NOTE TO USERS

Parametric Mixing for Centralized VoIP Conferencing Using ITU-T Recommendation G.722.2

submitted by

Giuseppe Agnello, B. Eng.

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

Master of Applied Science

Ottawa-Carleton Institute for Electrical and Computer Engineering
Faculty of Engineering
Department of Systems and Computer Engineering

Carleton University
Ottawa, Ontario, Canada, K1S 5B6
January 2006

Your file Votre référence
ISBN: 0-494-13433-X
Our file Notre référence
ISBN: 0-494-13433-X

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

Parametric Mixing for Centralized VoIP Conferencing Using ITU-T Recommendation G.722.2

submitted by

Giuseppe Agnello, B. Eng.

In partial fulfillment of the requirements for the degree of
Master of Applied Science

Thesis Supervisor

Prof. Richard M. Dansereau

Chair, Department of Systems and Computer Engineering

Prof. Rafik A. Goubran

Carleton University

Abstract

Excessive end to end delay or latency in a VoIP network is one of the major drawbacks of packetized voice communication. The two main components of end to end delay are codec processing and propagation delay. Codec processing delay becomes a significant factor in centralized VoIP conferencing where voice packets, originating from the participants, are sent to a central unit or bridge to be combined using tandem mixing.

This thesis investigates a novel way based on the G.722.2 codec of mixing the packets at the central unit that reduces algorithmic complexity and therefore delay. The parameters used to represent the speech, LPCs, pitch lags, fixed codebook, and gains, are extracted from the encoded bit stream, mixed, and re-encoded instead of full decoding, mixing, and then re-encoding of the speech signals.

This parametric mixing reduces the bridge complexity by up to 85 % while still retaining acceptable speech quality, 3.7 MOS on average, as shown by simulations.

Acknowledgements

First of all, my most sincere thanks goes out to my supervisor, Professor Richard M. Dansereau, for his guidance and support. His passion towards his work and dedication to his students are truly remarkable.

Many thanks to Professor Mohamed El-Tanany for being there when I needed a hand.

I would also like to express my gratitude to the National Capital Institute of Telecommunications (NCIT) and Carleton University for their financial support during my graduate studies.

All of the staff in the administration office were always available to help. In particular, special thanks to graduate assistant, Blazenka Power.

Finally, I would like to thank my family for their encouragements, in particular, Kerry McVey, for her boundless support.

Table of Contents

List of Figures

List of Tables

List of Acronyms

3GPP	3rd Generation Partnership Project
ACELP	Algebraic CELP
ACR	Absolute Category Rating
ADPCM	Adaptive Delta PCM
AMR	Adaptive Multi Rate
AMR-WB	Adaptive Multi-Rate Wideband
ATM	Asynchronous Transfer Mode
CELP	Code Excited Linear Prediction
CNAME	Canonical Name
DCR	Degradation Category Rating
DSLA	Digital Speech Level Analyzer
DSP	Digital Signal Processing
DT	Discrete Time
FCFS	First Come First Served

FIR	Finite Impulse Response
GK	Gatekeeper
GW	Gateway
IETF	Internet Engineering Task Force
IIR	Infinite Impulse Response
IP	Internet Protocol
IPSEC	IP Security
ISF	Immittance Spectral Frequency
ISP	Immittance Spectral Pair
ITU	International Telecommunication Union
ITU-T	ITU standards for Telecommunications
LE	Listening Effort
LP	Linear Prediction
LPC	Linear Prediction Coefficient
LQ	Listening Quality
LSP	Line Spectral Pair
LTP	Long Term Predictor
MCU	Multipoint Control Unit
MEGACO	Media Gateway Controller
MGCP	Media Gateway Control Protocol
MIPS	Millions of Instructions per Second
MOPS	Millions of Operations per Second
MOS	Mean Opinion Score

MSE	Mean Square Error
MTT1	Multi Talk Type 1
MTT2	Multi Talk Type 2
MTT3	Multi Talk Type 3
MTT4	Multi Talk Type 4
PAMS	Perceptual Analysis Measurement System
PCM	Pulse Code Modulation
PESQ	Perceptual Evaluation of Speech Quality
PSQM	Perceptual Speech Quality Measurement
PSTN	Public Switched Telephone Network
QoS	Quality of Service
RAM	Random Access Memory
RAS	Registration Admission and Status
ROM	Read Only Memory
RTCP	Real Time Control Protocol
RTP	Real-time Transport Protocol
SCTP	Stream Control Transmission Protocol
SD	Standard Deviation
SDP	Session Description Protocol
SIP	Session Initiation Protocol
TCP	Transmission Control Protocol
TE	Terminal Endpoints
TFC	Tandem Free Conferencing

UDP	User Datagram Protocol
VAD	Voice Activity Detector
VoIP	Voice over Internet Protocol
WAN	Wide Area Network
WMOPS	Weighted MOPS
Yle	Listening Effort Score
Ylq	Listening Quality Score

1 Introduction

Voice transmission using the Internet as the backbone communication network has become more and more popular in recent years, gradually replacing the traditional way of transmitting voice traffic over the Public Switched Telephone Network (PSTN). In Voice over Internet Protocol (VoIP) the speech is sampled, compressed, packetized, and transmitted over the network as IP packets. This enables service providers to integrate a vast variety of multimedia traffic such as voice, data, video, etc., on the same digital network thus gaining significant financial savings.

1.1 Overview

VoIP is used in many applications. Some applications such as voice messaging and audio streaming can tolerate some delays while applications such as real-time telephony are not as forgiving.

This thesis concentrates on the latter class of applications by focusing on the mixing aspects of centralized VoIP conferencing.

Traditional voice conferencing on the PSTN is achieved by establishing dedicated paths between all the participants and a central conferencing unit or bridge used to mix multiple speech signals. Due to the fixed bandwidth and short latency that this topology provides, the quality of service (QoS) of such a system is satisfactory. The drawback however is the large inefficiency in bandwidth usage since bandwidth is assigned whether the participant is talking or not. As the number of participants in the conference increases this inefficiency becomes larger and larger.

In VoIP there are no dedicated end to end paths, voice traffic is transmitted only when needed, furthermore speech codecs are used to compress the speech to low data rates thus using the bandwidth very efficiently [35]. A drawback is the introduction of variable and longer delays in delivering the speech from the originating user to the destination.

1.2 Problem Statement

Excessive end to end delay or latency in a VoIP network is one of the major undesirable characteristics of packetized voice communication. This latency directly hinders the natural flow of conversation among end users. The two main components of end to end delay are codec processing and propagation delay. This thesis focuses on the first component.

Codec processing delay becomes a significant factor in centralized VoIP conferencing where multiple tandem coding-decoding pairs take place [11]. The voice packets, originating from the participants, are sent to a central unit or bridge to be combined. At the central unit the packets are decoded, mixed in the linear domain and coded again in order to be sent to their destinations. When adding the codec processing delay at the bridge with the codec processing delay at the endpoints, the delay can become unacceptable. This thesis investigates a novel way of doing the mixing of voice signals at the central conferencing unit that can greatly reduce its codec processing delay by replacing the tandem decoding-encoding process with parametric mixing.

1.3 Thesis Scope

The compression algorithm taken into consideration in this thesis is the ITU-T Recommendation G.722.2 Adaptive Multi-Rate Wideband (AMR-WB) encoder/decoder [25]. This codec is primarily intended for 7 kHz bandwidth speech signals. It has nine different bit rates ranging from 6.6 kbit/s to 23.85 kbit/s that may be changed at any 20 ms frame boundary.

The number of speech signals mixed by this novel approach is limited to two. Investigation into the possibility of mixing more than two signals is left for future work.

1.4 Thesis Objectives and Contributions

The objectives of the thesis are:

- Reduce the algorithmic delay at the bridge of a central voice conferencing architecture through the novel approach of parametric mixing.
- Validate the parametric mixer approach through objective quality measurements of the speech produced.
- Provide an understanding of which parameters, used to represent the speech, impact the speech quality the most.

As a result of the research in this thesis, a list of corrections to the document outlining the G.722.2 standard has been submitted to the International Telecommunication Union (ITU). The list of corrections is included in Appendix A.

1.5 Thesis Outline

The thesis is organized as follows.

Chapter 2 goes over background information of VoIP in general and more specific information used in the thesis. VoIP standards are discussed and factors that affect the QoS in such systems are presented. A brief overview of the human speech production system is given in order to understand the model used by codecs in compressing the speech signal. An overview of the different mixing topologies in VoIP is given with their respective advantages and disadvantages. Generally accepted conversational models, used in the thesis' simulations, and speech quality evaluation techniques are also presented.

Chapter 3 describes the design of the parametric mixer and compares the parameters mixed by the parametric mixer with the same parameters mixed in a linear fashion.

Chapter 4 states the simulation environment, how the speech quality is evaluated and how unit and integration testing is performed.

Chapter 5 presents all the results obtained through the simulations and highlights which parameters affect the speech signal the most.

Chapter 6 summarizes the work performed and highlights possible areas of improvements in the parametric mixer that could be analyzed in future work.

2 Background

This chapter gives the background information for this thesis. The VoIP standards are described to provide an overview of VoIP systems and factors that affect QoS in such systems are discussed. The elementary topologies of VoIP conferencing, the mixing options and the conversation models on which the tests are based are introduced. In order to understand the digital signal processing techniques used in coders, a brief overview of the human speech production system is also given. Speech coders are discussed with special emphasis on the G.722.2 codec [25] used in this thesis. Audio quality evaluation techniques, used in evaluating the performance of the parametric mixer, are addressed. Finally, transcoding methods are presented.

2.1 VoIP Standards/Overview

VoIP is gradually replacing the PSTN systems, due to its more efficient use of bandwidth, its potential for richness of features available to the end user and the guarantee of substantial savings for both the service providers and its customers [35].

A large number of factors are involved in making a high-quality VoIP call. Factors that most influence the QoS are the speech/audio compression technique used, packetization, packet loss rate, end to end delay or latency, delay variation or jitter, and echo [35]. Other important factors are the signaling protocol used and security concerns [35]. In the sections to follow the different factors and their impact on VoIP are discussed.

2.1.1 Call Signaling Protocols

The four most common VoIP call signaling protocols are: H.323 protocol suite, Session Initiation Protocol (SIP), Media Gateway Control Protocol (MGCP) and Media Gateway Controller (MEGACO) or H.248 [1], [3], [6], [7]. The first two protocols are peer to peer control-signaling protocols, while MGCP and H.248 are master-slave control signaling protocols. MGCP is based on the PSTN model of telephony. H.323 and H.248 are designed to accommodate video conferencing as well as basic telephony functions, though being used for packet communications systems, they are still based on a connection oriented approach. The SIP protocol was designed appropriately for IP networks and it can accommodate intelligent

networks engaged in more advanced applications as well as straightforward voice conversations.

2.1.1.1 H.323 Protocol

H.323 [1], being the first standard that helped moved the VoIP industry away from proprietary solutions and towards interoperable products, is the most mature and most deployed solution in the market.

A typical H.323 network consists of a number of zones interconnected by a WAN (wide area network). Each zone has the following components: a single H.323 gatekeeper (GK), a number of H.323 terminal endpoints (TEs), a number of H.323 gateways (GWs), and a number of multipoint control units (MCUs) interconnected via a LAN [2]. The description of the functionality of each component follows:

- A H.323 TE is an endpoint which provides for 2-way real-time communication with another TE, GW or MCU. A terminal can setup a call to another terminal directly or with the help of a GW.
- The GK provides address translation and control access to the network for the other components. The GK is optional in an H.323 system.
- The GW provides real time 2-way communication between TEs in the packet based network and terminals in the PSTN.
- The MCU provides the capability of having three or more terminals participating in a conference.

H.323 is composed by the following four protocols:

- Registration admission and status (RAS) is a transaction oriented protocol between a TE and a GK.
- Q.931, a variation of the Q.931 defined for the PSTN, is the signaling protocol for call setup and teardown between two H.323 TEs.
- H.245 is used for connection control. It is used to negotiate audio/video codecs between two endpoints as well as to open and close logical channels between them.
- Real-Time Transmission Protocol (RTP) is the transport protocol for packetized VoIP.

Figure 2-1 illustrates the relationship of the different protocols in H.323.

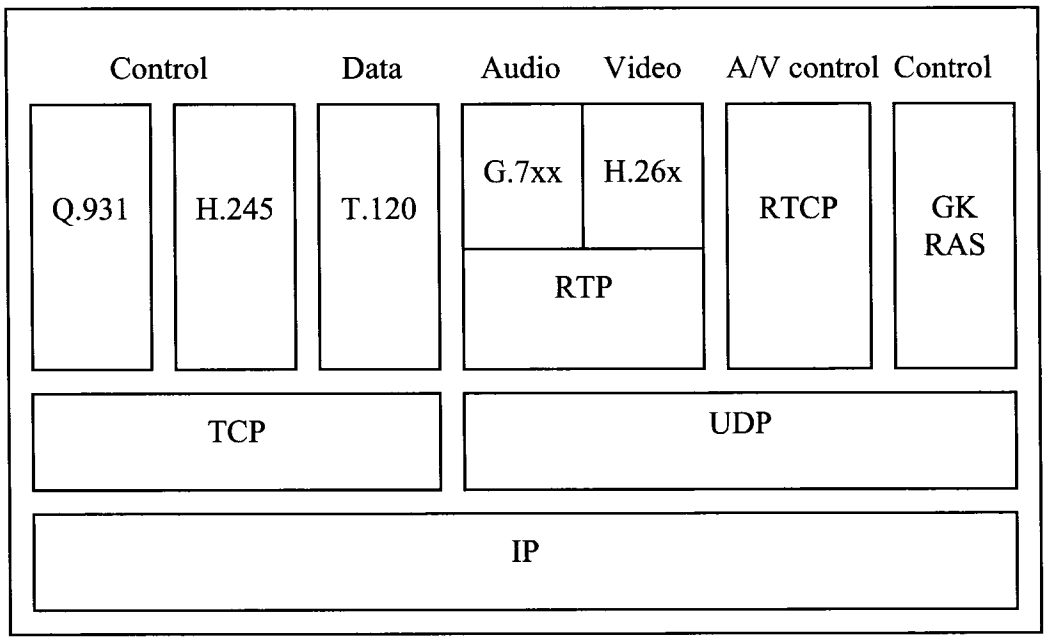

Figure 2-1 H.323 Protocols interaction.

2.1.1.2 SIP Protocol

SIP [3] developed by the Internet Engineering Task Force (IETF) is considered to be a future powerful alternative to H.323. SIP is an application layer control protocol that can establish, modify, and terminate multimedia sessions such as Internet telephony voice calls. SIP call control uses the session description protocol (SDP) [4] to describe the details of the call, such as if it is audio, video, or a different type of multimedia application, which codec to use, and the type and size of packets. Some of the important SIP functionality entities are:

- The user agent which performs the function of both a user agent client that can initiate a SIP request and a user agent server that can contact the user when a SIP request is received and returns a response on behalf of the user.

- The proxy server which is an intermediary entity that makes requests on behalf of other clients. These servers are in charge of tasks such as routing and proxy interpretation.
- The registrars are servers that process registration requests. They can update the location database with the contact information as specified in the request.

Other attractive features of SIP are: it is text-based, this allows easy implementation in object-oriented programming languages [5], making SIP flexible and extensible. SIP involves less signaling, since it is designed to meet only the basic requirements, this means that a call can be established faster than H.323. SIP is independent of the transport layer protocol used and it has the possibility of making parallel searches. For example, several VoIP phone extensions could be rung at once.

2.1.1.3 MGCP

As mentioned earlier MGCP is a master slave protocol [6]. The call processing functions are separated from the gateway functions in an entity called a call agent. The call agent is the master device which handles call signaling and call processing, and sends commands and responses to the media gateway. MGCP, like the SIP protocol, uses the SDP [4] to describe the details of the call.

2.1.1.4 MEGACO/H.248

The development of the MEGACO/H.248 protocol [7] has been a joint activity of ITU-T and IETF. MEGACO contains all of the functionality of MGCP

but it is more flexible in the areas of security and quality of service. While MGCP supports only IPSEC, MEGACO also supports an authentication header. While MGCP only supports UDP for signaling messages, MEGACO supports User Datagram Protocol (UDP), Transmission Control Protocol (TCP), Asynchronous Transfer Mode (ATM) and Stream Control Transmission Protocol (SCTP). MEGACO also has better stream management and resource allocation mechanisms.

2.1.2 Transport Protocols

2.1.2.1 RTP

RTP described by [8] provides end to end delivery services for real-time data such as interactive audio and video. Those services include payload type identification, sequence numbering, time stamping and delivery monitoring. While most applications use RTP on top of UDP, to make use of its multiplexing and checksum services, RTP can be used with any other suitable underlying network or transport protocol. RTP does not provide any QoS guarantees but it relies on lower level services to do so. Note that RTP does not guarantee delivery or prevent out of order delivery; it is up to the receiver to use the sequence number provided by RTP to reorder the packets and to detect missing ones.

2.1.2.2 RTCP

The Real Time Control Protocol (RTCP) also described in [8] is based on the periodic transmission of control packets to all participants in the session, using the same distribution mechanism as the data packets. The main functions provided by RTCP are:

- Provide feedback on the quality of the data distribution. This feedback may be used to control adaptive encoding or to diagnose faults in the distribution list.
- Carry a persistent transport-layer identifier called the canonical name (CNAME). Receivers require the CNAME to keep track of each participant in a set of related RTP sessions.
- Control the rate in order for RTP to scale up to a large number of participants.
- Convey minimal session control information. For example, participant identification to be displayed in the user interface.

2.1.3 Quality of Service

The basic routing philosophy on the Internet, which was originally designed for data communications, is "best-effort". While this might be acceptable by most Internet users, this is not satisfactory for the real-time continuous stream transmission of VoIP. The real-time characteristic for interactive voice communication makes real time audio and video applications intolerable to large end to end delays in the

delivery of packets, packet loss, and jitter. These three issues are the ones that mainly impair QoS in VoIP systems.

2.1.3.1 Delay

Transmission time includes delay due to codec processing as well as propagation delay. Acceptable one way delays, as recommended by ITU-T standard G.114 [9], are:

- 0 to 150 ms: acceptable for most user applications.
- 150 to 400 ms: acceptable for international connections.
- Over 400 ms: unacceptable for general network planning purposes.

In a voice network the two major drawbacks of excessive end to end delay are echo and talker overlap. Echo becomes a problem when the round trip delay is more than 50 ms [36], [37]. The VoIP system addresses the need for echo control by implementing echo cancellation. Talker overlap becomes an issue when the one-way delay is greater than 250 ms. The end to end delay budget is a major constraint and driving requirement for reducing latency in a packet network.

2.1.3.2 Jitter

Jitter, or delay variation, is the variation in inter-packet arrival rate [38]. The receiving gateway or telephone has to compensate for delay variation with a jitter buffer [39]. This buffer delays early packets while passing late packets with less delay, such that the decoded voice streams out of the receiver are at a steady rate.

Any packets that arrive later than the length of the jitter buffer are discarded. To minimize packet loss the length of the jitter buffer is usually set to the maximum delay variation expected over the communication link. The jitter buffer delay must be added to the total end to end delay experienced by a user during a VoIP conversation.

2.1.3.3 Packet Loss

Packet losses can happen for several reasons. Packets can be dropped under peak loads and congestion periods, they can be lost because of data corruption or simply for taking too long to reach the destination thus being discarded by the jitter buffer, as the packet is too late to be usable. Packet loss above a threshold rate introduces audio distortion that causes the voice quality to decrease as the packet loss rate increases. Due to the real time nature of voice transmission the normal TCP based retransmission schemes are not appropriate in this case, so other techniques are used to counteract the packet loss effects. Two of the most common are [40]: a loss concealment algorithm, that consists in replaying the last successful packet received, and forward error correction, that depends on the source sending redundant information that can be used by the receiver to replace the lost frame.

2.2 Speech Production System

The study of the human speech production system is a complex science in itself. In the next few pages only a brief overview is given in order to better

understand the techniques used by speech coders in the analysis of speech waveforms.

The speech waveform is an acoustic sound pressure wave that originates from the voluntary movements of anatomical structures which make up the human speech production system [10]. The main components of this system are the lungs, trachea (wind pipe), larynx (organ of voice production), the pharyngeal cavity (throat), the oral cavity (mouth) and the nasal cavity (nose). From a technical discussion point of view, the throat and the oral cavities are grouped into one common unit referred as the vocal tract while the nasal cavity is usually called the nasal tract [10]. A block diagram of the human speech production system is given in Figure 2-2.

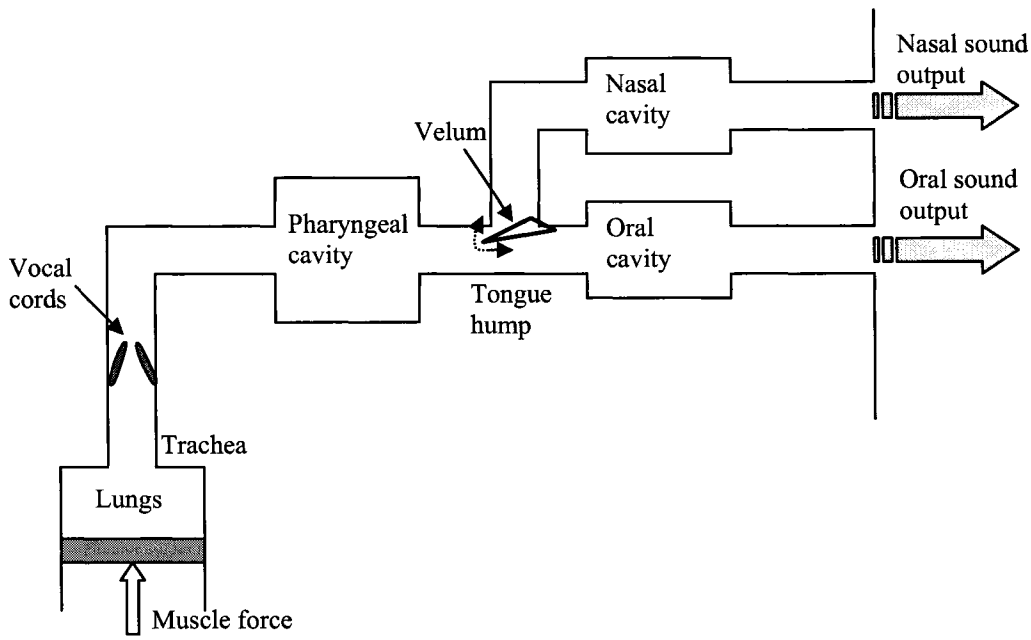

Figure 2-2 Block diagram of human speech production [10].

Other finer anatomical components that are critical to speech production include the vocal cords, the velum, the tongue, teeth and lips. These components,

known as articulators, move to different positions to produce a variety of speech sounds. To analyze the human speech production system, from an engineering point of view, the system can be thought of an acoustic filtering operation. The three main cavities consist of the main acoustic filter. This filter is excited by the organs below it and is loaded at the main output by a radiation impedance due to the lips. The articulators are used to change the properties of the system, its form of excitation and its output loading over time. The average length of the vocal tract for an adult male is about 17 cm, for an adult female about 14 cm, while for a child is about 10 cm [10]. The acoustic coupling between the vocal tract and the nasal tract is controlled by the size of opening at the velum (see Figure 2-2). This coupling can substantially influence the frequency characteristics of the sound radiated from the mouth. The function of the larynx is to provide a periodic excitation to the system for speech sounds that are termed voiced. The periodic vibration of the vocal cords is responsible for this voicing.

2.2.1 Characteristics of Speech Waveform

The spectral characteristics of the speech waveform are time varying (or non-stationary), since the physical human speech production system changes rapidly over time. During speech analysis, this unwanted characteristic can be overcome if the speech is divided into short sound segments, such that, over these short periods of time, the speech possesses similar acoustic properties. The two main categories into which speech sounds are partitioned are:

- Vowels, that contain no major airflow restriction through the vocal tract, for example /a/, /e/, /o/;
- Consonants, which involve significant airflow restriction and are therefore weaker in amplitude and noisier than vowels. Some examples are /z/, /p/ and /f/.

There are two main excitation types, voiced and unvoiced. From the combination of these two kinds of excitation, four more types are classified for modeling purposes: mixed, plosive, whisper and silence [10].

