	1.809	1.609	1.54
Standard1	1.348	1.683	1.583	1.468	1.566	1.574	1.407	1.368	1.835	1.662	1.54