Voiced sounds are produced by forcing air through the glottis, the opening between the vocal cords and the upper part of the larynx. The tension on the vocal cords is adjusted such that they vibrate in an oscillatory fashion. The periodic interruption of the sub-glottal airflow results in quasi-periodic puffs of air. Unvoiced sounds are generated by forming a constriction at some point in the vocal tract and forcing air through it. No matter what the source of the excitation is the vocal tract acts like a filter, amplifying certain frequencies and attenuating others.

As mentioned above, the variation of airflow through the glottis results in a periodic open and close phase for the source excitation. The time between successive vocal cord openings is called the fundamental period T_0 while the rate of vibrations is called the fundamental frequency f_0, which is the inverse of the fundamental period. The term **pitch** is often used interchangeably with fundamental frequency. The pitch range for a man is generally 50 to 250 Hz, while for a woman is 120 to 500 Hz.

Due to the limitations of the human speech production and auditory systems the typical human communication is limited to a bandwidth of 7-8 kHz [10]. Significant

information can be obtained from the frequency domain analysis of acoustic waveforms, i.e., the discrete Fourier transform of their time signals. From this analysis it can be observed that each vocal tract shape is characterized by a set of resonant frequencies. These resonances tend to form the overall spectrum of the speech signal and in speech analysis they are referred to as "Formants". In principle there are several formants in the spectrum of a given sound but in practice only the first three to five formants follow into the Nyquist band after sampling.

A generally accepted discrete-time speech production model, adapted with specific modifications by many codecs' standards, is shown in Figure 2-3 [10].

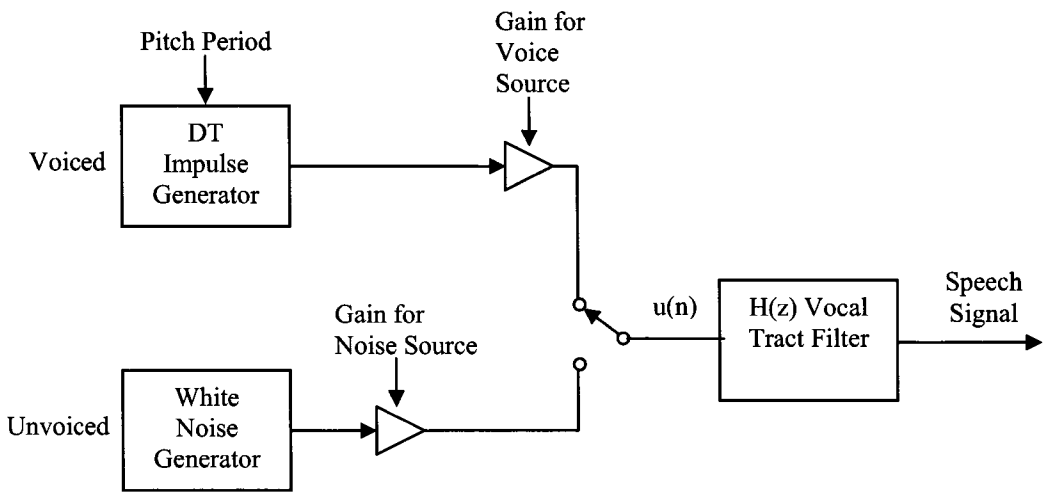

Figure 2-3 Discrete-time speech production model.

In this model the vocal tract is represented by the all pole filter $H(z)$ which is excited by the excitation signal $u(n)$. During unvoiced speech activity the excitation source is a flat spectrum noise source that is modeled by a white noise generator. During periods of voiced speech activity the excitation uses an estimate of the pitch

period to generate an impulse train to model the quasi-periodic characteristics of a voiced signal.

2.3 Speech Coders

Codecs consist of DSP algorithms usually implemented on DSP chips. These codecs enable voice to be transmitted at much lower data rates than the 64 kbps PSTN rate.

The speech quality produced by different codecs is a function of bit rate, complexity, delay and bandwidth. When evaluating speech coders it is important to take all those attributes in consideration. Table 2-1 includes some statistics for the most common speech coders [26], [27], [28], [29].

Table 2-1 Specifications for Different Codecs Standards

Codec	G.711 PCM	G.726 ADPCM	G.728 LD-CELP	G.729 CS-ACELP	G.722.2 ACELP	G.723.1 ACELP
Bit rate (kbps)	64	32	16	8	Multi-rate	5.3
Audio BW (Hz)	300-3400	300-3400	300-3400	300-3400	50-7000	300-3400
Sample rate (kHz)	8	8	8	8	16	8
Frame length (ms)	0.125	0.125	0.625	10	20	30
Algorithmic delay (ms)	0.125	1	2	15	25	37.5
MOS	4.3	4.1	4.0	3.92	4.25 at 12.65 kbps	3.65
Complexity	0.01 MIPS		40 MIPS	18 WMOPS	38 WMOPS	20 MIPS

MOS:

MOS, the mean opinion score, is one of the techniques used to measure the quality of the speech coder. Section 2.6 will discuss this in more detail.

Bit Rate:

Variable rate coders are preferred over fixed bit rate coders because of their adaptability. For example the G.722.2 codec can switch between nine different rates for every 20 ms packet [25] in order to adapt to the channel condition and or to deliver different levels of QoS.

Delay:

The speech codec delay consists of two components. The first component is the algorithmic delay. Speech coders process speech one frame at a time, therefore the whole frame has to be received and buffered before it can be processed. Also some coders use a look ahead technique in order to analyze the data properly, thus adding to the algorithmic delay. The second component is the processing delay which is the time it takes the codec to code and decode the signal.

Complexity:

Complexity of a coder consists of the number of resources needed to carry out the coder's tasks. There are several approaches used to quantify the complexity of a coder. Some of them are: Millions of Instructions per Second (MIPS), Millions of Operations per Second (MOPS), Weighted Millions of Operations per Second (WMOPS), Random Access Memory (RAM) usage, and Read Only Memory (ROM) usage.

Bandwidth:

Most of the speech coders in Table 2-1 are narrowband codecs. That is, the bandwidth of speech is limited to 200Hz to 3400 Hz and sampled at 8 kHz. This limitation alone already reduces the quality of speech since the speech does have components past the 3400 Hz [10]. Wideband codecs address this by increasing the bandwidth to 50 Hz to 7000 Hz. From a perceptual point of view, the extension of the lower side of the bandwidth to 50 Hz enhances the sensation of naturalness and comfort while the extension on the upper side of the bandwidth enhances fricative differentiation and, therefore, provides higher intelligibility [41]. G.722.2 is one of such wideband codecs and it is the codec used in this thesis.

2.4 G.722.2

The speech codec used for the purpose of this thesis is the AMR-WB encoder-decoder described by ITU-T recommendation G.722.2 [25]. The same codec is adopted by 3rd Generation Partnership Project (3GPP) for wireless applications. An overview of the codec is given in the following sections with specific focus on how the coded parameters are extracted from the speech signal.

2.4.1 Principles of the G.722.2 Speech Encoder

The AMR-WB codec consists of nine speech coding modes with bit rates from 6.6 kbps to 23.85 kbps. The bit rate can be changed at any 20 ms frame boundary.

Two frequency bands, 50 Hz to 6400 Hz and 6400 Hz to 7000 Hz, are coded separately in order to decrease complexity and to focus the bit allocation into the subjectively most important frequency range.

The codec is based on the code excited linear prediction (CELP) model [42]. The input signal is pre-emphasized and processed by the CELP algorithm. A 16th order linear prediction (LP) or short term, synthesis filter is used which is given by:

$$H(z) = \frac{1}{\hat{A}(z)} = \frac{1}{1 + \sum_{i=1}^{m} \hat{a}_i z^{-i}}$$

where $m = 16$ is the prediction order and $\hat{a}_i$ are the quantized LP parameters.

The long term or pitch synthesis filter is given by

$$\frac{1}{B(z)} = \frac{1}{1 - g_p z^{-T}}$$

where T is the pitch period and g_p is the pitch gain. The pitch synthesis filter is implemented using the adaptive codebook approach [25]. The CELP speech synthesis model is shown in Fig. 2.4. The excitation signal $u(n)$ is constructed by

adding two excitation vectors, the adaptive codebook vector and the fixed codebook vector both scaled by their respective gains. The excitation signal is then inputted into the LP synthesis filter to obtain the synthesized speech.

The analysis by synthesis approach is used to find the optimum excitation sequence in a codebook, that is by minimizing the error between the original and synthesized speech.

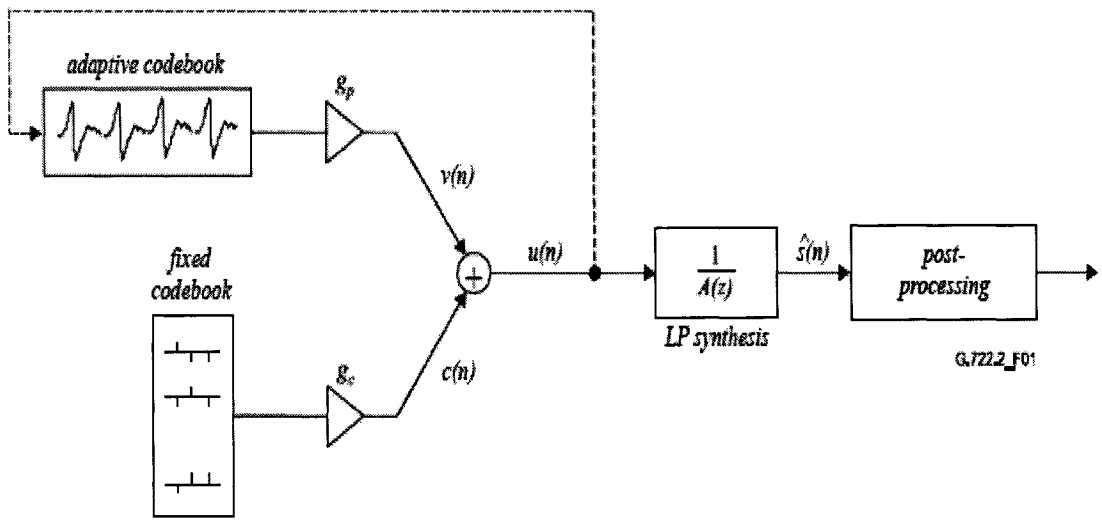

Figure 2-4 Block Diagram of the CELP Synthesis Model [25].

The G.722.2 coder operates on a speech frame of 20 ms which corresponds to 320 samples at a 16 kHz sampling rate. Before performing the CELP analysis the signal is down-sampled to 12.8 kHz. For every frame the speech signal is analyzed and the CELP parameters to be coded are extracted.

The signal flow for the encoder is shown in Fig. 2.5. The speech frame is divided into four sub-frames of 5 ms each, corresponding to 64 samples at 12.8 kHz sampling rate. The LP analysis is performed once per frame. The LP parameters are

converted to immittance spectral pairs (ISP) and vector quantized. The adaptive and fixed codebook parameters are transmitted for every sub-frame.

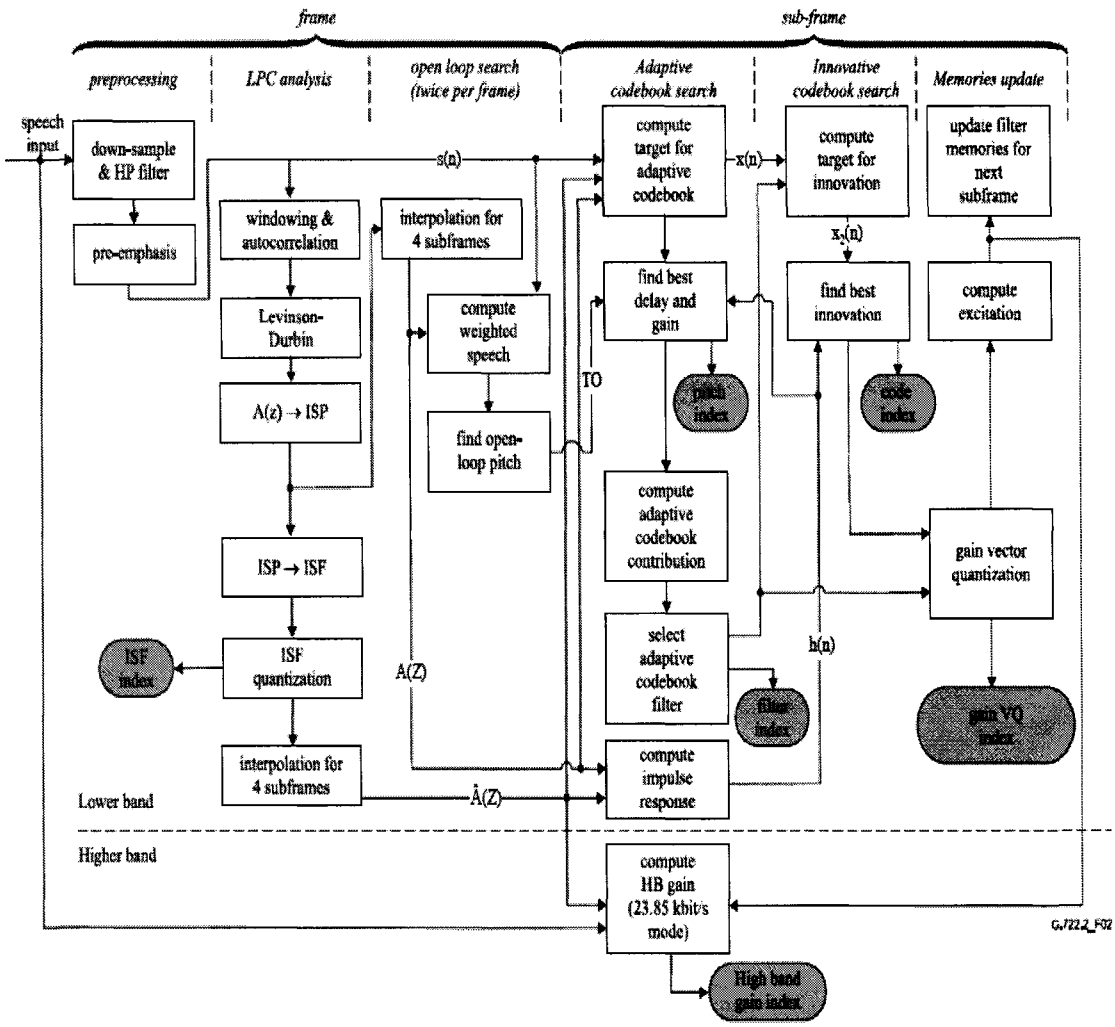

Figure 2-5 Block Diagram of the G.722.2 Encoder [25].

The following operations are repeated for each sub-frame:

- The target signal $x(n)$ is computed by filtering the LP residual through the weighted synthesis filter $W(z)H(z)$;
- The impulse response $h(n)$ of the weighted synthesis filter is computed;

- Closed-loop pitch analysis is performed to find the pitch lag and gain;
- The target signal $x(n)$ is updated by subtracting from it the adaptive codebook contribution;
- The new target signal is used in the fixed codebook search;
- The gains for the two codebooks are quantized.

The bit allocation for the rates used in the thesis are shown in Table 2-2.

Table 2-2 G.722.2 Bit Allocation for Coding Rates 18.25 kbps and 12.65 kbps [25].

Mode	Parameter	1st sub-frame	2nd sub-frame	3rd sub-frame	4th sub-frame	Total per frame
18.25 kbps	VAD-flag					1
	ISP					46
	LTP filtering	1	1	1	1	4
	Pitch delay	9	6	9	6	30
	Fixed codebook	64	64	64	64	256
	Gains	7	7	7	7	28
	Total					365
12.65 kbps	VAD-flag					1
	ISP					46
	LTP filtering	1	1	1	1	4
	Pitch delay	9	6	9	6	30
	Fixed codebook	36	36	36	36	144
	Gains	7	7	7	7	28
	Total					253

2.4.2 Principles of the G.722.2 Speech Decoder

The signal flow at the decoder is shown in Fig. 2.6. The transmitted indices are extracted from the transmitted bit stream and decoded to obtain the coder parameters for each frame. The LP filter coefficients are obtained and then for each sub-frame the excitation is constructed, the 12.8 kHz speech is synthesized, by filtering the excitation through the LP filter, de-emphasized and up-sampled to 16 kHz. The higher frequency band (6400 Hz to 7000 Hz) is reconstructed using the parameters of the lower band and a random excitation.

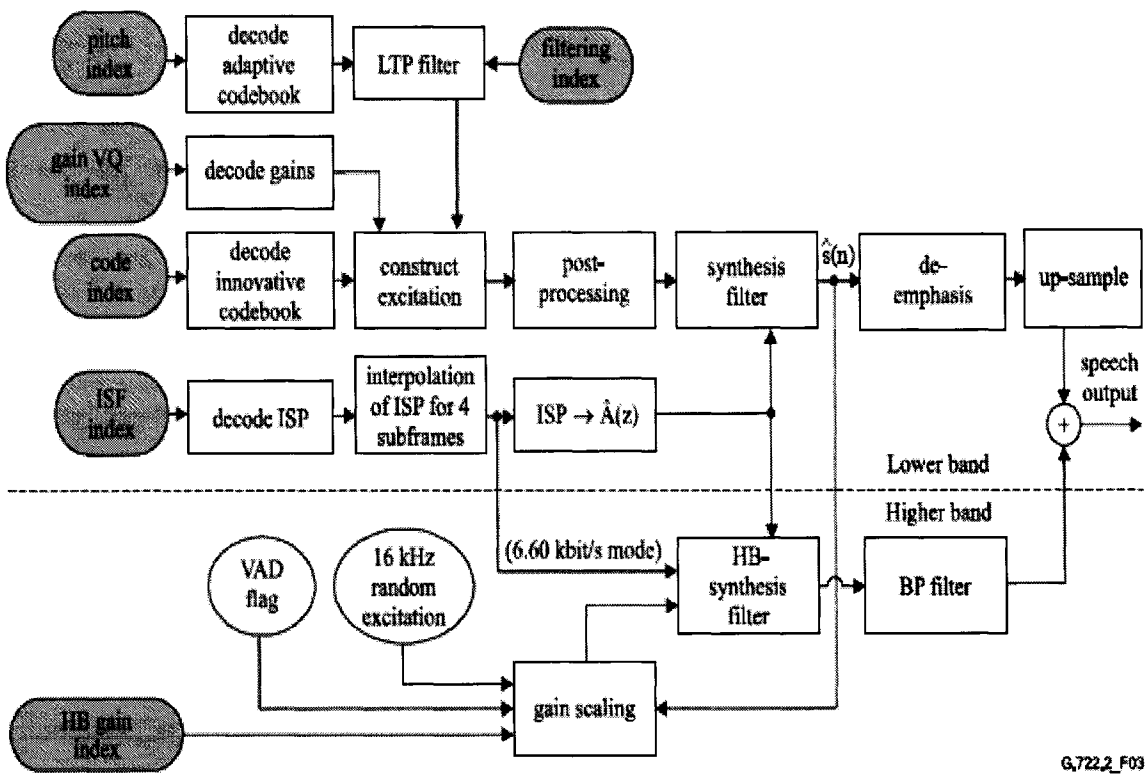

Figure 2-6 Block Diagram of the G.722.2 Decoder [25].

2.4.3 G.722.2 Parameters Extraction Techniques

This section will present in more details how the parameters needed to represent the speech signal are extracted by the G.722.2 coder.

These parameters are:

- LPCs, used to obtain the formant frequencies and reconstruct the spectral envelope of the speech frame, extracted by linear prediction analysis.
- Pitch lags and pitch gains, used to represent the fundamental frequency of the speech and construct the excitation to the LPC synthesis filter, extracted by adaptive codebook analysis.
- Fixed codebook and fixed codebook gains, used to further refine the excitation to the synthesis filter, extracted by algebraic codebook analysis and gain quantization.

2.4.3.1 Linear Prediction Analysis

Short term prediction, or LP, is performed once per speech frame using the autocorrelation approach with 30 ms windows which include the 20 ms frame being analyzed, a 5 ms look-ahead and the last 5 ms of the previous frame [25]. Once the autocorrelations are calculated they are converted to LP coefficients using the Levinson-Durbin algorithm. The LP coefficients are then converted to ISPs before quantization.

The ISP representation of the LP filter coefficients are quantized in the frequency domain. The immittance spectral frequencies (ISF) lie in the interval 0 Hz

to 6400 Hz for a sampling frequency of 12.8 kHz. The ISF are quantized by using a split-multistage vector quantization.

2.4.3.2 Adaptive Codebook Analysis

The adaptive codebook analysis consists of finding the pitch lag and pitch gain of the speech frame being analyzed. This analysis is performed in two parts, open loop pitch analysis and closed loop pitch analysis.

Open-Loop Pitch Analysis

Open loop pitch analysis is performed twice per frame for all modes except 6.6 kbps, for which pitch analysis is done once per frame. The goal of open loop pitch analysis is to find an estimate of the pitch lag which is afterwards refined by the closed loop pitch analysis. The open loop pitch search is done in order to simplify the overall pitch search and confine the closed loop pitch analysis to a small number of lags around the open loop pitch estimate.

To find the open loop pitch estimate, first the speech signal is decimated by two, then the correlation of the decimated weighted speech is determined for each pitch lag in the interval 17 samples to 115 samples which corresponds to ~ 57 Hz to 376Hz. The correlations are determined using:

$$C(d) = \sum_{n=0}^{63} s_{wd}(n) s_{wd}(n-d) w(d), \qquad d = 17,\ldots,115 \quad [25]$$

where $w(d)$ is a weighing function used to emphasize lower pitch lags in order to reduce the possibility of selecting a multiple of the correct lag. The delay that maximizes $C(d)$ is the estimated pitch lag.

Closed-Loop Pitch Analysis

The closed loop pitch analysis is performed around the open-loop pitch estimates on a sub-frame basis. For all modes except 6.6 kbps, in the first and third sub-frame the range searched is T_{op} (open loop pitch) ±7 bounded by 34 and 231 samples. For the second and fourth frame the closed loop search is done around the integer pitch of the previous sub-frame.

Depending on the pitch interval the pitch resolution can be ¼ , ½ or an integer.

The search for the optimal pitch is performed by minimizing the mean-square error between the original speech and the synthesized speech. This is achieved by maximizing the term:

$$T_k = \frac{\sum_{n=0}^{63} x(n) y_k(n)}{\sqrt{\sum_{n=0}^{63} y_k(n) y_k(n)}} \quad [25]$$

where $x(n)$ is the target vector and $y_k(n)$ is the past excitation at delay k filtered trough the synthesis filter, i.e. the synthesized speech. Once the optimum integer pitch delay

is determined the appropriate fractions around that integer value are tested for a more accurate pitch estimate.

For the rates taken in consideration in the thesis the pitch delay for the first and third sub-frame is coded with 9 bits while the relative delay of the other two sub-frames is coded with 6 bits.

There are also two signal paths to calculate the adaptive codebook excitation. The signal path having the lowest calculated pitch error is selected and 1 bit is used to code which path is taken.

Once the optimum pitch delay is found the pitch gain is calculated as follows:

$$g_p = \frac{\sum_{n=0}^{63} x(n)y(n)}{\sum_{0}^{63} y(n)y(n)}, \qquad \text{bounded by } \; 0 \le g_p \le 1.2, \quad [25]$$

where $y(n)$ is the filtered adaptive codebook vector.

2.4.3.3 Algebraic Codebook

The fixed codebook or algebraic codebook vector analysis is performed on a sub-frame basis as well. The codebook structure is based on the interleaved single-pulse permutation design. There are 64 possible pulse positions in the code-vector corresponding to the 64 sample size of each sub-frame. The 64 positions are divided into 4 tracks of interleaved positions with 16 possible pulse positions per track. Depending on the rate used, the codebook is constructed by placing 1 to 6 signed

pulses in each track. The codebook index transmitted represents the positions of the pulses in each track, therefore no codebook storage is required.

Again in this analysis, the algebraic codebook is searched by minimizing the mean square-error (MSE) between the weighted input speech and the synthesized speech. The target signal is updated by subtracting the adaptive codebook contribution:

$$x_2(n) = x(n) - g_p y(n), \qquad n = 0,...,63$$

where the parameters are as described in the previous sections. The minimization of the MSE is achieved by maximizing:

$$Q_k = \frac{\left(\mathbf{x}_2^t \mathbf{H} \mathbf{c}_k\right)^2}{c_k^t \mathbf{H}^t \mathbf{H} c_k} \quad [25]$$

where c_k is the code vector at index k and H is the lower triangular Toeplitz convolution matrix.

2.4.3.4 Gains Quantization

One other coded parameter needed to reconstruct the speech at the decoder side is the gain index. The pitch gain and the fixed codebook gain are jointly vector quantized using a 6 bit codebook for rates 8.85 and 6.6 Kbps and a 7 bit codebook for

all the other rates. The gain codebook search is performed by minimizing the MSE between the original and reconstructed speech. The error is given by:

$$E = x^t x + g_p^2 y^t y + g_c^2 z^t z - 2g_p x^t y - 2g_c x^t z + 2g_p g_c y^t z$$

where x is the target vector, g_p is the pitch gain, y is the filtered adaptive codebook vector, g_c is the fixed codebook gain and z is the filtered fixed codebook vector.

2.4.4 Voice Activity Detector

The G.722.2 codec also utilizes an integrated Voice Activity Detector (VAD). The function of the VAD is to indicate whether the 20 ms frame contains signals that should be transmitted or whether there is no speech and the frame could be optionally discarded. The output of the VAD flag is a Boolean indicating the presence of such a signal thus it is encoded using 1 bit.

The VAD algorithm uses parameters of the speech encoders, such as pitch gains, and signal levels to make its decision.

2.5 Mixing in VoIP

The topology of a VoIP conferencing system is key to factors such as speech quality, scalability, and computational complexity of bridge and endpoints. The topology influences the number of transcodings and end to end delay which directly influence the synthesized speech quality. Perceived quality is also affected by the level of participation allowed to the conferees or how natural the conversation can be. For example, some topologies only allow one speaker to be heard at one time which reduces interactivity and forces the participants to "compete" for talking privileges. In general, to maintain a good level of interaction among participants having two to three talkers being allowed to speak simultaneously is acceptable [11].

Traditional conferences have been provided by a centralized conference bridge to which users dial-in. The endpoints establish a one to one connection with the bridge and the bridge provides voice paths among the endpoints by summing the input signal and returning that sum to the conferees. To prevent echo the summed signal sent to a user includes all of the participant's voice except for its own. To further improve on factors, such as background noise, the bridge might sum only M out of N active talkers. This implies that $M + 1$ sums are formed by the bridge. One sum for each of the M talkers and one for the listeners.

The main debate over VoIP conferencing has been whether centralized or decentralized architectures provide the better speech quality, flexibility, and ultimately the most economical way of doing VoIP conferencing.

The following sections focus on the different topologies of VoIP conferencing, with their advantages and disadvantages.

2.5.1 Conventional VoIP Conferencing

A generic VoIP conference bridge receives compressed speech encapsulated into RTP packets. These packets are extracted and added to a jitter buffer. At the scheduled play out time the speech data are decoded and added to the mix buffer from which M+1 sums are formed [12]. The M + 1 sums are then encoded again and encapsulated into RTP packets to be sent to the endpoints. The speaker selection is used to limit the number of output sums and therefore the number of encodings needed thus reducing the computational complexity of the bridge. But such an architecture can result is a significant reduction of speech quality due to the tandem arrangements of low-bit-rate speech codecs [12]. Furthermore the QoS is reduced by the additional delay imposed by jitter buffers and codec processing which can result in doubling the end to end delay. The speech processing operations, other than causing delay, are very computationally demanding limiting the scalability of the bridge.

2.5.2 Select and Forward VoIP Conferencing

A slight improvement over traditional VoIP conferencing systems is the select and forward architecture. In this case the bridge, instead of the usual decode-mix-code process, forwards the coded speech to the endpoints after selecting M out of N streams. This technique is characterized by:

- The number of simultaneous speakers allowed.

- The use of partial or full decoding to decide which data streams to forward to the endpoints.
- The amount of tandeming and or transcoding.
- The bridge's codec dependencies.

The first tandem-free bridge was built by Forgie [13]. The compressed signal of the primary speaker was selected and forwarded to the N - 1 conferees while the signal of the primary interrupter was sent to the primary speaker. Fig. 2.7 shows such a topology with $N = 4$ and $M = 1$. The speaker selection was accomplished by a first come first served (FCFS) scenario. The main disadvantage of this approach is that the conference is turned into a series of monologues, thus taking away from the natural flow of conversation.

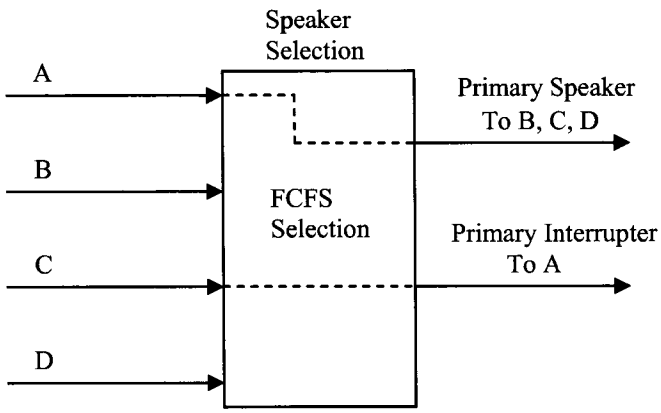

Figure 2-7 Select and Forward Tandem Free Bridge with $N = 4$ and $M = 1$.

A different approach from [13] is to use select and forward during single talk, while going back to the normal mixing process during multi-talk [43]. This way

speech quality is improved "most of the time" as multi-talk usually accounts for a small percentage of the total conference time. Computational complexity at the bridge is reduced as well. FCFS is used again to select M speakers and a partial decoding process is used to monitor parameters, such as gain, in order to make a VAD decision. By having limited the number of speakers allowed to talk at the same time to M, then only M full decoders and $M + 1$ encoders would have to be allocated to the bridge. Figure 2.8 shows such a configuration. In this case tandem encoding is only performed while two speakers are talking at the same time.

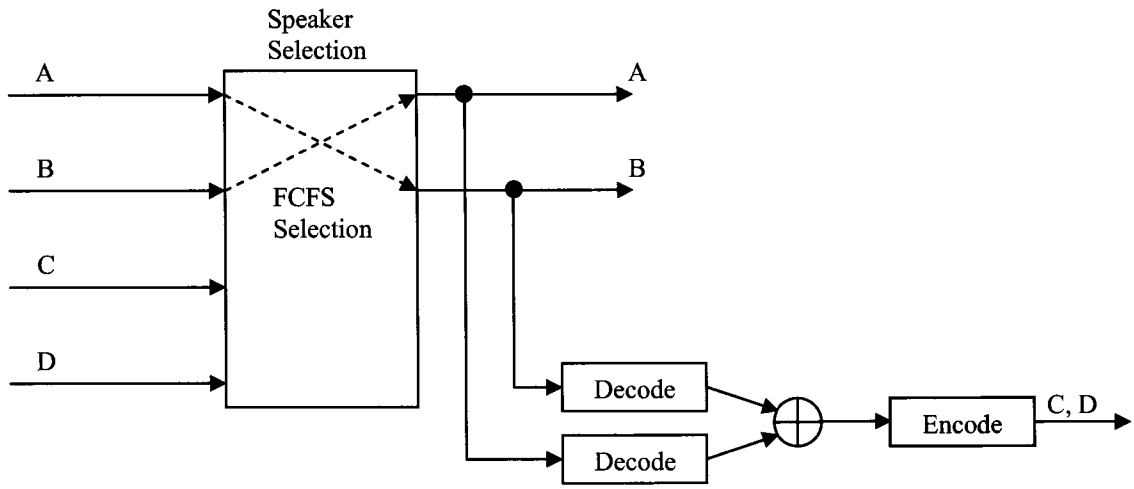

Figure 2-8 Select and Forward Bridge with $N = 4$ and $M = 2$.

One of the disadvantages of this technique is that audible pops are introduced in the synthesized speech when transitioning from single to multi-talk states since the two states have different algorithmic delay [12]. This becomes more and more undesirable as interactivity of the conference increases.

Both of the techniques above transmit only one output stream to the endpoints. If the end units are able to receive multiple streams then the technique described in [14] could be used. This design selects and forwards the signal of the primary speaker during single-talk and the signals of both primary and secondary speakers during multi-talk. In order to keep the downstream channel bandwidth required the same, the bridge transcodes the two streams to half the bandwidth before forwarding them to the N - 2 listeners as shown in Fig. 2.9.

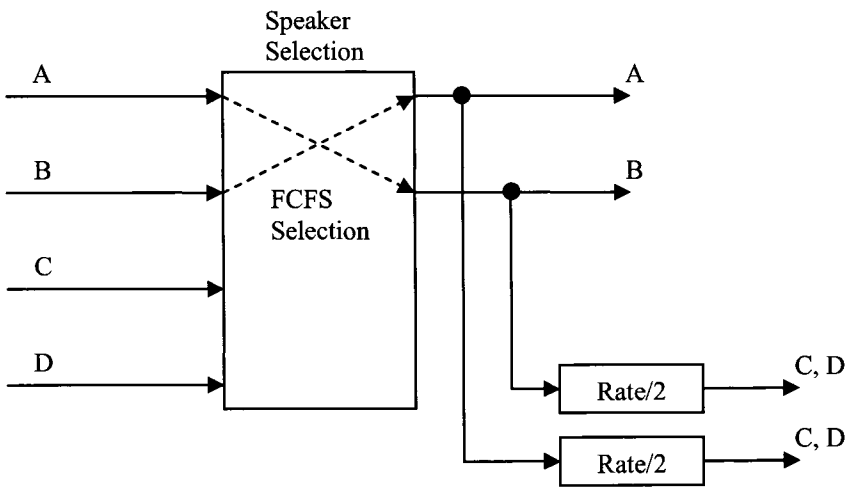

Figure 2-9 Tandem Free Dual-Rate Bridge, $N = 4$, $M = 2$.

Speaker selection is computed by FCFS in this case as well but the VAD decision is computed at the source and included in the upstream packets as a header. While tandeming and complexity is taken out of the bridge, the result is that the terminals, i.e. endpoint phones, become more complex.

2.5.3 Decentralized VoIP Conferencing

In decentralized VoIP, conferencing media streams are exchanged between the endpoints without the use of centralized bridges. Obviously this improves the speech quality but the endpoints have to have the ability to receive and mix multiple streams. The two most common applications of this type are full mesh conferencing and multicast conferencing [15].

In full mesh conferencing each endpoint establishes a full duplex media connection with every other endpoint resulting in a mesh of connections as showed in Fig. 2.10.

Each endpoint transmits N - 1 bit streams and receives N - 1 bit streams from the other conferees. This type of scenario is only suitable if there are a limited number of participants in the conference as its scalability is limited. In the worst case N^2 - N streams will flow through the network and at the endpoints there must be bandwidth available for N - 1 full duplex connections. Bandwidth can be improved slightly if silent suppression techniques are used. In this case the worst bandwidth requirement only occurs if all N participants are talking at the same time.

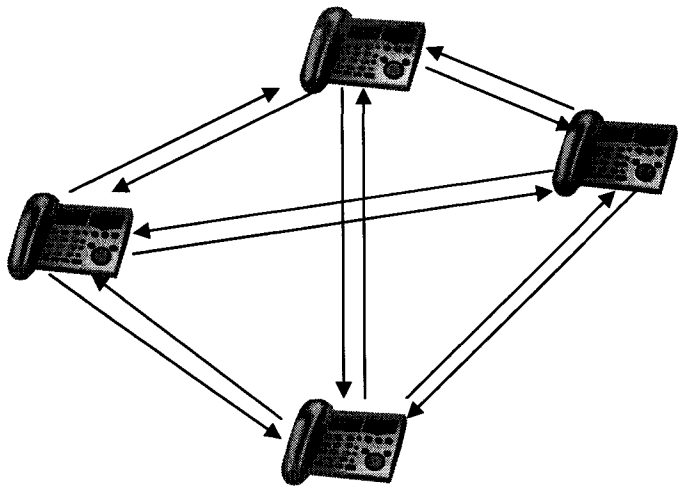

Figure 2-10 Full Mesh Conferencing Topology, $N = 4$.

In a multicast conferencing scenario each endpoint transmits a single copy of its stream to the conference multicast address, thus making a more efficient use of bandwidth, but it still has to be capable of receiving N - 1 streams.

2.5.4 Tandem Free VoIP Conferencing

Tandem Free Conferencing (TFC) architecture [15] is a cross-over between the traditional centralized approach and the decentralized one. This design uses a tandem free bridge which is a multi-talker select and forward conference bridge. The bridge selects M current speakers and forwards their compressed signals to the other N - M endpoints. Unlike the architectures described in the previous paragraphs no tandeming or transcoding occurs during multi-talk. This provides consistent speech quality over the course of the conference. The sources compute and encapsulate the parameters used for speaker selection into TFC data frames. The bridge will monitor

these data frames in order to decide which data streams to forward, therefore being free of any codec dependencies. Speaker selection can be obtained through different techniques. Some of the most common are FCFS as described previously or Multi-Speaker/Interrupter [16]. The latter algorithm assigns talking privileges according to the order of activity, the power signal envelope, and a "barge-in" threshold.

While this design has great advantages, no tandeming, independency from speech codecs, and low computational complexity at the bridge, it requires protocol extensions to carry the TFC data. Also the endpoints must still support multiple streams termination and mixing. It is however bandwidth scalable as it limits the downstream bit rate to that of M streams with M usually set to two or three.

2.6 Conversation Models

The conversation model is a key component in modeling and evaluating the different VoIP conferencing architectures. In order for this analysis to happen the conference has to be classified into different states of single talk, double talk, silence and interruption. VAD technology is used, as mentioned earlier, to accomplish this classification.

ITU-T P.59 [17] provides a four-state model that can be used to generate artificial two-way conversations with single talk, double talk and mutual silence accounting for 71 %, 11 % and 18 %, respectively. While this provides initial design parameters for speech activity detection it may not be applicable to the conferencing

environment since it is based on two party conversational dynamics. Conferences with different objectives can have different models. Some conferences may be used for announcements where interactions are very limited, while others can be used for problem solving where all the conferees have an active role and therefore the interaction level is very high. The research in [11] tries to determine the optimal number for M, the number of speakers allowed to talk at the same time. In general, the results indicate that a multi-speaker conference system is preferred, which is not surprising. In general, setting $M = 2$ does improve the user acceptance of the system. $M = 3$ further improves the performance and anything higher does not differ significantly from the performance obtained with $M = 3$ [11].

In general, the multi-talk intervals accounts for 6 % to 11 % of the total conferencing time, and can be characterized in the following five categories [13]:

- Reinforcement: Consists of feedback to the primary speaker. They are usually short exclamations such as "yes", "right", "no", or non speech sounds such as a chuckle.
- Overlap: A new speaker judges that the person currently speaking is finishing and starts talking before the other one is done.
- Interruption: The new speaker cuts off the primary speaker.
- Collision: More than one person starts talking at the same time. In general, when this happens one of them will back off and let the other one finish.
- Noise: Such as the typing on a keyboard, the shutting of a door, etc., which ideally is not supposed to be heard by the other conferees.

2.7 Audio Quality Evaluation Techniques

The voice quality of the PSTN has become the standard to compare and quantify, in an objective manner, the VoIP network voice quality. Two different measurement approaches can be used to determine the voice quality: subjective and objective.

2.7.1 Subjective Measurements

The most widely used subjective quality assessment methodology is opinion rating defined in ITU-T Recommendation P.800 [18]. The performance of the system under test is rated directly (absolute category rating, ACR) or relative to the subjective quality of a reference signal (degradation category rating, DCR). The following opinion scale is the most frequently used in ITU-T: excellent (5), good (4), fair (3), poor (2), and bad (1). A score of four is considered toll quality.

In order to perform an evaluation of speech quality, a statistically mixed panel of men and women listen to the collection of voice samples that have been processed through a coder and decoder, and votes on their quality using an integer opinion score. The arithmetic mean of all the opinion scores collected is the mean opinion score (MOS).

Determining the speech quality through a subjective measurement has always been an expensive and time consuming process, thus it is not the preferred method in many situations.

2.7.2 Objective Measurements

The most common objective measurement techniques are Perceptual Speech Quality Measurement (PSQM) [32], the Perceptual Analysis Measurement System (PAMS) [19] and the Perceptual Evaluation of Speech Quality (PESQ) [33]. In order to evaluate the speech quality of the parametric mixer, in this thesis the PAMS objective measurement system is used since a Digital Speech Level Analyzer, that performs such an evaluation, is available in the Digital Signal Processing lab of Carleton University.

The PAMS objective measurement is signal-based. It uses the reference signal and degraded signal as inputs and identifies the audible distortions based on the perceptual domain representation of two signals incorporating human auditory models.

PAMS was developed by British Telecom and is widely accepted in Europe and North America. The current version was released in December 1999 [19].

The PAMS score is mapped to a scale similar to MOS, from 1 to 5. A listening quality score (Ylq) and a listening effort score (Yle) are two measurements generated by PAMS.

2.8 Transcoding

In this section some transcoding techniques are discussed. Transcoding avoids tandeming by translating from one codec to another at the parametric level. This is a similar idea to the parametric mixer design discussed in this thesis.

In [20] a transcoding technique is proposed between the G.729 and 3GPP Adaptive Multi Rate (AMR) codecs. The idea is to use a bit-stream mapping approach to convert the bit-stream from one coder to another in the parametric domain. Both coders are based on the CELP architecture and therefore extract similar sets of parameters used to represent the speech. Table 2-3 shows the specifications for the codecs. The AMR is a multi-rate codec and for the purpose of this transcoding only the 7.95 kbps rate is taken into consideration for comparison with G.729

Table 2-3 G.729 vs AMR Parameters.

	G.729	**AMR (7.95 kbps)**
Algorithm	CELP	CELP
Bit-rates	8 kbps	7.95 kbps
Frame Length	10 ms	20 ms
Sub-frame length	5 ms	5 ms
Transmission parameters		
LSP	18 bits / 10ms	27 bits / 20ms
Pitch delay	13 bits / 10 ms	14 bits / 10 ms
Fixed codebook	17 bits / 5 ms	17 bits / 5 ms
Gains	7 bits / 5 ms	9 bits / 5 ms

Four sets of parameters, LSP, pitch delay, fixed codebook index, and adaptive and fixed codebook gains, are decomposed and converted into the corresponding parameters of the other standard.

A subjective listening test conducted to evaluate this transcoding technique shows that the quality of the proposed translation is almost equivalent to the original speech. A clear advantage of this transcoding technique is a reduction in processing delay. In a conventional tandem configuration between G.729 and AMR the frame length must be aligned to 20 ms. In addition, both standards require a 5 ms look-ahead. Thus the processing delay for a one-way transmission can be reduced by 25 ms.

[21], [22], [23] and [24] propose other transcoding techniques among different CELP codecs. The results obtained by these transcoding techniques versus the tandem transcoding consistently show a significant saving in computational complexity, a significant reduction in processing delay and an equivalent or better speech quality as one stage of decoding and coding has been removed.

2.9 Strong Pitch Domination Mixer

The research in [34] introduces several innovative methods used to simplify the mixing of speech signals at the bridge of a centralized voice conferencing architecture. One of particular interest to this thesis is the simplification introduced in

the "Strong Pitch Domination Mixer". In this scenario a form of parametric mixing is used to reduce the algorithmic delay in tandem mixing for CELP based codecs.

Fig. 2.7 shows a block diagram of such design. In this case two active speakers are present. The two active speech signals coming into the bridge are decoded, and the parameters for pitch delay and pitch gain are extracted for both signals. The Pitch Estimator block compares the pitch delay parameters and chooses the one with the strongest pitch energy and less pitch delay variation as the open loop pitch delay of the mixed signal. This open loop pitch delay is then sent to the encoder as the candidate pitch delay for the closed loop pitch analysis. This eliminates the need for the open loop pitch search in the encoder of the two linearly added speech signals thus reducing the algorithmic delay of the overall mixing architecture of the bridge.

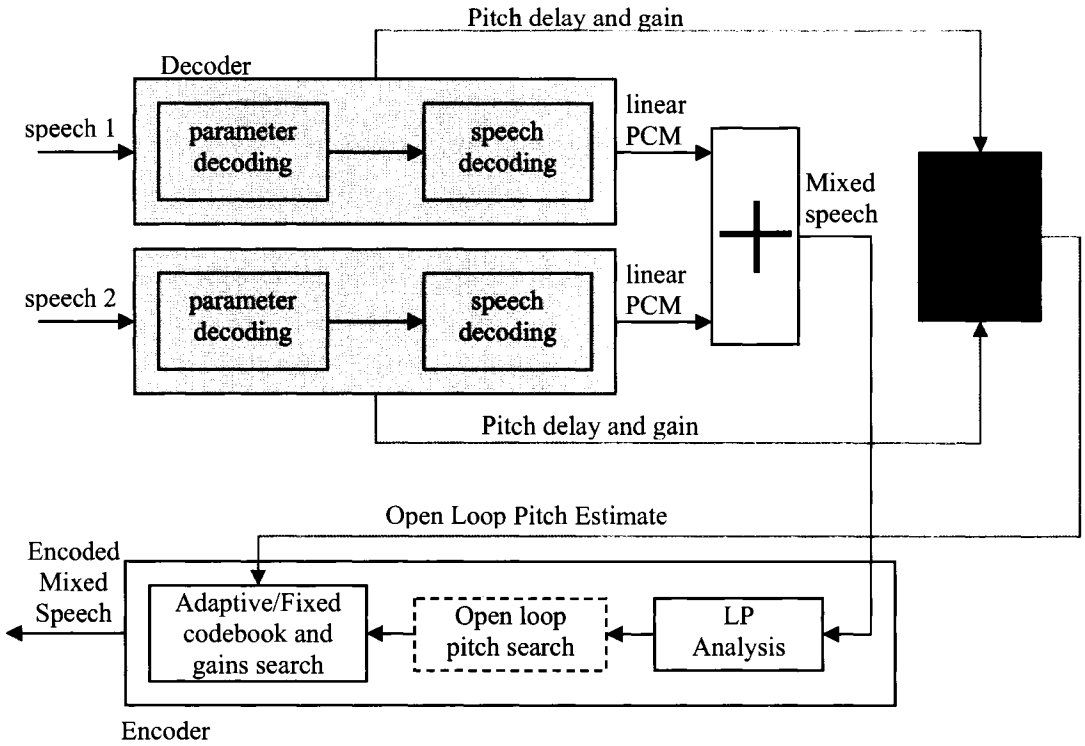

Figure 2-11 Block Diagram of Strong Pitch Domination Mixer [34].

The technique used by the Pitch Estimator in selecting the best pitch delay is also used, with appropriate modifications, in the parametric mixer design discussed in Chapter 3.

3 Parametric Mixer Design

This chapter describes the design of the parametric mixer for a centralized VoIP conferencing topology implemented in this thesis. A high level block diagram of the mixer is shown in Fig. 3.1.

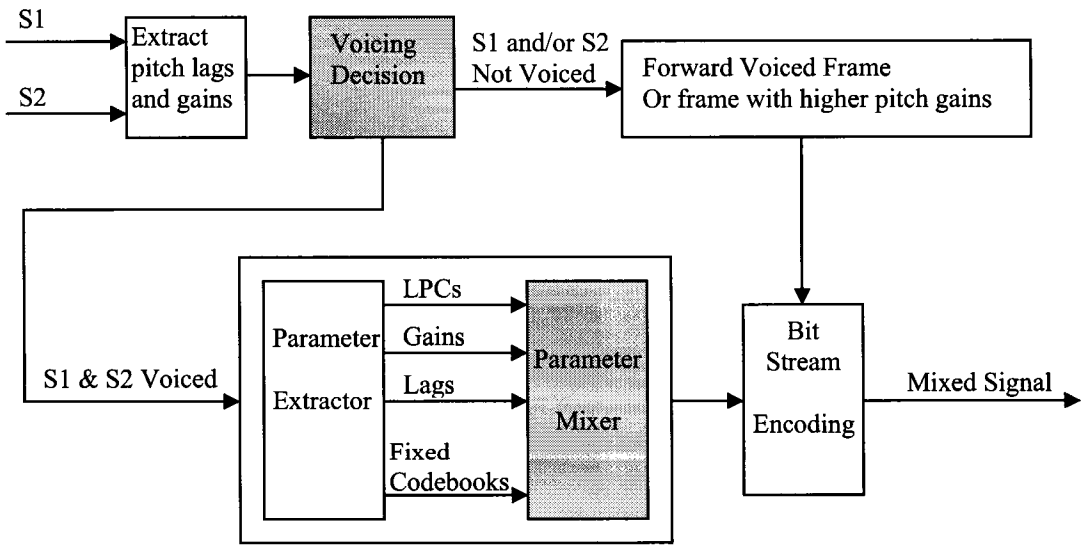

Figure 3-1 High Level Block Diagram of Parametric Mixer.

For the two speech signals, *S1* and *S2*, being mixed at the bridge, the pitch gains are extracted at every frame. The voicing decision block uses the pitch gains to decide if the frames being analyzed are voiced or unvoiced. From the outcome of this decision one of the following actions is taken:

- If both frames are deemed voiced then all parameters are extracted, mixed, and the mixed parameters are encoded in the bit stream.
- If only one frame is voiced then only the parameters from the voiced frame are encoded into the bit stream.
- If both frames are unvoiced then only the parameters from the frame with the greater mean pitch gain are encoded in the bit stream.

The major implementation blocks of the parametric mixer are the voicing decision block and the parameter mixer block highlighted in Fig. 3.1. These two blocks consist of the following four algorithms:

1. Algorithm used to make decision on whether frames should be mixed or not;
2. Algorithm used to mix linear prediction coefficients;
3. Algorithm used to mix pitch lags and gains;
4. Algorithm used to mix the fixed codebooks;

The following sections describe the design of these components in more details.

3.1 Voicing Decision Functional Block

This functional block of the parametric mixer uses the named voicing algorithm to make the decision on whether two frames have to be mixed or not by analyzing the pitch gains and the pitch lags.

As mentioned in Chapter 2, for every 20 ms frame the G.722.2 encoder calculates four pitch gains, one for each 5 ms sub-frame, by using the following equation [25]

$$g_p = \frac{\sum_{n=0}^{63} x(n)y(n)}{\sum_{n=0}^{63} y(n)y(n)}, \qquad \text{bounded by } 0 \le g_p \le 1.2,$$

In the ratio above *x(n)* is the target speech signal, used for the adaptive codebook search, and *y(n)* is the synthesized speech which consists of the estimated excitation signal filtered through the synthesis filter.

If the speech frame analyzed is a voiced speech frame, due to its periodicity, the estimate *y(n)* will be very close to the target signal *x(n)*. In this case g_p can be considered as a normalized cross-correlation coefficient which will approach one for a fully voiced speech frame. This characteristic, and the fact that in the presence of voiced signals the gains across the frame will stay approximately constant, is used in making the voicing decision.

Another factor, also used in making the voicing decision, is derived from the characteristics of the pitch lags. The G.722.2 encoder, as for the pitch gains, calculates four pitch lags for every 20 ms speech frame. One per 5 ms sub-frame.

The pitch lags as well as the pitch gains will also stay approximately constant across a voiced 20 ms frame as they represent the fundamental frequency in the speech.

Fig. 3.2 shows a sample of pitch lags for voiced speech over ten 20 ms frames for a total of forty lags. Each grid line on the x-axis is a start of a new frame. Note that each frame is split into four subframes, so Fig. 3.2 shows a pitch lag for each subframe. As seen from this figure, the variation in the lags is very minimal considering that the search range in the G.722.2 encoder for the pitch lags extends from 34 samples to 231 samples. For every speech frame, except for the last one in Fig. 3.2, the absolute difference between first and second lag and third and fourth is less than 3 samples with an average lag of 133 samples. The pairing between the lags of the first two sub-frames and the lags of the last two sub-frames for each frame is used due to the dependency between each other. For an explanation of this dependency refer to Section 3.3.

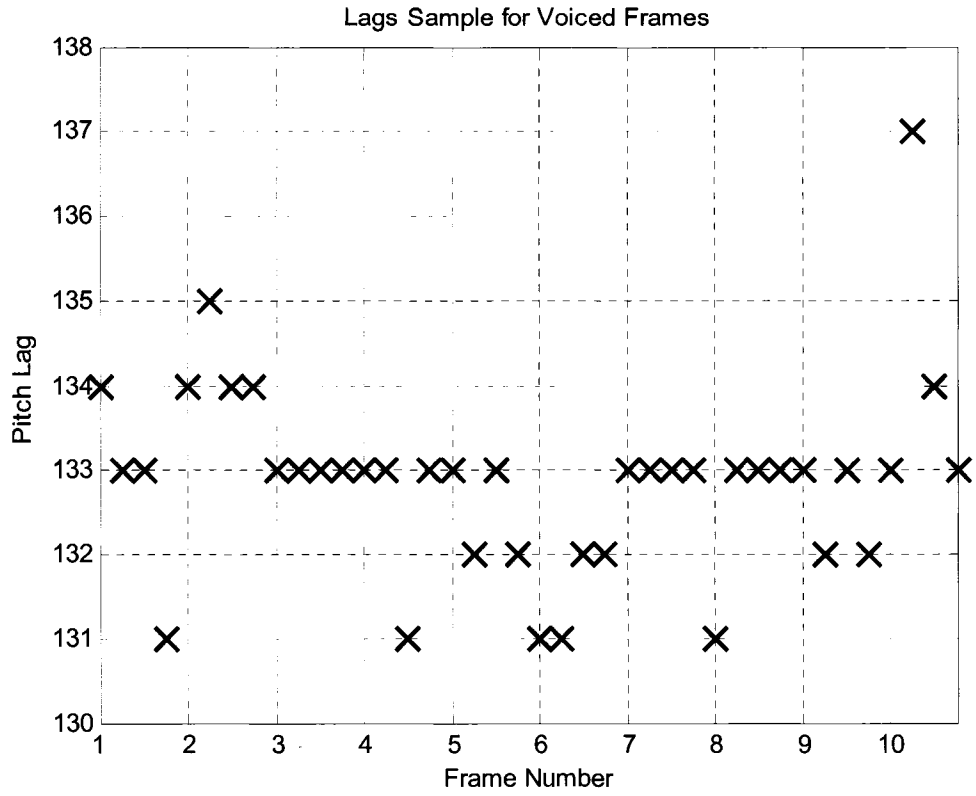

Figure 3-2 Pitch Lags Sample for Voiced Frames.

The voicing algorithm used by the parametric mixer analyzes the frames by dividing them into two sections. The first section corresponding to the first two sub-frames and the second section corresponding to the last two sub-frames.

The thresholds used in the voicing algorithm chosen by heuristic measurements and analysis as the one above, are shown in Table 3-1.

Table 3-1 Thresholds Used in Voicing Algorithm.

	Threshold Value	Notes
gain_mean_th	0.7	Mean voicing threshold
gain_std_dev	0.45	Maximum gain standard deviation between gain pairs 1, 2, and 3, 4
gain_th	0.66	Minimum gain value
lag_std_dev	3	Maximum standard deviation between lag pairs 1, 2, and 3, 4

In the parametric mixer design, a section is considered un-voiced unless one of the following two conditions is true:

1. The mean of the two gains is greater than *gain_mean_th* **and** the standard deviation between the two is less than *gain_std_dev* **and** the standard deviation between the two corresponding lags is less or equal to *lag_std_dev.*
2. Both gains are greater than *gain_th* **and** the standard deviation between the two corresponding lags is less or equal to *lag_std_dev.*

If the voicing algorithm determines that at least one section of the frame is voiced, then that frame will be forwarded on to the parameter extractor and mixing block (see Fig. 3.1).

From a visual inspection of the 20 ms frame it is possible to determine if the frame is fully voiced, partially voiced or completely unvoiced. Examples of these are given in the Figs. 3.1-3.3.

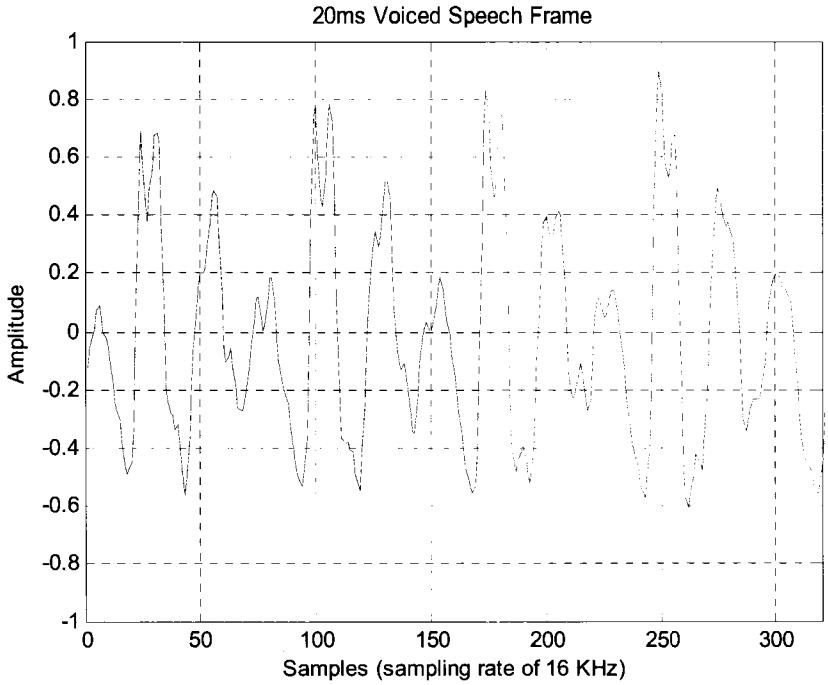

Figure 3-3 Sample of Voiced Speech Frame.

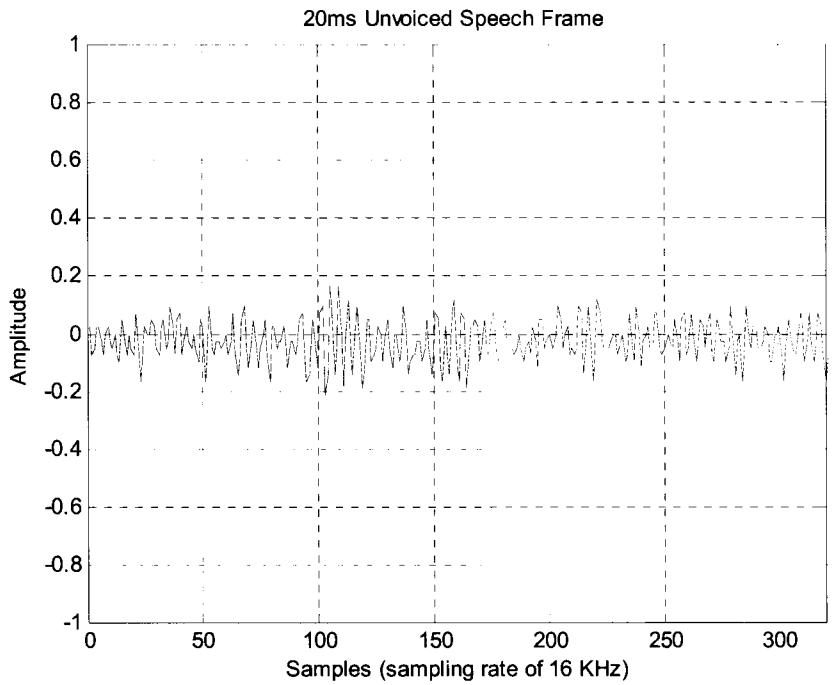

Figure 3-4 Sample of Un-Voiced Speech Frame.

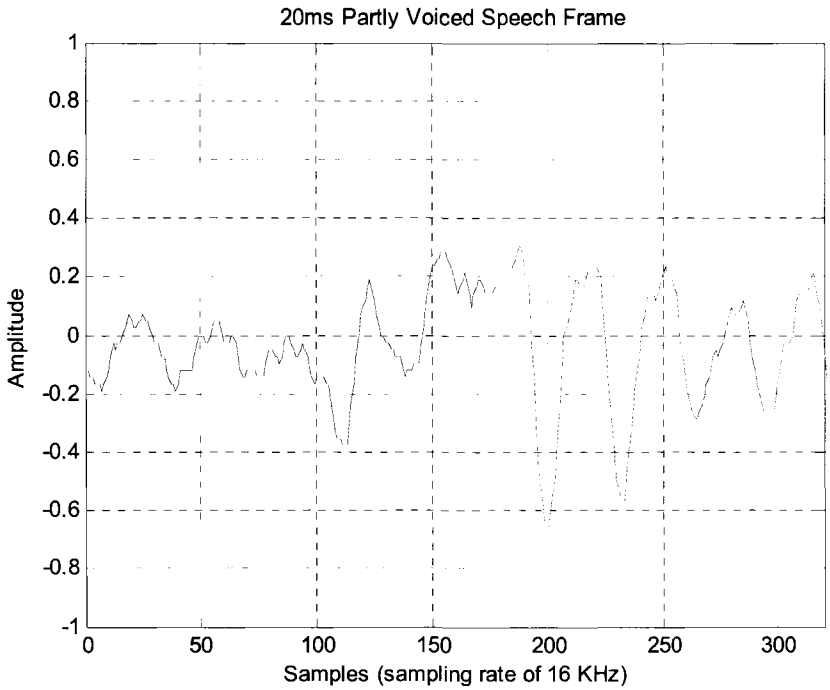

Figure 3-5 Sample of Partially Voiced Speech Frame.

Table 3.2 compares the results obtained from the voicing algorithm with the results obtained, for the same frames, through a visual inspection. A one means that the frame has been labeled as voiced and a zero as un-voiced. The columns labeled "Visual" and "Gain" show the results from visual inspection and the voicing algorithm, respectively.

Table 3-2 Performance Results for Voicing Algorithms.

Frame	Visual	Gain	Frame	Visual	Gain	Frame	Visual	Gain	Frame	Visual	Gain
1	0	0	26	1	1	51	0	0	76	1	1
2	0	0	27	1	1	52	0	0	77	1	1
3	0	0	28	1	1	53	0	0	78	1	1
4	0	0	29	0		54	0	0	79	1	1
5	0	0	30	0		55	0	0	80	1	1
6	0	0	31	0	0	56	0	0	81	0	
7	0	0	32	0	0	57	0	0	82	0	
8	0	0	33	1		58	1	1	83	0	0
9	0	0	34	1	1	59	1	1	84	0	
10	0		35	1	1	60	1	1	85	0	0
11	0	0	36	1	1	61	1	1	86	1	
12	0		37	0	0	62	1	1	87	1	1
13	0	0	38	0		63	1	1	88	1	1
14	0		39	0	0	64	1	1	89	1	1
15	0	0	40	0	0	65	0	0	90	1	1
16	1	1	41	1	1	66	1	1	91	1	1
17	1	1	42	1	1	67	1	1	92	1	1
18	1	1	43	1	1	68	1	1	93	1	1
19	1	1	44	1	1	69	1	1	94	1	1
20	1	1	45	1	1	70	1	1	95	1	1
21	1	1	46	1	1	71	1	1	96	1	1
22	1	1	47	0		72	0		97	1	1
23	1	1	48	0	0	73	0	0	98	1	1
24	1	1	49	0	0	74	1	1	99	0	
25	1	1	50	0	0	75	1	1	100	0	

In Table 3-2, a highlighted cell means that the wrong choice has been made by the algorithm. For this set of data the voicing algorithm has an accuracy of 85 %.

A drawback of the voicing algorithm is that some frames containing unvoiced sounds, such as /s/, can be discarded. For example, if such a frame is received at the same time as a fully voiced frame, the voiced one would be selected and forwarded on.

In analyzing the frames of two signals, if one is determined to be voiced and the other one un-voiced, the voiced frame is forwarded without the need for further decoding or processing. If both frames are determined to be voiced all the parameters coded in the bit stream are extracted and mixed as described in the following sections.

3.2 Parameter Mixer Functional Block

This functional block contains the algorithms used to mix the parameters of two voiced frames. The following sections describe the design of these algorithms and present parameter specific results in order to visualize their performance. These results are obtained by comparing the parameters mixed by the parametric mixer with the corresponding parameters obtained by linearly mixing the two speech signals.

Figure 3.6 illustrates how the linearly mixed parameters are extracted. The two speech signals are first linearly added together and then passed through the G.722.2 encoder. The parameter extractor component receives the bit stream from the encoder and extracts the parameters of interest.

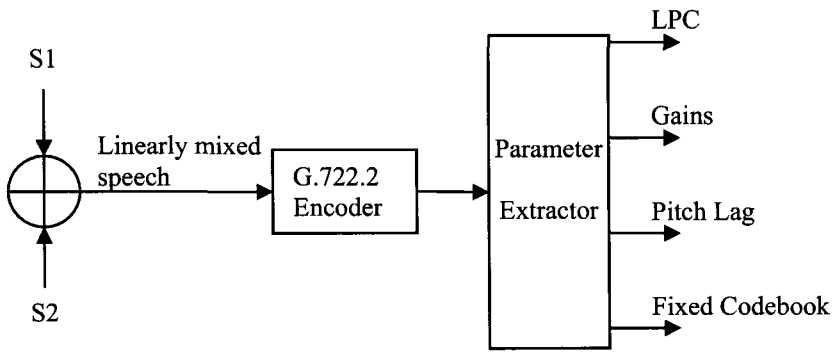

Figure 3-6 Linear Parameters Extractor.

For formal test cases and test results, performed with a Digital Speech Level Analyzer (DSLA), refer to Chapters 4 and 5.

3.2.1 Linear Prediction Coefficients Mixer

The LPCs transmitted are used to obtain the locations of the formant frequencies in the sound spectrum and therefore to reconstruct the spectral envelope of the speech frame.

Fig. 3.7 is an example of such a spectral envelope for a voiced frame, in this case the sound /o/.

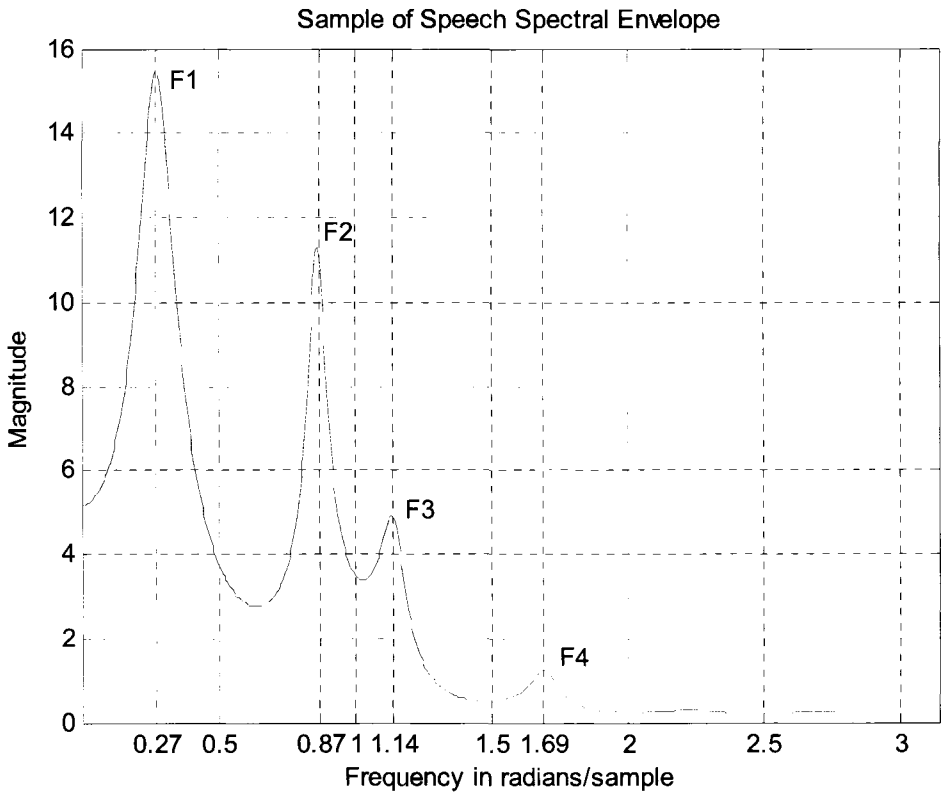

Figure 3-7 Sample of Spectral Envelope for a Speech Frame.

From Fig. 3.7, four formants are clearly visible. The G.722.2 encoder performs the analysis at a sampling rate of 12.8 kHz after filtering the speech signal through a finite impulse response (FIR) filter with cutoff frequency of 6.4 kHz. This

gives the frequency of the four formants F_1, F_2, F_3 and F_4, which, for Fig. 3.7, are at approximately 550 Hz, 1772 Hz, 2322 Hz, and 3443, Hz respectively. The LPC synthesis filter used by the encoder is a 16 pole filter. The pole-zero plot, for the signal spectrum shown is Fig. 3.7, is given in Fig. 3.8. A total of 16 poles are visible around the unit circle.

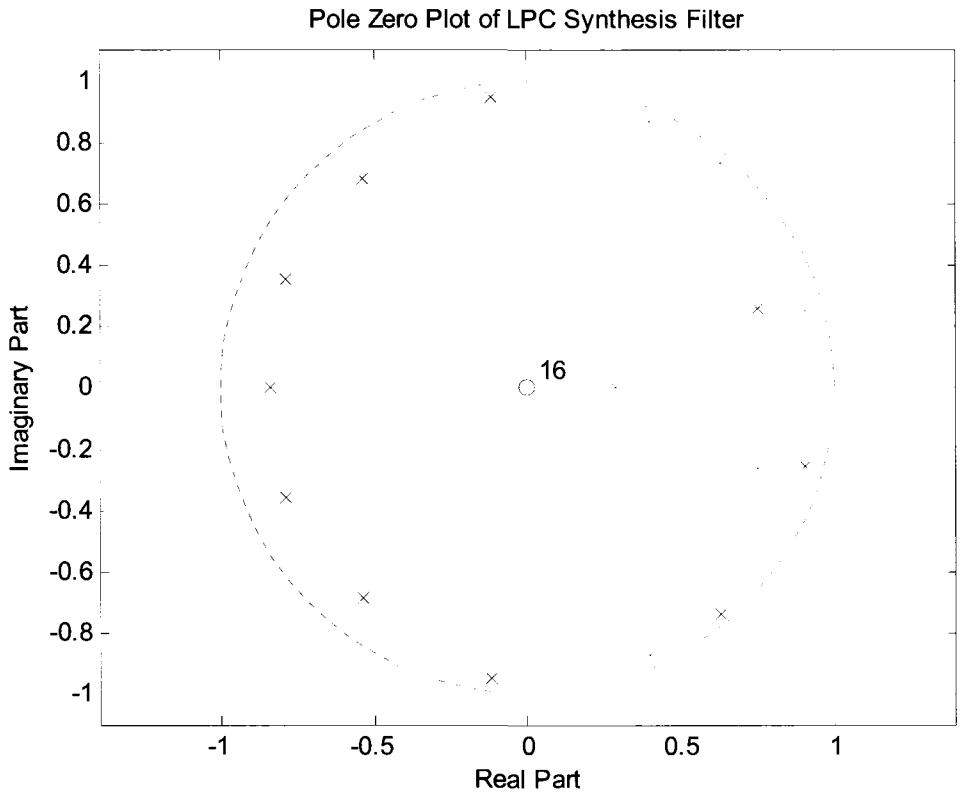

Figure 3-8 Pole-Zero Plot Sample for LPC Synthesis Filter.

The LPC mixer algorithm works as follows:

1. ISFs transmitted in the bit-stream are decoded for both speech signals *S1* and *S2* (refer to Fig. 3.1);
2. The ISFs are converted back to LPCs;

3. Prony's Method is used to find the mixed LPCs [44];
4. The mixed LPCs are converted back to ISFs for quantization;
5. The ISFs are quantized using a combination of split vector quantization and multistage vector quantization as specified by the G.722.2 standard.

The speech frames entering the LPC mixer algorithm will be either fully voiced, both sections of the frame have been labeled as voiced by the voicing algorithm, or partially voiced, only one section of the frame has been labeled as voiced, as the unvoiced frames have been filtered out by the voicing algorithm.

A fully voiced frame will have a bigger contribution on the outcome of the mixed LPCs than a partially voiced frame, thus the design decision to assign weights to the frequency responses of the two frames being mixed. In the simulations, if a frame is fully voiced a weight of 1 is attributed to its frequency response, if a frame is partially voiced a weight of 0.5 is attributed to its frequency response.

Fig. 3.9 illustrates what is described above. W_1 and W_2 are the weights by which the frequency responses for the two signals are multiplied. The weighted frequency responses are then added together and the LPCs are calculated using Prony's method.

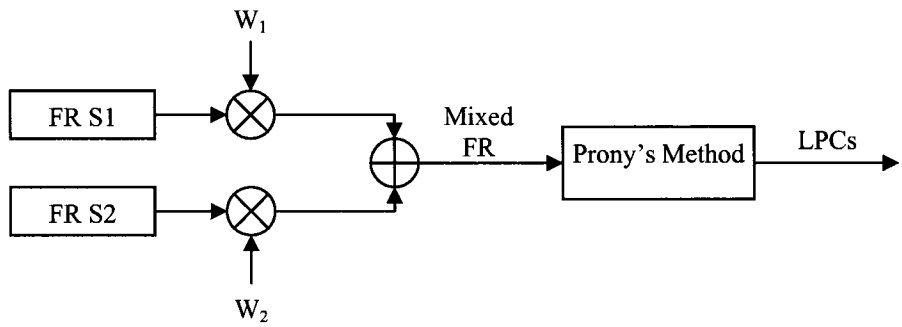

Figure 3-9 Mixing of Synthesis Filters' Frequency Responses.

Prony's Method

Prony's method is an efficient technique in estimating the coefficients of an all pole filter, given its impulse response, as it uses a least-squares method approach [44]. Fig. 3.10 illustrates how it works.

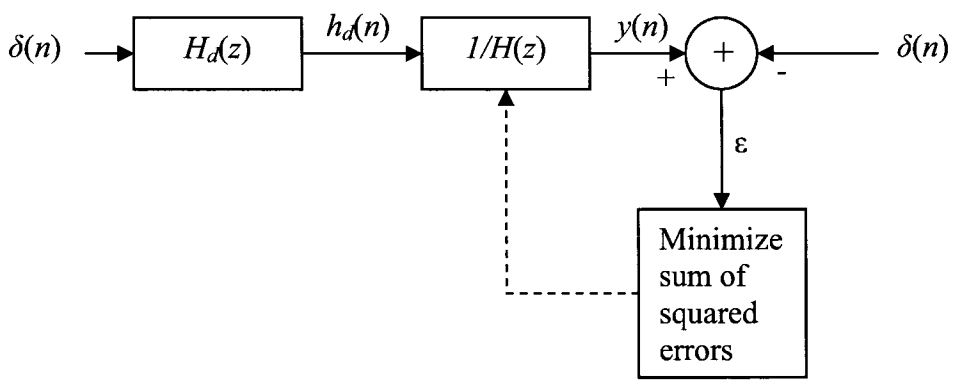

Figure 3-10 Prony's IIR Design Method [44].

$H_d(z)$ is the desired filter and $1/H(z)$ is its reciprocal. $H_d(z)$ is excited with a unit sample sequence $\delta(n)$, thus the input to the inverse system is the impulse response of $H_d(z)$. Ideally $y(n)$ should be equal to $\delta(n)$ and the error, ε, should be zero. The coefficients of the filter $H(z)$ are chosen by minimizing the sum of squares of the error sequence $\varepsilon = \sum_{n=1}^{L} y^2(n)$, where L is the length of the impulse response.

In the LPC mixing algorithm the impulse response of the mixed frequency response is calculated as follows. As mentioned earlier in this section the mixed frequency response is given by

$$H_{mix}(z) = W_1 H_1(z) + W_2 H_2(z) = \frac{W_1}{1 + \sum_{k=1}^{16} LPC_{1k} z^{-k}} + \frac{W2}{1 + \sum_{k=1}^{16} LPC_{2k} z^{-k}}$$

where H_1 and H_2 are the all pole LPC filters from signals S1 and S2. The coefficients of H_{mix} can therefore be calculated by

$$b_0 = W_1 + W_2, \; a_0 = 1 \text{ for } k = 0$$

$$\{b_k\} = W_1\{LPC_{2k}\} + W_2\{LPC_{1k}\}, \text{ for } 1 \le k \le 16$$

$$\{a_k\} = \{LPC_{1k}\} * \{LPC_{2k}\}$$

where $\{b_k\}$ and $\{a_k\}$ are the sets of numerator and denominator coefficients and * denotes convolution. Given the coefficients of H_{mix} its impulse response is found by filtering an impulse vector through H_{mix}.

Using this approach there is no need to calculate the frequency responses of the LPC filters for S1 and S2, nor the frequency response for the mixed LPC filter as knowing its coefficients is sufficient, thus saving on computational complexity.

Fig. 3.11 shows some LPCs specific results. The blue and red waveforms are the frequency spectra of the two signals being mixed. The black waveform is the frequency spectrum obtained by mixing the LPC parameters of signals 1 and 2. The green waveform is the corresponding frequency spectrum obtained from the LPCs extracted from the linearly mixed speech. Ideally the black and green waveforms should be the same but in reality this is not the case as the frequency response of the linearly mixed signal is not a linear combination of the frequency response of signal 1 and signal 2 due to the non-linearity of the G.722.2 encoder. Nonetheless this is a valid approach as far as visualizing what the mixed parameters should look like.

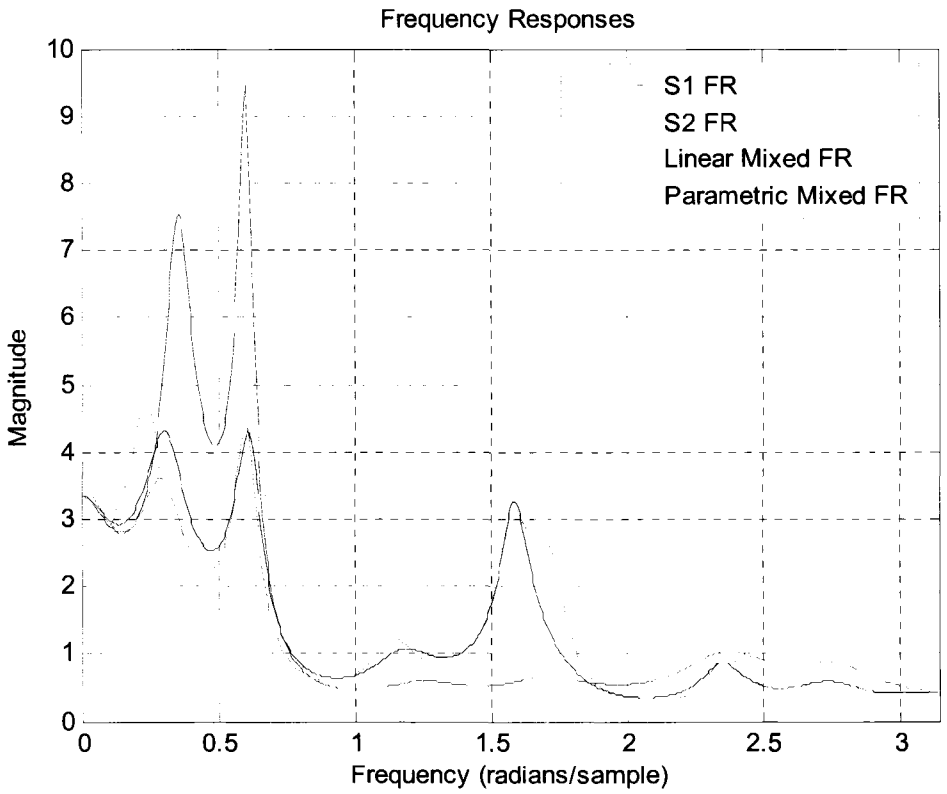

Figure 3-11 Frequency Responses for Signal 1, 2 and Mixed.

3.2.2 Pitch Lags and Gains Mixer

Pitch is arguably the most important characteristic of voiced sound. As mentioned in Chapter 2 voiced sounds are generated by forcing airflow through the glottis to the vocal tract. This flow of air is interrupted by the opening and closing of the vocal cords which produces quasi-periodic pulses of air as the excitation to the vocal tract. The rate of this opening and closing is the pitch of the sound, also known as fundamental frequency.

The pitch delay range searched by the G.722.2 encoder goes from a lag of 34 samples to 231 samples, which corresponds to 55.4 Hz to 376 Hz [25].

The G.722.2 encoder performs the search of the optimal pitch in two stages. The open loop search and the closed loop search. In the open loop search an estimate of the pitch is obtained twice per frame (every 10 ms) by calculating the correlation of the speech for each pitch lag in the search range. The lag that maximizes the correlation function is chosen as the estimated pitch lag. This estimate is used in the closed loop search to find the optimal pitch lag. The pitch lag chosen will be the one that minimizes the mean squared error between the synthesized speech and the original speech. Once the optimal pitch lag is found the pitch gain for that specific lag is calculated as described in Section 3.1.

The G.722.2 encoder computes the pitch lags and gains once for every 5 ms sub-frame. In the first and third sub-frames the delays T_1 and T_3 are found by searching a range of fourteen samples around the open-loop delay estimates T_{op1} and T_{op2}. For the second and fourth sub-frames the pitch lags T_2 and T_3 are computed by searching a range of sixteen samples around the integer pitch selected in the previous

sub-frames. This dependency between first-second pitch lag and third-fourth pitch lag is considered when implementing the pitch lags and gains mixer algorithm of the parametric mixer.

Another important aspect of the pitch lag is that it represents the long-term correlation in the voiced speech. In the presence of two voiced speech signals the long-term correlation is dominated by the one with stronger pitch energy. This is another characteristic that is exploited by the pitch lags and gains mixer algorithm. Again in this case the speech frames can be either fully voiced or partially voiced as determined from the voicing algorithm.

The pitch and gains to be transmitted are selected as follows:

- If a section from one of the speech signals is voiced while the corresponding section of the other speech signal is not, then the pitch lags and gains parameters of the voiced section are transmitted.
- If both sections of the speech signals are voiced then the pitch gains from signal one and signal two are compared. The lags and gains of the section with the greater mean pitch gain are transmitted.

In Fig. 3.12 a sample of mixed pitch lags are analyzed. The blue and red plots correspond to the pitch lags of the two signals before being mixed. The plot in black represents the parametric mixed lags and the plot in green represents the lags extracted from the linearly mixed version of the signals. From the section of pitch lags shown in the figure it can be easily concluded that signal one is always voiced while signal two is unvoiced up to approximately lag 69, it goes through a transient period from unvoiced to voiced, it is voiced between lags 75 and 100, then it goes

through another transient period and it becomes unvoiced again. Using the linearly mixed lags as metric it can be seen that the parametric mixed lags track the voiced signal quite accurately. In the portion where both signals are voiced the parametric mixed lags transmitted move from signal one to signal two as this is the one with higher pitch gain. Overall the parametric mixed lags follow the linearly mixed lags quite accurately (i.e., the black data follows the green in Fig. 3.12).

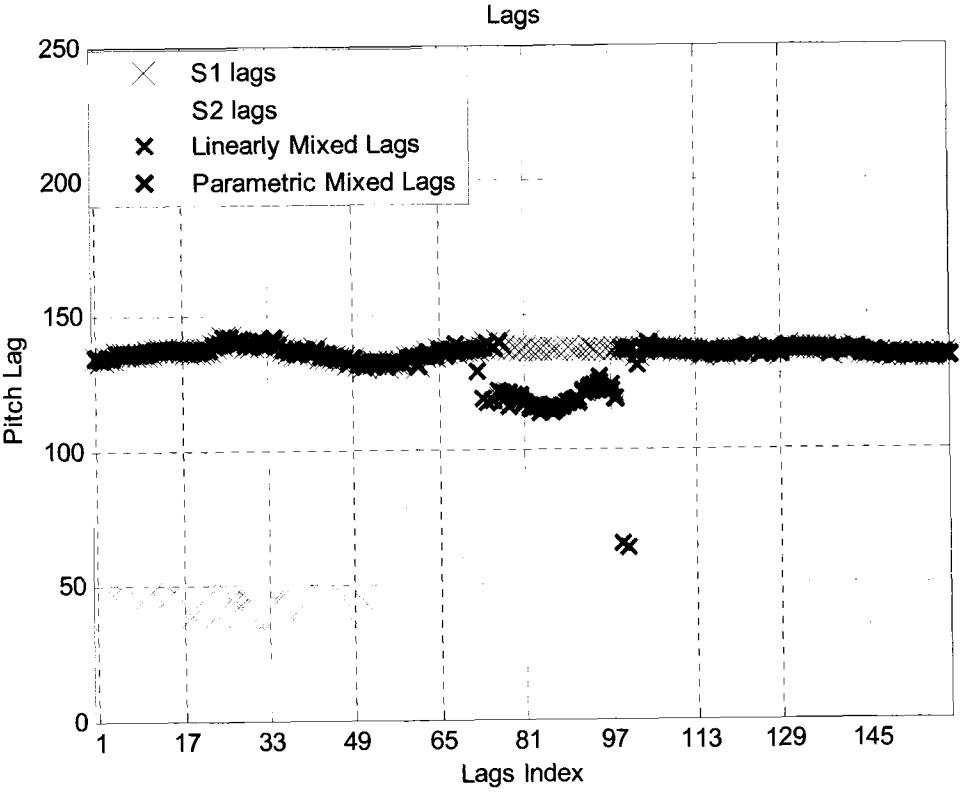

Figure 3-12 Sample of Mixed Pitch Lags.

Fig. 3.13 shows the same subset of data as Fig. 3.12 but for the pitch gains. In this case the variation between the parametric mixed gains and the linearly mixed gains is higher. This suggests that choosing the pitch gain based on the pitch lag

chosen might not be the best way of mixing this parameter. This leaves the door open to further improvements specifically with the pitch gain mixing.

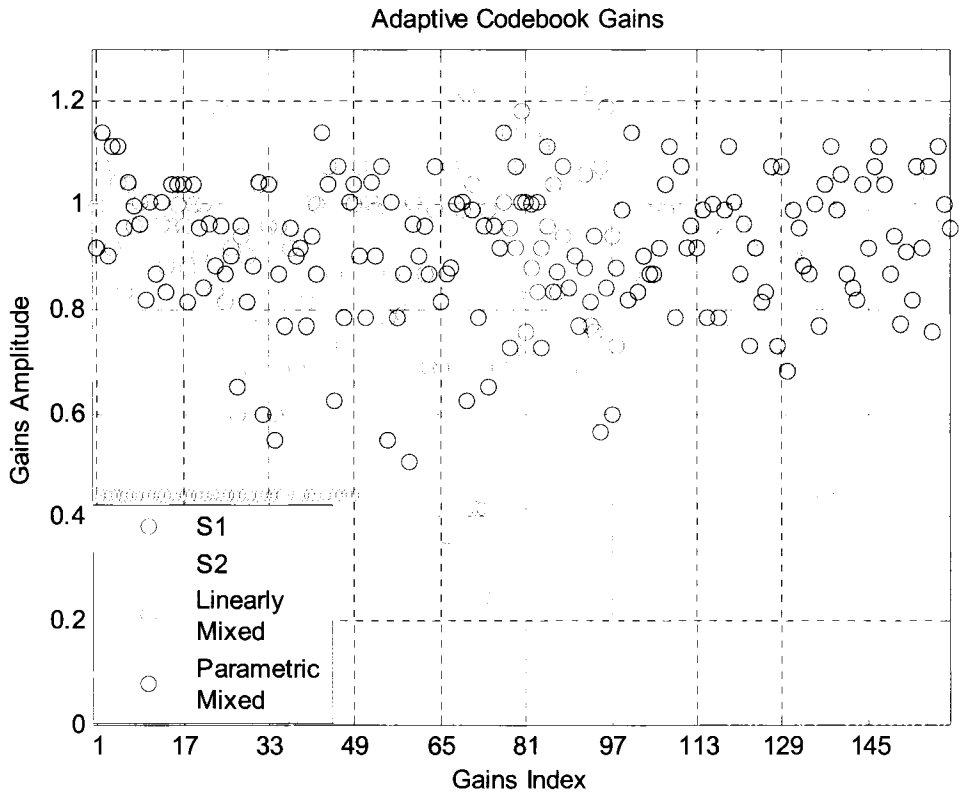

Figure 3-13 Sample of Mixed Pitch Gains.

3.2.3 Fixed Codebook Mixer

The fixed codebook component of the G.722.2 codec aims at refining the excitation signal of the LP synthesis filter by strategically inserting pulses of unity amplitude in the excitation.

The G.722.2 encoder searches the optimal positions of these pulses by minimizing the mean squared error between the weighted input speech and the

weighted synthesis speech. To simplify the search, the G.722.2 encoder updates the target signal used in the closed loop pitch search by subtracting the adaptive codebook contribution. The new target signal is given by:

$$x_2(n) = x(n) - g_p y(n), \qquad n = 0,...,63$$

This adds a level of dependency of the fixed codebook on the previous parameters, gains, pitch and LPCs, that makes it more difficult to find a suitable approach to mix the fixed codebooks of two voiced signals.

The G.722.2 codebook structure is based on interleaved single-pulse permutation design. For each sub-frame there are a total of 64 possible positions where these pulses can be placed. These 64 positions are divided into four tracks of interleaved positions with 16 positions in each track. Based on the rate used, each track can have from one to a maximum of six pulses in it where each pulse can have amplitude of 1 or -1. Table 3-3 shows the possible pulses positions for the 12.65 kbps rate, which is the rate used to encode the signals for the purpose of this thesis. For this rate there are exactly two pulses per track.

Table 3-3 Potential Positions of Individual Pulses for 12.65 kbit/s Rate.

Track	Pulse	Positions
1	i_0, i_4	0, 4, 8, 12, 16, 20, 24, 28, 32, 36, 40, 44, 48, 52, 56, 60
2	i_1, i_5	1, 5, 9, 13, 17, 21, 25, 29, 33, 37, 41, 45, 49, 53, 57, 61
3	i_2, i_6	2, 6, 10, 14, 18, 22, 26, 30, 34, 38, 42, 46, 50, 54, 58, 62
4	i_3, i_7	3, 7, 11, 15, 19, 23, 27, 31, 35, 39, 43, 47, 51, 55, 59, 63

The actual codebook index transmitted by the G.722.2 encoder represents the pulse positions and signs in each track thus there is no need for codebook storage.

The parametric mixer uses two different approaches in mixing the fixed codebooks for two voiced signals.

Fixed Codebook Mixer Algorithm 1

In the first approach, the fixed codebook transmitted by the mixer algorithm is taken from the signal from which the gains and pitch lags were chosen. For example, if, in the pitch lag and gain mixer algorithm, the parameters from signal one were chosen, then the fixed codebook would be taken from signal one as well. Figure 3.13 shows a sample for such mixing. The top two plots show the fixed codebook for a voiced frame of signal one and two. For each sub-frame of 64 samples there are exactly eight pulses since the two signals have been coded at 12.65 kbps. In the third sub-frame of signal two there is a pulse with amplitude two. This is because the encoder allows multiple pulses in one position as long as they are the same sign. Such a pulse is counted as two. The third plot, in Fig. 3.14, is the fixed codebook extracted from the linearly mixed version of the two signals for the same frame. After close scrutiny of several samples as the one shown, it has been concluded that there is no clear relationship between this fixed codebook and the ones from signal one and two. This is due to the fixed codebook being largely dependent on the previous calculated parameters. Thus a small change in those parameters can drastically vary the fixed codebook. Hence the difficulty in finding a suitable mixing solution at the parametric level without having to do full tandem decoding/encoding. The last plot in Fig. 3.14

is the parametric mixed fixed codebook which, in this case, is the same as the fixed codebook for signal 2.

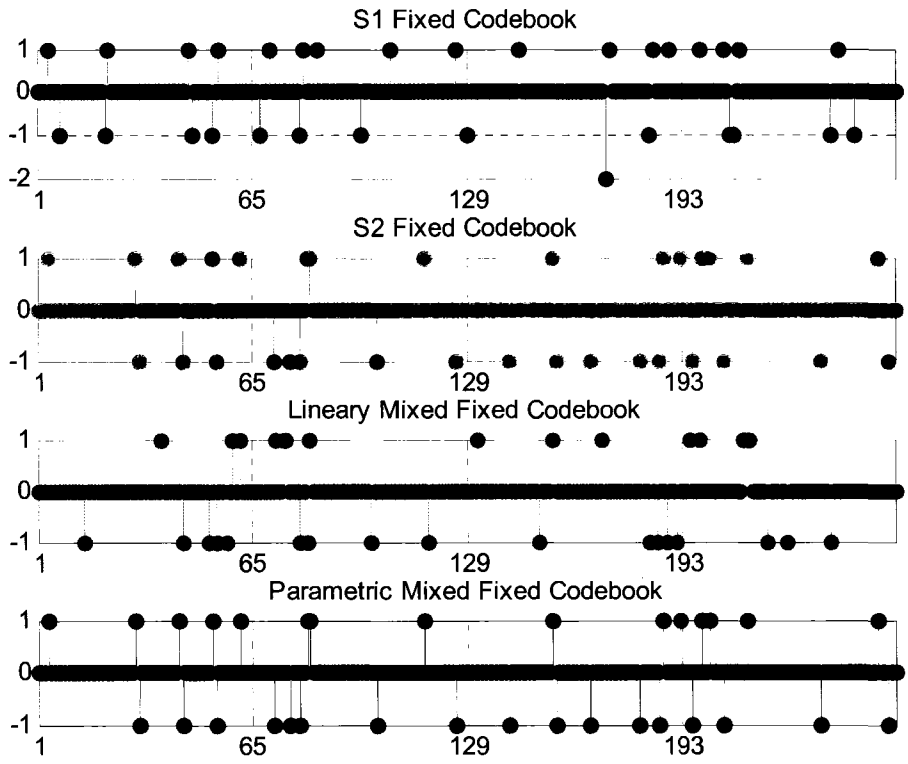

Figure 3-14 Sample of Fixed Codebook Mixing for Algorithm 1.

Fixed Codebook Mixer Algorithm 2

The second algorithm used to mix the fixed codebook exploits the capability of the G.722.2 codec to change rate at every 20 ms frame boundary. Table 3-4 shows the bit allocation per frame for 12.65 and 18.25 kbps rates.

Table 3-4 Bit Allocation per Frame for Rates 12.65 kbps and 18.25 kbps.

Rate	VAD	ISP	LTP-Filtering	Pitch Lag	Gains	Fixed Codebook	Total
12.65	1	46	4	30	28	**144**	**253**
18.25	1	46	4	30	28	**256**	**365**

As can be seen from the table, the only difference in the coding of a frame for the two rates comes from the fixed codebook encoding. While the 12.65 kbps rate encodes two pulses per track with 144 bits, the 18.25 kbps rate encodes four pulses per track with 256 bits. The second proposed algorithm adds the two fixed codebooks from signal one and two thus bringing the pulses per track to four which can be coded using the 18.25 kbps rate. In this case the parametric mixer would work at two different rates, 12.65 kbps if there is no mixing and 18.25 kbps when two voiced frames are mixed. While this algorithm has been implemented using these two rates it is not limited to them. Any other suitable pair of coding rates could be used.

Fig. 3.15 shows such a mixing. Fig 3.15 is identical to Fig. 3.14 except for the last plot which is the addition of the codebooks for signal one and two. In this case there are 16 pulses per sub-frame. If in the addition of the codebooks two pulses cancel each other out then one of the pulses is moved to the next closest position in the track as there have to be exactly four pulses per track when coding at the 18.25 kbps rate.

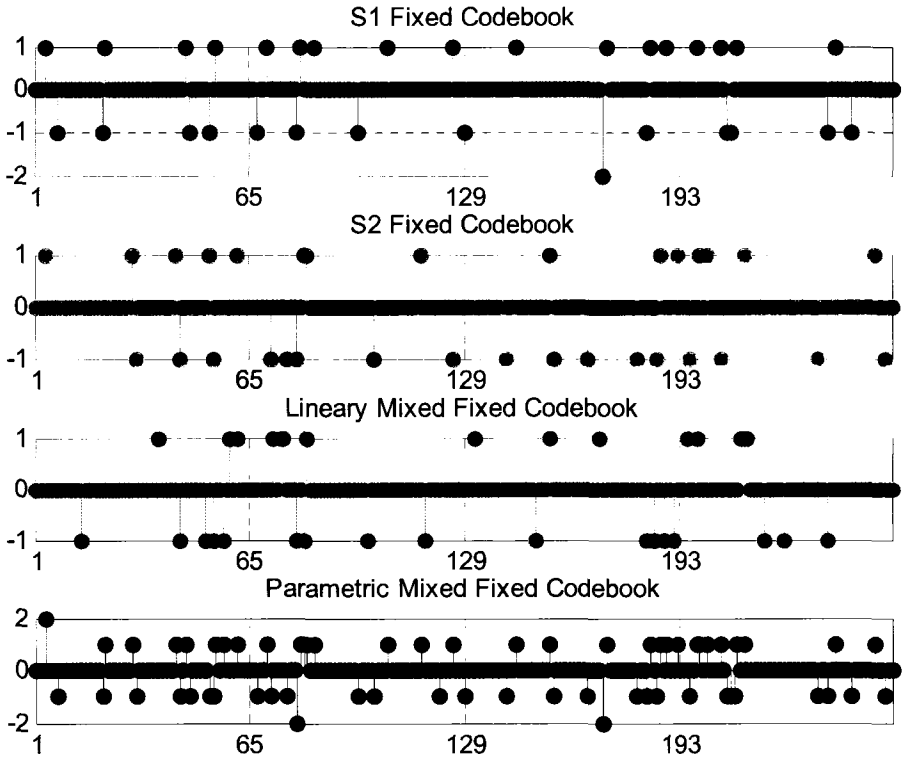

Figure 3-15 of Fixed Codebook Mixing for Algorithm 2.

Chapter 4 will describe in details how all of the mixing algorithms presented in this Chapter are tested using a Digital Speech Level Analyzer.

4 Experimental Setup

This chapter explains the process followed during testing, the tools employed and the speech wave-files used to simulate the voice conferencing scenarios.

4.1 Speech Samples and Conferencing Scenarios

The speech samples used have been taken from the TIMIT Acoustic-Phonetic Continuous Speech Corpus database [30]. This database contains broadband recordings of 630 speakers, both males and females, each reading 10 phonetically rich sentences, subdivided in 8 major dialects of American English. This database is designed to provide speech data for acoustic-phonetic studies such as speech recognition.

The TIMIT database contains phonetic and word transcriptions as well as a 16 bit 16 kHz waveform for each utterance spoken. The development of the TIMIT database was a joint effort from the Massachusetts Institute of Technology (MIT),

SRI International (SRI) and Texas Instruments (TI). The speech was recorded at TI, transcribed at MIT and prepared for CD-ROM production by the National Institute of Standards and Technology (NIST). The data on the CD is organized by dialects, 1 to 8, speaker gender, male or female, and finally by sentence ID. An example is /dr1/fcjf0/sa1.wav for sentence sa1 spoken by a female speaker of dialect region 1. The speech files used for the experiments will be listed later in this chapter.

As mention in the literature review, multi talk in a conference accounts for 6 % to 11 % of the total conferencing time. In all simulation scenarios below, except for the first one, a slightly more degraded version of the worst case is considered and multi talk is present 11 % to 20 % of the total.

Four different multi talk scenarios are considered for the voice conferencing simulations.

The first scenario is with no speech overlap such that speaker one talks first, then speaker two. Therefore there is no multi talk in this case but this simulation is useful for comparison purposes. Fig. 4.1 shows a speech sample from this scenario.

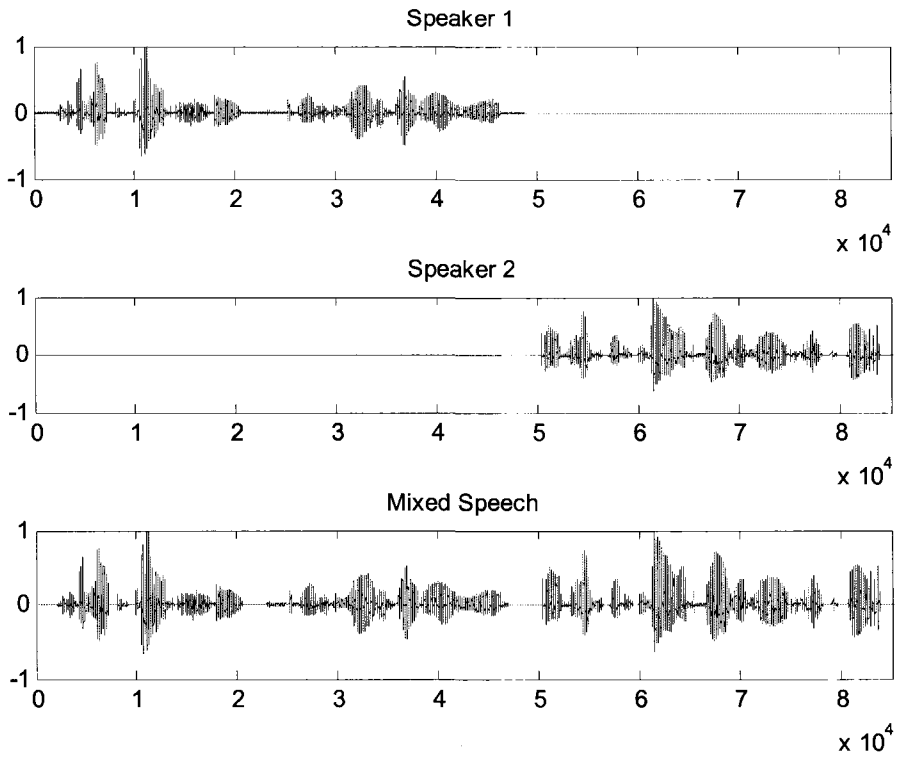

Figure 4-1 Sample of Multi-Talk Scenario 1.

The second scenario is the reinforcement case, while speaker one is talking speaker two intervenes with short expressions like "yes", "right", "no", etc. In this case, as with the following cases, multi-talk accounts for at least 11 % of the total speech sample being analyzed. A sample from this scenario is shown in Fig. 4.2.

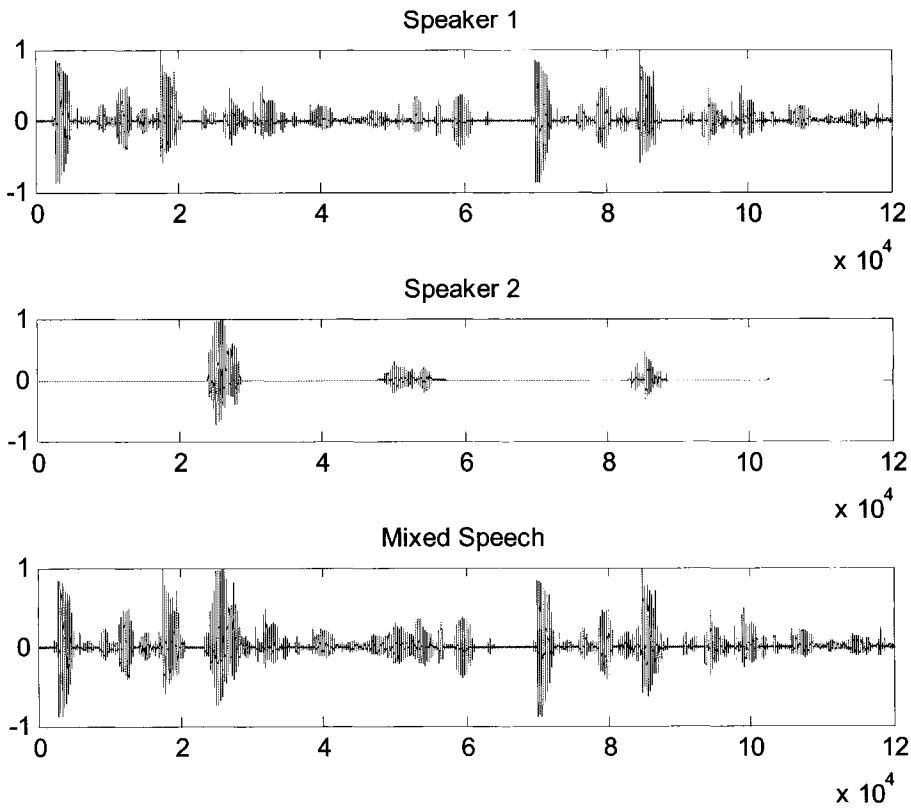

Figure 4-2 Sample of Multi Talk Scenario 2.

The third multi talk scenario is with overlapping phrases. While speaker one is talking, speaker two starts talking before speaker one is finished. A sample is shown Fig. 4.3.

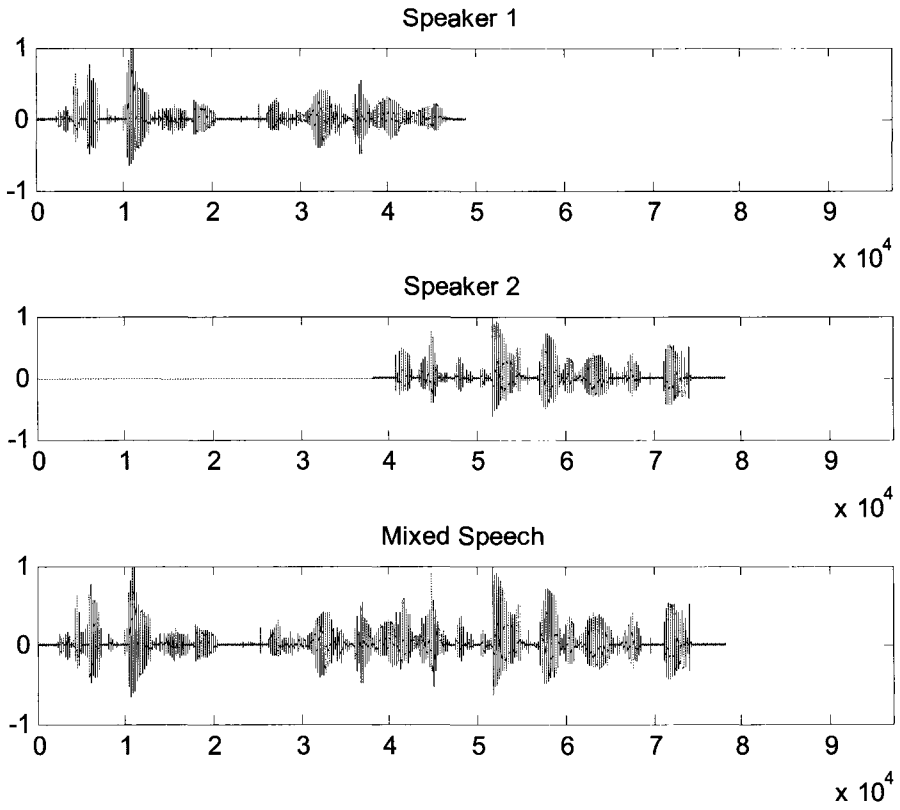

Figure 4-3 Sample of Multi-Talk Scenario 3.

The last scenario is the collision case, the two speakers start talking at the same time, then one of them backs off as shown in Fig. 4.4.

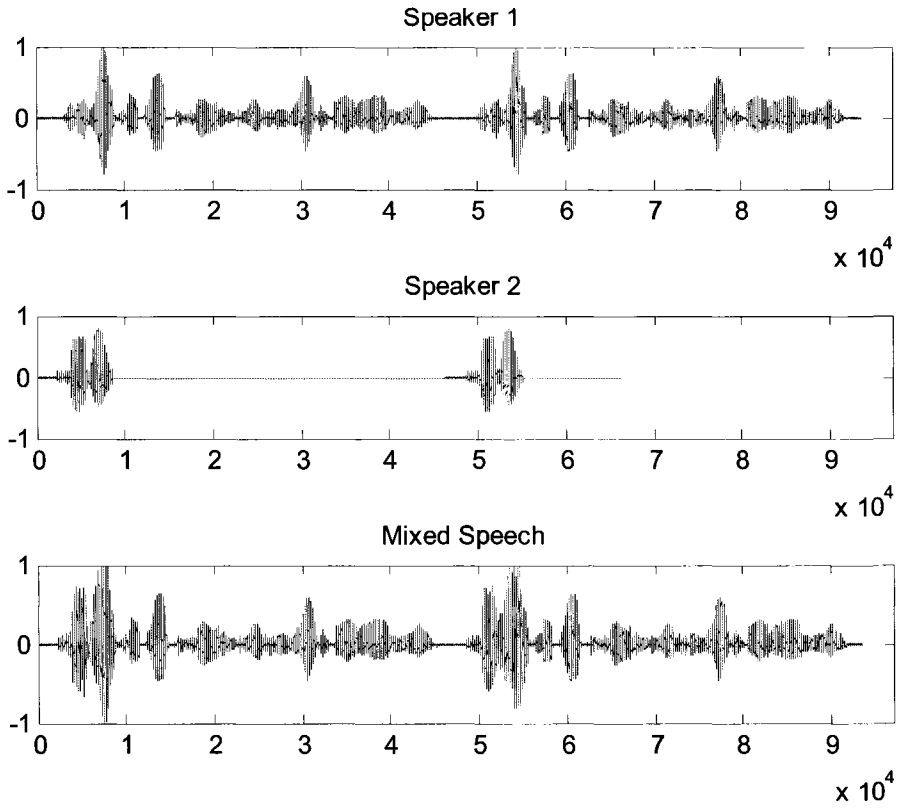

Figure 4-4 Sample of Multi-Talk Scenario 4.

For each of the scenarios above twelve test cases are run. In the first four the speakers are both female. In the next four the speakers are both male and in the last four one speaker is a male and the other is a female.

The TIMIT wave files used in the simulations are given in Table 4.1. The green background represents the female speakers while the blue the male speakers.

Table 4-1 Wave Files Used in Simulations

Test Case	Speech File for Speaker 1	Speech File for Speaker 2
T1		
T2		
T3		
T4		
T5		
T6		
T7		
T8		
T9		
T10		
T11		
T12		

Table 4-2 shows the statistics on the number of frames that did need mixing, and the number of frames that did not need mixing in each of the twelve test cases for the multi talk scenarios where there is speech overlap. The "No Mix" column represents the number of frames that did not need to be mixed as speech was only present in one frame, i.e., only one speaker talking. The "Dom" column represents the number of frames that were not mixed as one frame dominated the other one, refer to Sec. 3.1. The "Mix" column contains the number of frames that were mixed.

Table 4-2 Frame Statistics for Simulations with Speech Overlap.

Test	Multi Talk Type 2			Multi Talk Type 3			Multi Talk Type 4		
case	No Mix	Dom	Mix	No Mix	Dom	Mix	No Mix	Dom	Mix
T1	238	38	32	260	31	26	246	25	42
T2	222	47	44	190	27	10	251	21	24
T3	179	34	36	195	50	26	207	20	16
T4	181	25	22	272	48	28	174	22	42
T5	251	39	28	205	27	24	261	34	18
T6	313	51	38	260	31	12	340	21	44
T7	333	66	40	247	39	10	362	36	42
T8	251	35	26	272	56	26	252	52	8
T9	239	42	22	204	27	20	244	24	30
T10	316	58	18	228	33	6	339	30	28
T11	188	112	42	274	47	38	179	33	38
T12	265	33	14	317	63	28	239	52	18

4.2 DSLA

The speech quality of the simulated conference type scenarios is evaluated through the PAMS. The results obtained through PAMS are then converted to MOS.

This is achieved by using a DSLA developed by Malden Electronics [31] which implements PAMS.

The score calculated by PAMS will generally be within 0.5 of the MOS of that determined by a subjective test in a laboratory. PAMS returns quality scores on two different opinion scales, listening quality and listening effort as defined in [18]. They are reproduced in Table 4-3, along with the prompt that is given in a subjective test.

Table 4-3 MOS Rating.

MOS Rating	Listening Quality	Listening Effort
5	Excellent	Complete relaxation possible; no effort required
4	Good	Attention necessary; no appreciable effort required
3	Fair	Moderate effort required
2	Poor	Considerable effort required
1	Bad	No meaning understood with any feasible effort

The listening quality and listening effort scores lie between 1 and 5 and are quoted to two decimal places. The two sets of scores are usually different since they relate to different quality of subjectivity with the listening effort normally higher than the listening quality.

To obtain the PAMS scores the DSLA needs two signals, the original unprocessed signal and the degraded version of the signal from the output of the system being considered.

In the simulations the signals used are the reference signal, the degraded signal obtained through tandem mixing and the degraded signal obtained through parametric mixing.

Fig. 4.5 shows how these signals are obtained, all the coding and decoding is performed using the G.722.2 coder at 12.65 kbps rate or, in the case when the fixed codebooks in the parametric mixer are combined, at 18.25 kbps.

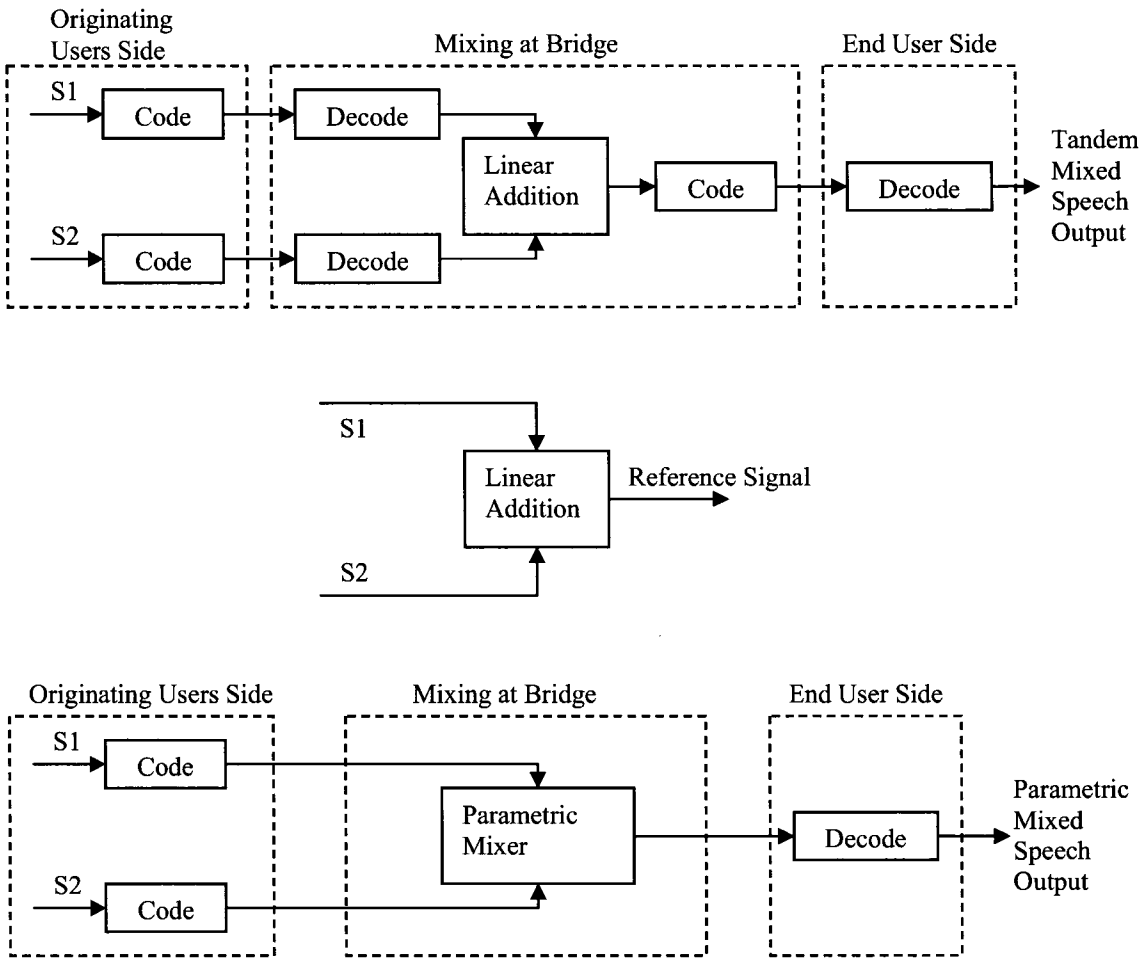

Figure 4-5 Reference Signals and Degraded Signals Used in Simulations.

In the DSLA the tandem mixed signal is compared first against the reference signal and the MOS scores recorded. Then the parametric mixed signal is compared against the reference signal and the new MOS scores are obtained. These two sets of results are then compared to each other to see if there was an increase or decrease in speech quality when the parametric mixer is used instead of the tandem mixer.

Lastly one final comparison is done. In order to understand how much of the speech degradation comes from the codec itself, and how much from the mixer, one more degraded signal is obtained and compared against the reference signal. Fig. 4.7 shows how the degraded linear mixed signal is obtained by coding and decoding the reference signal with G.722.2.

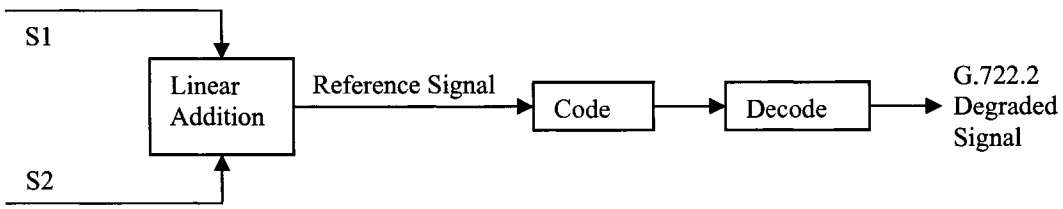

Figure 4-6 Degraded Signal of G.722.2 Codec.

Again the output of the system shown in the Fig. 4.7 is compared against the reference signal and the MOS scores are recorded.

Fig. 4.6 is a snapshot of the DSLA window showing the results of one such comparison.

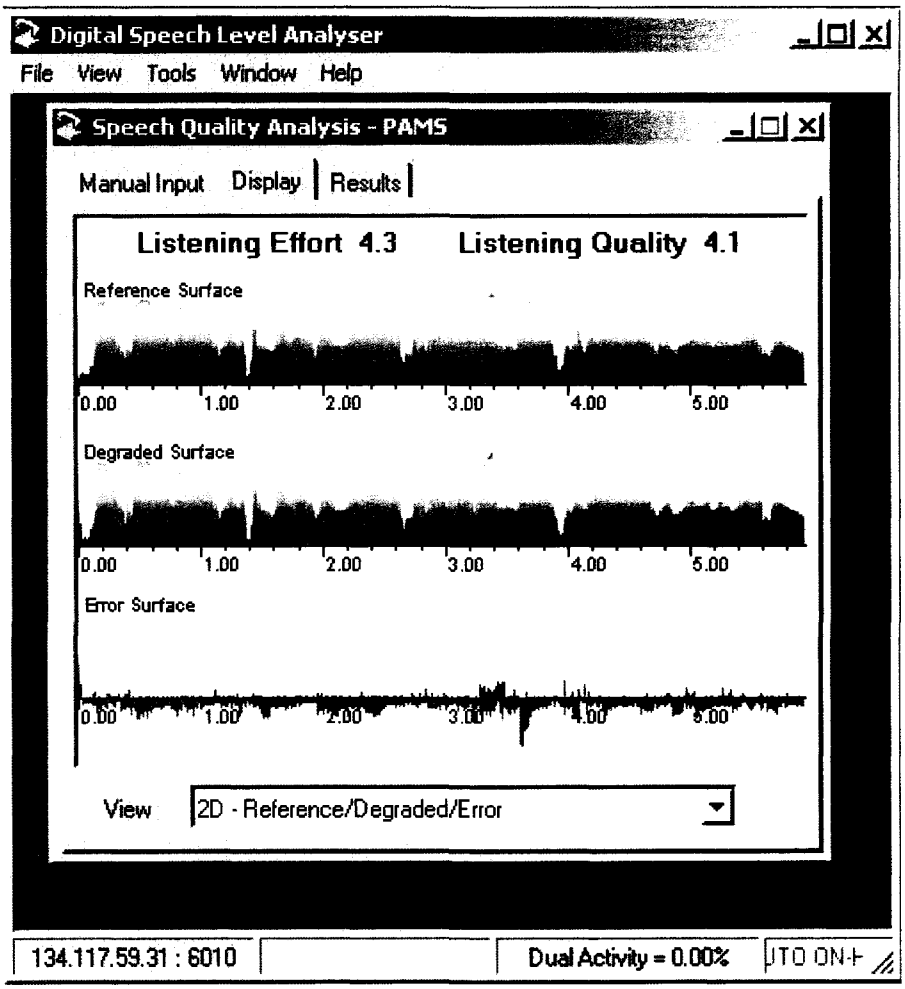

Figure 4-7 DSLA Output.

Other factors taken in consideration when constructing the speech samples to be inputted into the DSLA, are:

- No more than 15 seconds of speech is used, as the record buffer on the DSLA will lose the earlier material.
- At least 4 seconds of speech has to be present for the DSLA to produce a PAMS prediction.

4.3 Testing Procedure

An important objective during testing is to understand how exactly each component of the designed parametric mixer affects the final speech quality. To accomplish this, the mixer is divided into six functional units. These six units are:

1. Voicing algorithm;
2. LPC mixing;
3. Pitch lag mixing;
4. Gains mixing;
5. Fixed codebook mixing at 12.65 kbps;
6. Fixed codebook mixing at 18.25 kbps;

Testing is performed as follows. First the voicing algorithm alone is tested. All the test cases for the four multi talk scenarios are run, as described in the previous sections, and the MOS scores recorded. Then the remaining functional units are tested one by one with the voicing algorithm, since we need the voicing algorithm to know which frames to mix. Again the same test cases are run and the MOS scores recorded.

Fig. 4.8 shows how this is accomplished. The linearly added reference signal is coded and all the parameters, such as LPCs, lags etc., are extracted. In the parametric mixer, when two frames to be mixed are detected, only the parameters for the functional unit being tested are mixed, while all the other parameters are taken from the extracted ones of the reference signal. For example if unit three is being tested, then only the lags are mixed by the parametric mixer, all other parameters are taken

from the linearly added version. This way only one parameter mixing can be considered at a time, even though ultimately all parameters will be mixed as described in Chapter 3.

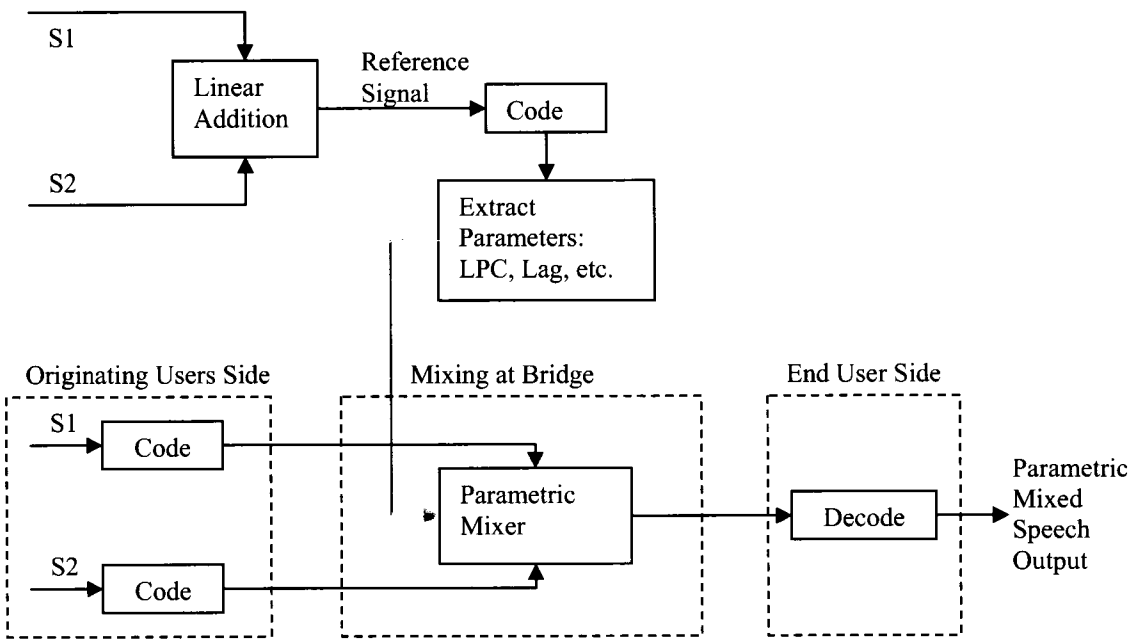

Figure 4-8 Unit Testing.

Once unit testing is completed integration testing is performed, where all parameters are mixed by the parametric mixer. Again all test cases from the four multi talk scenarios are run. Once for the parametric mixer at 12.65 kbps rate and once for the parametric mixer at 12.65/18.25 kbps rate. The MOS scores are recorded.

The next Chapter will present and discuss the results obtained from testing of the functional units and from integration testing.

5 Results

5.1 PAMS Evaluation of Degraded Signals

Table 5-1 shows the results obtained through the DSLA by comparing the reference signal with two degraded versions of itself. The first degraded output is obtained by passing the reference signal through one stage of the G.722.2 coding-decoding processing, while the second one is obtained by tandem mixing the speech. In the table the results for the four multi-talk scenarios are presented with each scenario having twelve test-cases as described in Table 4-1 in Chapter 4.

The listening effort (LE) and listening quality (LQ) scores are shown for the G.722.2 and the tandem mixed degraded signals. The average score and standard deviation for the twelve test cases is also recorded.

The results obtained by comparing the reference signal with the G.722.2 degraded one are used to understand how much of the speech degradation is due to the G.722.2 codec and how much to the mixing architecture used. On average, across all test-cases the G.722.2 codec introduces a degradation of 0.65 MOS points for the LQ and 0.45 for the LE. The tandem mixing architecture degrades the perceptual quality of the signal even further as expected. The LQ deteriorates by an extra 0.31

MOS points, bringing the overall degradation to almost a full point, 0.96, and the LE by 0.2.

Table 5-1 Speech Quality Results for G.722.2 Coder and Tandem Mixing Degraded Signals.

	Multi Talk Type 1				Multi Talk Type 2			
Test	G.722.2 Degr		Tand Mix Speech		G.722.2 Degr		Tand Mix Speech	
Case	LE	LQ	LE	LQ	LE	LQ	LE	LQ
T1	4.59	4.44	4.44	4.18	4.59	4.41	4.41	4.13
T2	4.55	4.41	4.39	4.13	4.55	4.42	4.36	4.11
T3	4.57	4.43	4.43	4.18	4.56	4.29	4.39	3.98
T4	4.58	4.42	4.4	4.13	4.62	4.43	4.47	4.2
T5	4.51	4.33	4.33	4	4.53	4.26	4.31	3.91
T6	4.57	4.47	4.39	4.11	4.61	4.45	4.43	4.13
T7	4.58	4.43	4.39	4.08	4.6	4.43	4.41	4.09
T8	4.43	4.17	4.21	3.81	4.43	4.18	4.14	3.75
T9	4.54	4.39	4.37	4.09	4.57	4.39	4.40	4.11
T10	4.62	4.48	4.43	4.14	4.64	4.49	4.46	4.16
T11	4.54	4.38	4.36	4.03	4.56	4.29	4.39	3.99
T12	4.47	4.25	4.25	3.9	4.41	4.17	4.15	3.74
AV:	**4.55**	**4.38**	**4.37**	**4.07**	**4.56**	**4.35**	**4.36**	**4.03**
SD:	**0.05**	**0.09**	**0.07**	**0.11**	**0.07**	**0.11**	**0.11**	**0.15**
	Multi Talk Type 3				**Multi Talk Type 4**			
T1	4.56	4.4	4.4	4.14	4.57	4.39	4.41	4.13
T2	4.56	4.41	4.37	4.1	4.54	4.4	4.39	4.17
T3	4.64	4.5	4.5	4.25	4.59	4.35	4.39	4.01
T4	4.61	4.42	4.38	4.05	4.58	4.42	4.4	4.14
T5	4.55	4.32	4.35	3.98	4.49	4.27	4.28	3.88
T6	4.6	4.46	4.43	4.14	4.59	4.42	4.43	4.13
T7	4.56	4.4	4.35	4.03	4.6	4.45	4.42	4.12
T8	4.4	4.12	4.11	3.66	4.45	4.21	4.17	3.75
T9	4.50	4.33	4.37	4.08	4.59	4.42	4.38	4.10
T10	4.61	4.47	4.42	4.12	4.59	4.42	4.43	4.11
T11	4.54	4.35	4.37	4.05	4.58	4.35	4.37	3.96
T12	4.45	4.21	4.15	3.75	4.47	4.23	4.19	3.82
AV:	**4.55**	**4.37**	**4.35**	**4.03**	**4.55**	**4.36**	**4.36**	**4.03**
SD:	**0.07**	**0.11**	**0.11**	**0.17**	**0.05**	**0.08**	**0.09**	**0.14**

In the following sections the MOS scores obtained for the tandem mixed speech are used as a benchmark in analyzing the perceptual speech quality of the parametric mixer. In general more emphasis is given to the LQ score in understanding the

performance of the parametric mixer as this can be directly related to quantifiable adjectives such as excellent, good, fair and poor.

5.2 Unit Testing Results

The results in this section are obtained by analyzing the main components of the parametric mixer individually. If a specific parameter is not mixed by the parametric mixer, it is extracted from the reference signal as described in Chapter 4.

5.2.1 Voicing Algorithm

Table 5-2 shows the results for the voicing algorithm. These tests aim at understanding how well the decision works on which frames to mix and which frames to not mix based on voiced speech. In this case all the parameters for the voiced frames being mixed are taken from the reference signal. The LQ and LE scores are compared against the LQ and LE score of the tandem mixed speech. In Table 5-2 this is shown under the "Degradation" columns by subtracting the scores of the tandem mixed speech from the voicing algorithm scores.

For the multi-talk type one (MTT1) case the LQ and LE are both improved by 0.31 and 0.17, on average, respectively. This is because in this case there is no talker overlap and the voiced frames are forwarded by the parametric mixer without the need for decoding and coding at the bridge, while in the tandem mixing approach the speech signals still go through the extra decoding coding processing. In the MTT2

case the voicing algorithm introduces extra degradation in the signal, 0.17 and 0.12 for LE and LQ on average and it is mostly due to test cases T9 and T11. Also the standard deviation (SD) of the scores has increased significantly from the MTT1 scenario from 0.04 to 0.24 for the LQ. For MTT3 and MTT4 the average degradation is minor and the high SD is introduced by only two cases out of the twelve, T4, T12 for MTT3 and T2, T12 for MTT4.

It can be concluded that overall the voicing algorithm works reasonably well.

Table 5-2 Speech Quality Results for Voicing Algorithm.

	Multi Talk Type 1				Multi Talk Type 2			
Test	Voicing Alg		Degradation		Voicing Alg		Degradation	
Case	LE	LQ	LE	LQ	LE	LQ	LE	LQ
T1	4.58	4.43	0.14	0.25	4.3	4.16	-0.11	0.03
T2	4.55	4.41	0.16	0.28	4.17	3.95	-0.19	-0.16
T3	4.57	4.43	0.14	0.25	4.38	4.11	-0.01	0.13
T4	4.6	4.44	0.2	0.31	4.19	3.96	-0.28	-0.24
T5	4.5	4.33	0.17	0.33	4.15	3.73	-0.16	-0.18
T6	4.57	4.47	0.18	0.36	4.36	4.2	-0.07	0.07
T7	4.51	4.39	0.12	0.31	4.33	4.1	-0.08	0.01
T8	4.39	4.19	0.18	0.38	4.09	3.77	-0.05	0.02
T9	4.52	4.38	0.15	0.29	4.13	3.76	-0.27	
T10	4.59	4.46	0.16	0.32	4.31	3.99	-0.15	-0.17
T11	4.52	4.35	0.16	0.32	3.96	3.39	-0.43	
T12	4.48	4.27	0.23	0.37	3.88	3.73	-0.27	-0.01
AV:	**4.53**	**4.38**	**0.17**	**0.31**	**4.19**	**3.90**	**-0.17**	**-0.12**
SD:	**0.06**	**0.08**	**0.03**	**0.04**	**0.16**	**0.24**	**0.12**	**0.21**
	Multi Talk Type 3				**Multi Talk Type 4**			
T1	4.46	4.33	0.06	0.19	4.32	4.13	-0.09	0
T2	4.39	4.29	0.02	0.19	4.04	3.46	-0.35	
T3	4.34	4.07	-0.16	-0.18	4.47	4.18	0.08	0.17
T4	3.94	3.78	-0.44		4.33	4.1	-0.07	-0.04
T5	4.35	4.16	0	0.18	4.2	3.92	-0.08	0.04
T6	4.34	4.26	-0.09	0.12	4.56	4.37	0.13	0.24
T7	4.28	4.09	-0.07	0.06	4.45	4.19	0.03	0.07
T8	3.88	3.53	-0.23	-0.13	3.88	3.56	-0.29	-0.19
T9	4.32	4.17	-0.05	0.09	4.38	4.17	0	0.07
T10	4.46	4.26	0.04	0.14	4.44	4.27	0.01	0.16
T11	4.29	4.13	-0.08	0.08	4.39	4.04	0.02	0.08
T12	3.66	3.37	-0.49		3.93	3.49	-0.26	
AV:	**4.23**	**4.04**	**-0.12**	**0.01**	**4.28**	**3.99**	**-0.07**	**-0.04**
SD:	**0.26**	**0.31**	**0.18**	**0.20**	**0.22**	**0.31**	**0.15**	**0.26**

The next four figures give a closer look of the test cases where the worst LQ degradation was recorded, specifically test case T11 for MTT2 and test case T2 for MTT4.

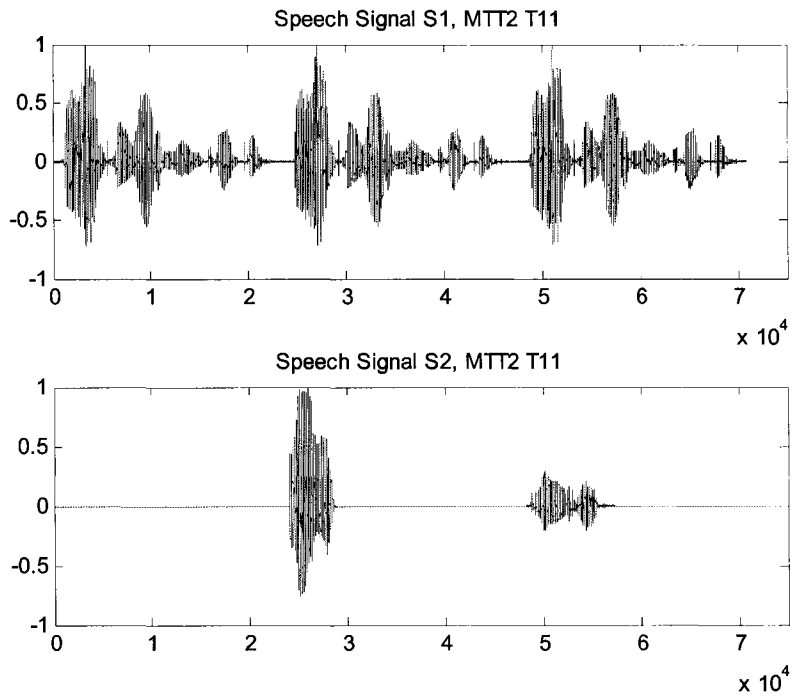

Figure 5-1 Speech Signals Being Mixed for T11 of MTT2.

Fig. 5.1 shows the speech waveforms of the two signals being mixed in T11 of MTT2 while Fig. 5.2 shows the speech waveforms of the mixed reference signal and the signal mixed by using the voicing algorithm. From Fig. 5.2 it is apparent that most of the degradation detected from the DSLA is coming from the mixing in the interval from 48000 samples to 57000 samples, approximately. In this case the wrong choice in the frames to be mixed from signal one and signal two causes large gains in the synthesized speech and therefore distortion and a decrease in listening quality.

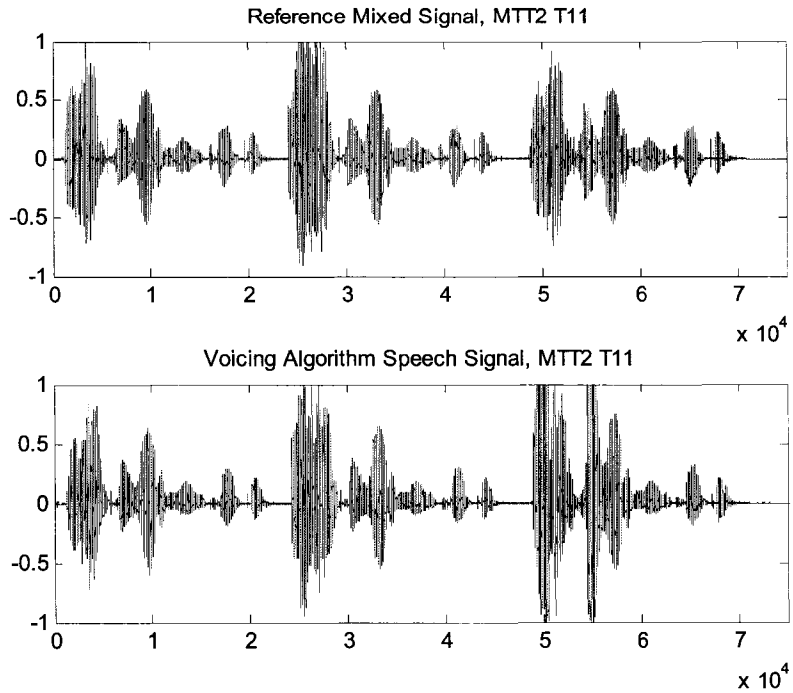

Figure 5-2 Reference Mixed Signal and Parametric Mixed Signal for T11 MTT2.

The same behaviour as the one observed in test case T11 of MTT2 can be seen in test case T2 of MTT4 and it is shown in Figs. 5.3 and 5.4. In this case the larger distortion, in the signal mixed by using the voicing algorithm, can be seen between samples 50000 and 54000, approximately, in the second subplot of Fig. 5.4.

In the following sections the test cases where a large degradation is recorded also show this type of behaviour. Future work could be used to exploit this finding and improve the parametric mixer.

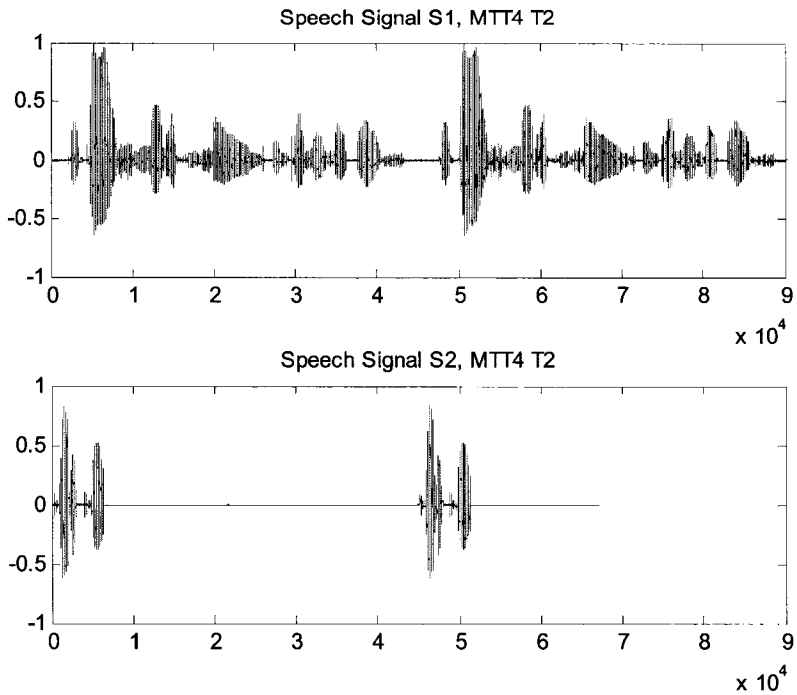

Figure 5-3 Speech Signals Being Mixed for T2 of MTT4.

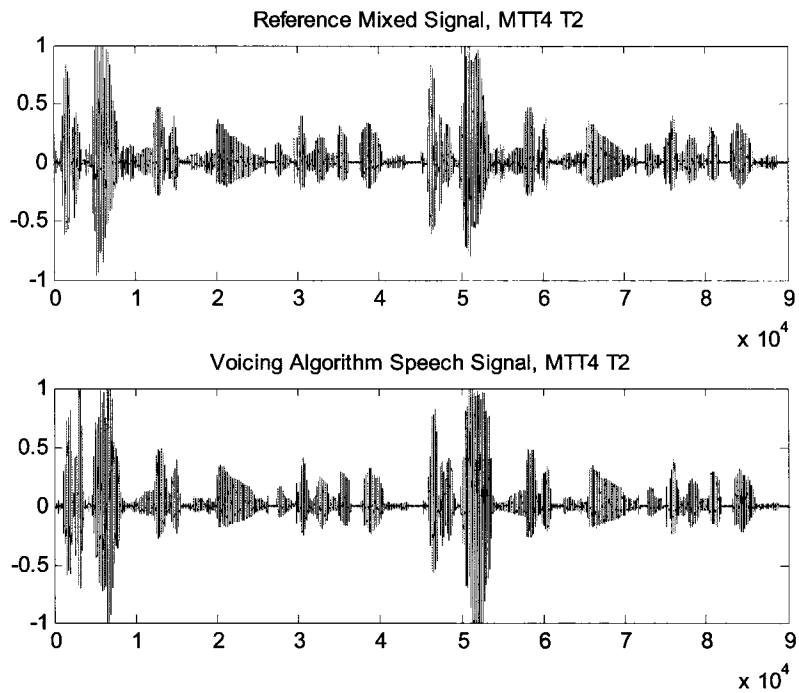

Figure 5-4 Reference Mixed Signal and Parametric Mixed Signal for T11 MTT2.

5.2.2 LPC Mixing

Table 5-3 shows the results obtained when only the LPC information is mixed once the voicing algorithm makes its decision. The MTT1 results are the same as the voicing algorithm results as for that case there is no mixing. The same applies in the sections to follow.

Table 5-3 Speech Quality Results for LPC Mixing.

	Multi Talk Type 1				Multi Talk Type 2			
Test	LPC Mixing		Degradation		LPC Mixing		Degradation	
Case	LE	LQ	LE	LQ	LE	LQ	LE	LQ
T1	4.58	4.43	0.14	0.25	4.29	4.1	-0.12	-0.03
T2	4.55	4.41	0.16	0.28	4.07	3.76	-0.29	
T3	4.57	4.43	0.14	0.25	4.26	3.81	-0.13	-0.17
T4	4.6	4.44	0.2	0.31	4.17	3.95	-0.3	-0.25
T5	4.5	4.33	0.17	0.33	4.13	3.65	-0.18	-0.26
T6	4.57	4.47	0.18	0.36	4.36	4.17	-0.07	0.04
T7	4.51	4.39	0.12	0.31	4.29	4.01	-0.12	-0.08
T8	4.39	4.19	0.18	0.38	4.05	3.7	-0.09	-0.05
T9	4.52	4.38	0.15	0.29	3.76	3.48	-0.64	
T10	4.59	4.46	0.16	0.32	4.16	3.79	-0.3	
T11	4.52	4.35	0.16	0.32	3.34	2.99	-1.05	
T12	4.48	4.27	0.23	0.37	3.86	3.67	-0.29	-0.07
AV:	**4.53**	**4.38**	**0.17**	**0.31**	**4.06**	**3.76**	**-0.30**	**-0.27**
SD:	**0.06**	**0.08**	**0.03**	**0.04**	**0.29**	**0.31**	**0.28**	**0.30**
	Multi Talk Type 3				**Multi Talk Type 4**			
T1	4.43	4.25	0.03	0.11	4.2	3.87	-0.21	-0.26
T2	4.35	4.23	-0.02	0.13	3.97	3.38	-0.42	
T3	4.34	4.07	-0.16	-0.18	4.23	3.82	-0.16	-0.19
T4	3.9	3.71		-0.34	4.35	4.08	-0.05	-0.06
T5	4.34	4.11	-0.01	0.13	4.19	3.89	-0.09	0.01
T6	4.34	4.25	-0.09	0.11	4.5	4.26	0.07	0.13
T7	4.28	4.08	-0.07	0.05	4.42	4.12	0	0
T8	3.58	3.12		-0.54	3.89	3.58	-0.28	-0.17
T9	4.31	4.1	-0.06	0.02	4.35	4.05	-0.03	-0.05
T10	4.44	4.21	0.02	0.09	4.43	4.22	0	0.11
T11	4.28	4.09	-0.09	0.04	4.27	3.82	-0.1	-0.14
T12	3.64	3.32		-0.43	3.97	3.54	-0.22	-0.28
AV:	**4.19**	**3.96**	**-0.16**	**-0.07**	**4.23**	**3.89**	**-0.12**	**-0.14**
SD:	**0.30**	**0.38**	**0.21**	**0.24**	**0.20**	**0.28**	**0.14**	**0.24**

Again, in this case the worst degradation appears in the MTT2 case. This is a direct consequence of the voicing algorithm having a higher error rate in choosing the

proper voiced frames. To understand how much further degradation is introduced by the LPC mixing the results in Table 5-3 are compared with the ones obtained from the voicing algorithm. It can be noticed that the LPC mixing introduces an extra 0.15, 0.08, and 0.1 MOS degradation for the MTT2, MTT3, and MTT4 cases respectively in LQ. Again most of the degradation is introduced from two or three test cases only which are highlighted in red in the table.

5.2.3 Lags Mixing

In Table 5-4 the results obtained from mixing the lags are shown. The degradation introduced in LQ by the lag mixing is a further 0.06, 0.08, and 0.07 MOS score for the MTT2, MTT3, and MTT4 cases. This degradation is very small therefore it can be concluded that across all scenarios the lag mixing procedure used by the parametric mixer performs reasonably well.

Again in this case most of the degradation comes from a few isolated test cases.

Table 5-4 Speech Quality Results for Lag Mixing.

	Multi Talk Type 1				Multi Talk Type 2			
Test	Lag Mixing		Degradation		Lag Mixing		Degradation	
Case	LE	LQ	LE	LQ	LE	LQ	LE	LQ
T1	4.58	4.43	0.14	0.25	4.29	4.12	-0.12	-0.01
T2	4.55	4.41	0.16	0.28	4.1	3.82	-0.26	-0.29
T3	4.57	4.43	0.14	0.25	4.31	3.96	-0.08	-0.02
T4	4.6	4.44	0.2	0.31	4.16	3.93	-0.31	-0.27
T5	4.5	4.33	0.17	0.33	4.14	3.68	-0.17	-0.23
T6	4.57	4.47	0.18	0.36	4.36	4.16	-0.07	0.03
T7	4.51	4.39	0.12	0.31	4.32	4.08	-0.09	-0.01
T8	4.39	4.19	0.18	0.38	4.09	3.76	-0.05	0.01
T9	4.52	4.38	0.15	0.29	4.00	3.54	-0.4	
T10	4.59	4.46	0.16	0.32	4.32	3.98	-0.14	-0.18
T11	4.52	4.35	0.16	0.32	3.93	3.38	-0.46	
T12	4.48	4.27	0.23	0.37	3.87	3.69	-0.28	-0.05
AV:	**4.53**	**4.38**	**0.17**	**0.31**	**4.16**	**3.84**	**-0.20**	**-0.18**
SD:	**0.06**	**0.08**	**0.03**	**0.04**	**0.17**	**0.24**	**0.14**	**0.22**
	Multi Talk Type 3				**Multi Talk Type 4**			
T1	4.43	4.23	0.03	0.09	4.3	4.05	-0.11	-0.08
T2	4.33	4.14	-0.04	0.04	4.01	3.45	-0.38	
T3	4.32	4.02	-0.18	-0.23	4.45	4.11	0.06	0.1
T4	3.89	3.7		-0.35	4.27	3.92	-0.13	-0.22
T5	4.35	4.12	0	0.14	4.18	3.9	-0.1	0.02
T6	4.31	4.23	-0.12	0.09	4.5	4.23	0.07	0.1
T7	4.27	4.08	-0.08	0.05	4.43	4.12	0.01	0
T8	3.86	3.49	-0.25	-0.17	3.86	3.53	-0.31	-0.22
T9	4.33	4.12	-0.04	0.04	4.33	4.02	-0.05	-0.08
T10	4.47	4.28	0.05	0.16	4.41	4.21	-0.02	0.1
T11	4.16	3.87	-0.21	-0.18	4.37	3.98	0	0.02
T12	3.62	3.28		-0.47	3.93	3.49	-0.26	
AV:	**4.20**	**3.96**	**-0.16**	**-0.07**	**4.25**	**3.92**	**-0.10**	**-0.11**
SD:	**0.26**	**0.32**	**0.19**	**0.21**	**0.21**	**0.28**	**0.15**	**0.24**

5.2.4 Gains Mixing

Table 5-5 shows the results obtained when mixing the gains. The degradation introduced in this case is the most significant among all parameters. The additional degradation introduced for the LQ in MTT2, MTT3 and MTT4 is a further 0.28, 0.15, and 0.22, respectively. This demonstrates that the mixing method used for the gains

is not the most reliable and that the proper choice of gain significantly impacts the perceptual quality of the speech.

Table 5-5 Speech Quality Results for Gain Mixing.

	Multi Talk Type 1				Multi Talk Type 2			
Test	Gain Mixing		Degradation		Gain Mixing		Degradation	
Case	LE	LQ	LE	LQ	LE	LQ	LE	LQ
T1	4.58	4.43	0.14	0.25	4.17	3.96	-0.24	-0.17
T2	4.55	4.41	0.16	0.28	4.15	3.89	-0.21	-0.22
T3	4.57	4.43	0.14	0.25	4.01	3.68	-0.38	
T4	4.6	4.44	0.2	0.31	4.04	3.8	-0.43	
T5	4.5	4.33	0.17	0.33	3.71	3.27	-0.6	
T6	4.57	4.47	0.18	0.36	4.3	4.06	-0.13	-0.07
T7	4.51	4.39	0.12	0.31	4.18	3.92	-0.23	-0.17
T8	4.39	4.19	0.18	0.38	3.82	3.49	-0.32	-0.26
T9	4.52	4.38	0.15	0.29	4.08	3.78	-0.32	
T10	4.59	4.46	0.16	0.32	4.4	4.21	-0.06	0.05
T11	4.52	4.35	0.16	0.32	2.7	1.73	-1.69	
T12	4.48	4.27	0.23	0.37	3.88	3.7	-0.27	-0.04
AV:	**4.53**	**4.38**	**0.17**	**0.31**	**3.95**	**3.62**	**-0.41**	**-0.40**
SD:	**0.06**	**0.08**	**0.03**	**0.04**	**0.44**	**0.65**	**0.43**	**0.61**
	Multi Talk Type 3				**Multi Talk Type 4**			
T1	4.27	4.12	-0.13	-0.02	4.16	3.79	-0.25	
T2	4.34	4.23	-0.03	0.13	4.14	3.84	-0.25	
T3	4.38	4.14	-0.12	-0.11	4.16	3.93	-0.23	-0.08
T4	3.75	3.52	-0.63		3.98	3.45	-0.42	
T5	4.3	4.08	-0.05	0.1	4.12	3.89	-0.16	0.01
T6	4.36	4.26	-0.07	0.12	4.26	4.09	-0.17	-0.04
T7	4.24	4.04	-0.11	0.01	4.35	4.08	-0.07	-0.04
T8	3.5	3.18	-0.61		3.87	3.55	-0.3	-0.2
T9	4.28	4.11	-0.09	0.03	4.26	4.03	-0.12	-0.07
T10	4.46	4.27	0.04	0.15	4.34	4.15	-0.09	0.04
T11	4.06	3.73	-0.31		3.74	3.24	-0.63	
T12	3.38	3.01	-0.77		3.67	3.17	-0.52	
AV:	**4.11**	**3.89**	**-0.24**	**-0.14**	**4.09**	**3.77**	**-0.27**	**-0.26**
SD:	**0.36**	**0.43**	**0.27**	**0.30**	**0.23**	**0.34**	**0.17**	**0.28**

5.2.5 Fixed Codebook Mixing

Table 5-6 and 5-7 show the results obtained when mixing the fixed codebook with algorithm 1 and algorithm 2 respectively. The degradation introduced by the

first algorithm on the LQ is an average of 0.05, 0.04, 0.06 for the three different multi-talk scenarios thus it does not significantly impact the speech quality. For the second algorithm the degradation introduced is slightly worse 0.12, 0.15, and 0.22.

Table 5-6 Speech Quality Results for Fixed Codebook Mixing Algorithm 1.

	Multi Talk Type 1				Multi Talk Type 2			
Test	Fixed CB Mixing		Degradation		Fixed CB Mixing		Degradation	
Case	LE	LQ	LE	LQ	LE	LQ	LE	LQ
T1	4.58	4.43	0.14	0.25	4.27	4.06	-0.14	-0.07
T2	4.55	4.41	0.16	0.28	4.21	3.97	-0.15	-0.14
T3	4.57	4.43	0.14	0.25	4.27	3.93	-0.12	-0.05
T4	4.6	4.44	0.2	0.31	4.18	3.9	-0.29	-0.3
T5	4.5	4.33	0.17	0.33	4.13	3.73	-0.18	-0.18
T6	4.57	4.47	0.18	0.36	4.29	4.07	-0.14	-0.06
T7	4.51	4.39	0.12	0.31	4.34	4.1	-0.07	0.01
T8	4.39	4.19	0.18	0.38	4.01	3.69	-0.13	-0.06
T9	4.52	4.38	0.15	0.29	4.16	3.82	-0.24	
T10	4.59	4.46	0.16	0.32	4.4	4.17	-0.06	0.01
T11	4.52	4.35	0.16	0.32	3.74	3.08	-0.65	
T12	4.48	4.27	0.23	0.37	3.9	3.73	-0.25	-0.01
AV:	**4.53**	**4.38**	**0.17**	**0.31**	**4.16**	**3.85**	**-0.20**	**-0.17**
SD:	**0.06**	**0.08**	**0.03**	**0.04**	**0.19**	**0.29**	**0.16**	**0.26**
	Multi Talk Type 3				**Multi Talk Type 4**			
T1	4.44	4.27	0.04	0.13	4.26	4.04	-0.15	-0.09
T2	4.37	4.25	0	0.15	4.14	3.68	-0.25	
T3	4.36	4.1	-0.14	-0.15	4.43	4.11	0.04	0.1
T4	3.88	3.67	-0.5		4.2	3.82	-0.2	
T5	4.32	4.09	-0.03	0.11	4.19	3.92	-0.09	0.04
T6	4.32	4.22	-0.11	0.08	4.49	4.24	0.06	0.11
T7	4.26	4.07	-0.09	0.04	4.41	4.13	-0.01	0.01
T8	3.86	3.54	-0.25	-0.12	3.86	3.54	-0.31	-0.21
T9	4.32	4.15	-0.05	0.07	4.33	4.08	-0.05	-0.02
T10	4.46	4.28	0.04	0.16	4.39	4.2	-0.04	0.09
T11	4.22	4.01	-0.15	-0.04	4.33	3.92	-0.04	-0.04
T12	3.64	3.35	-0.51		3.91	3.47	-0.28	
AV:	**4.20**	**4.00**	**-0.15**	**-0.03**	**4.25**	**3.93**	**-0.11**	**-0.10**
SD:	**0.26**	**0.31**	**0.19**	**0.20**	**0.20**	**0.25**	**0.13**	**0.20**

In comparing the results for the two algorithms it is interesting to point out how for some test cases one algorithm has a much better performance than the other. For example in T11, for the MTT2 scenario, algorithm one introduces a degradation of 0.91 while algorithm 2 introduces a degradation of 0.45. In T8 of MTT3 algorithm

one has the better performance with a degradation of 0.12 compared to 0.52 of algorithm 2. The overall performance of the parametric mixer could be improved if, during mixing, the fixed codebook algorithm with the best performance is always chosen. Further investigation would have to be done to understand when one algorithm would be preferable to the other.

Table 5-7 Speech Quality Results for Fixed Codebook Mixing Algorithm 2.

	Multi Talk Type 1				Multi Talk Type 2			
Test	Fixed CB Mix 2		Degradation		Fixed CB Mix 2		Degradation	
Case	LE	LQ	LE	LQ	LE	LQ	LE	LQ
T1	4.58	4.43	0.14	0.25	4.15	3.97	-0.26	-0.16
T2	4.55	4.41	0.16	0.28	3.65	3.32	-0.71	
T3	4.57	4.43	0.14	0.25	4.31	3.91	-0.08	-0.07
T4	4.6	4.44	0.2	0.31	4.02	3.98	-0.45	-0.22
T5	4.5	4.33	0.17	0.33	3.87	3.41	-0.44	
T6	4.57	4.47	0.18	0.36	4.22	4	-0.21	-0.13
T7	4.51	4.39	0.12	0.31	4.27	4.09	-0.14	0
T8	4.39	4.19	0.18	0.38	3.92	3.6	-0.22	-0.15
T9	4.52	4.38	0.15	0.29	4.11	3.85	-0.29	-0.26
T10	4.59	4.46	0.16	0.32	4.36	4.13	-0.1	-0.03
T11	4.52	4.35	0.16	0.32	4.09	3.54	-0.3	
T12	4.48	4.27	0.23	0.37	3.72	3.63	-0.43	-0.11
AV:	**4.53**	**4.38**	**0.17**	**0.31**	**4.06**	**3.79**	**-0.30**	**-0.24**
SD:	**0.06**	**0.08**	**0.03**	**0.04**	**0.23**	**0.27**	**0.18**	**0.23**
	Multi Talk Type 3				**Multi Talk Type 4**			
T1	4.22	4.05	-0.18	-0.09	3.87	3.67	-0.54	-0.46
T2	4	3.85	-0.37	-0.25	4.35	4.05	-0.04	-0.12
T3	4.24	4.11	-0.26	-0.14	4.34	3.94	-0.05	-0.07
T4	3.68	3.61	-0.7		3.98	3.6	-0.42	
T5	4.19	4	-0.16	0.02	3.92	3.73	-0.36	-0.15
T6	4.08	4.11	-0.35	-0.03	4.46	4.16	0.03	0.03
T7	4.04	3.93	-0.31	-0.1	4.26	3.88	-0.16	-0.24
T8	3.48	3.14	-0.63		3.58	3.34	-0.59	
T9	4	4.14	-0.37	0.06	4.14	3.84	-0.24	-0.26
T10	4.52	4.39	0.1	0.27	4.3	4.11	-0.13	0
T11	4.17	4.03	-0.2	-0.02	4.06	3.58	-0.31	-0.38
T12	3.44	3.26	-0.71		3.78	3.34	-0.41	
AV:	**4.01**	**3.89**	**-0.35**	**-0.14**	**4.09**	**3.77**	**-0.27**	**-0.26**
SD:	**0.32**	**0.37**	**0.24**	**0.24**	**0.27**	**0.28**	**0.20**	**0.20**

The histogram in Fig. 5.1 summarizes the results discussed in the previous sections. The values shown are the average speech degradation across multi-talk scenarios two, three and four for the LQ. The first multi talk scenario has not been included as there is no actual parametric mixing in that case.

As already highlighted previously the worst performance among the parameter mixing algorithms comes from the gain mixing at 0.22. The degradation coming from the LPC mixing is also significant but it is on average half the degradation coming from the gain mixing. The fixed codebook mixing algorithm one is in general preferable to algorithm two. The performance obtained from the remaining components is quite acceptable from a perceptual speech quality perspective.

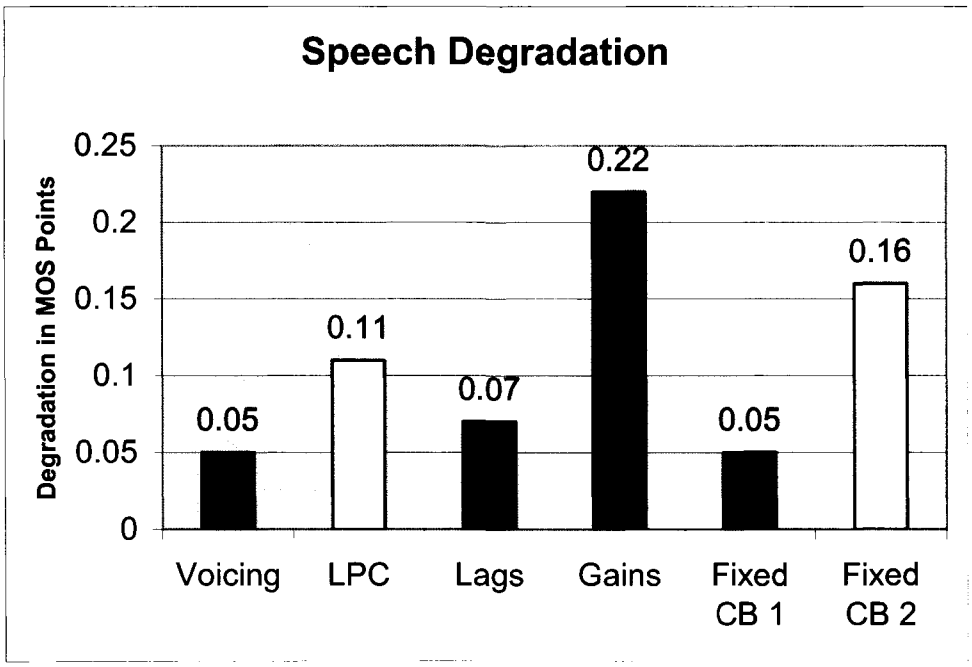

Figure 5-5 Average Speech Degradation Across all Parametric Mixer's Algorithms.

5.3 Integration Testing Results

The results presented in this section reflect the MOS scores when all parameters are mixed by the parametric mixer. This would be the case for a full operational mixing at a bridge as described earlier.

Table 5-8 Speech Quality Results for Parametric Mixer at 12.65 kbps rate.

	Multi Talk Type 1				Multi Talk Type 2			
Test	12.65 Mixer		Degradation		12.65 Mixer		Degradation	
Case	LE	LQ	LE	LQ	LE	LQ	LE	LQ
T1	4.58	4.43	0.14	0.25	4.2	3.99	-0.21	-0.14
T2	4.55	4.41	0.16	0.28	3.96	3.61	-0.4	
T3	4.57	4.43	0.14	0.25	3.98	3.65	-0.41	-0.33
T4	4.6	4.44	0.2	0.31	3.96	3.75	-0.51	
T5	4.5	4.33	0.17	0.33	3.77	3.39	-0.54	
T6	4.57	4.47	0.18	0.36	4.26	4.02	-0.17	-0.11
T7	4.51	4.39	0.12	0.31	4.19	3.95	-0.22	-0.14
T8	4.39	4.19	0.18	0.38	3.8	3.49	-0.34	-0.26
T9	4.52	4.38	0.15	0.29	4.12	3.88	-0.28	-0.23
T10	4.59	4.46	0.16	0.32	4.42	4.27	-0.04	0.11
T11	4.52	4.35	0.16	0.32	2.53	1.73	-1.86	
T12	4.48	4.27	0.23	0.37	3.86	3.65	-0.29	-0.09
AV:	**4.53**	**4.38**	**0.17**	**0.31**	**3.92**	**3.62**	**-0.44**	**-0.41**
SD:	**0.06**	**0.08**	**0.03**	**0.04**	**0.48**	**0.64**	**0.47**	**0.61**
	Multi Talk Type 3				**Multi Talk Type 4**			
T1	4.28	4.13	-0.12	-0.01	3.49	3.19	-0.92	
T2	4.25	4.14	-0.12	0.04	3.7	3.31	-0.69	
T3	4.33	4.07	-0.17	-0.18	4.2	3.9	-0.19	-0.11
T4	3.74	3.52	-0.64		4.05	3.6	-0.35	
T5	4.2	3.99	-0.15	0.01	4.09	3.84	-0.19	-0.04
T6	4.33	4.24	-0.1	0.1	4.15	3.92	-0.28	-0.21
T7	4.22	4.02	-0.13	-0.01	4.35	4.06	-0.07	-0.06
T8	3.66	3.34	-0.45	-0.32	3.84	3.51	-0.33	-0.24
T9	4.19	3.97	-0.18	-0.11	4.28	4.02	-0.1	-0.08
T10	4.43	4.25	0.01	0.13	4.21	3.97	-0.22	-0.14
T11	3.92	3.53	-0.45		3.66	3.05	-0.71	
T12	3.25	2.84	-0.9		3.69	3.19	-0.5	
AV:	**4.07**	**3.84**	**-0.28**	**-0.19**	**3.98**	**3.63**	**-0.38**	**-0.40**
SD:	**0.35**	**0.43**	**0.27**	**0.32**	**0.29**	**0.37**	**0.27**	**0.36**

Table 5-7 shows the results for the 12.65 kbps mixer. For MTT1 the parametric mixer performs better than the tandem mixer as explained in Sec. 5.2.1.

The MTT2 and MTT3 scenarios yield the worst speech quality. For the MTT2 case the LE and LQ degraded by 0.44 and 0.41 MOS points, respectively, for the MTT3 case they degraded by 0.38 and 0.4. In the speaker overlap scenario, MTT3, the speech degradation is more contained with 0.28 and 0.19.

The average LQ MOS score over the MTT2, MTT3 and MTT4 scenarios is 3.7. With a LQ MOS score of four meaning good and three meaning fair the 3.7 obtained by the parametric mixer is acceptable. If the MTT1 MOS score is added to this average, then the overall listening quality score jumps to 3.87 which is very close to toll quality.

Table 5-8 shows the results for the multi-rate parametric mixer. In the MTT2 scenario the results are very similar to the 12.65 kbps rate mixer while for the MTT3 and MTT4 there is a further listening quality degradation of 0.07 and 0.11. While the speech quality of the multi-rate mixer is slightly worse, the standard deviation of the MOS scores has decreased. This decrease is especially significant for the MTT2 case where it is 0.35. This highlights that if the overall performance of the mixer could be improved the multi rate mixer would be able to provide more consistent results.

Table 5-9 Speech Quality Results for Parametric Mixer at 12.65/18.25 kbps rate.

	Multi Talk Type 1				Multi Talk Type 2			
Test	12.65/18.25 Mixer		Degradation		12.65/18.25 Mixer		Degradation	
Case	LE	LQ	LE	LQ	LE	LQ	LE	LQ
T1	4.58	4.43	0.14	0.25	4.13	3.78	-0.28	-0.35
T2	4.55	4.41	0.16	0.28	3.85	3.28	-0.51	
T3	4.57	4.43	0.14	0.25	4	3.48	-0.39	-0.5
T4	4.6	4.44	0.2	0.31	4.08	3.82	-0.39	-0.38
T5	4.5	4.33	0.17	0.33	4.09	3.76	-0.22	-0.15
T6	4.57	4.47	0.18	0.36	4.08	3.57	-0.35	
T7	4.51	4.39	0.12	0.31	4.07	3.58	-0.34	
T8	4.39	4.19	0.18	0.38	3.96	3.53	-0.18	-0.22
T9	4.52	4.38	0.15	0.29	4.16	3.86	-0.24	-0.25
T10	4.59	4.46	0.16	0.32	4.34	4.02	-0.12	-0.14
T11	4.52	4.35	0.16	0.32	3.52	2.92	-0.87	
T12	4.48	4.27	0.23	0.37	3.8	3.54	-0.35	-0.2
AV:	**4.53**	**4.38**	**0.17**	**0.31**	**4.01**	**3.60**	**-0.35**	**-0.43**
SD:	**0.06**	**0.08**	**0.03**	**0.04**	**0.21**	**0.29**	**0.19**	**0.29**
	Multi Talk Type 3				**Multi Talk Type 4**			
T1	4.07	3.7	-0.33	-0.44	3.09	2.74	-1.32	
T2	4.31	4.16	-0.06	0.06	4.06	3.53	-0.33	
T3	4.19	3.68	-0.31		4.28	3.91	-0.11	-0.1
T4	3.81	3.57	-0.57		3.88	3.26	-0.52	
T5	4.25	3.86	-0.1	-0.12	4.07	3.69	-0.21	-0.19
T6	4.29	4.13	-0.14	-0.01	4.37	4.03	-0.06	-0.1
T7	4.23	4	-0.12	-0.03	4.11	3.6	-0.31	
T8	3.78	3.41	-0.33	-0.25	3.86	3.51	-0.31	-0.24
T9	4.29	4.01	-0.08	-0.07	3.97	3.57	-0.41	
T10	4.43	4.2	0.01	0.08	4.27	3.86	-0.16	-0.25
T11	4.04	3.59	-0.33		3.73	3.18	-0.64	
T12	3.36	2.92	-0.79		3.83	3.33	-0.36	
AV:	**4.09**	**3.77**	**-0.26**	**-0.26**	**3.96**	**3.52**	**-0.40**	**-0.51**
SD:	**0.30**	**0.37**	**0.23**	**0.29**	**0.34**	**0.36**	**0.33**	**0.38**

5.4 Parametric Mixer Complexity Analysis

One of the main objectives of the research in this thesis is to reduce the algorithmic complexity of the bridge in a centralized VoIP conferencing topology. In this section the complexity of the parametric mixer is analyzed and compared against the classic tandem mixing approach using Millions of Operations per Second (MOPS) measurements for such a comparison. It is mandatory to point out that this is only an approximate estimate as the MOPS figure would change depending on the hardware the parametric mixer is implemented on.

Table 5-9 gives the estimated number of operations for the mixing of two 20 ms frames. The number of operations is calculated by considering the theoretical worst case, when the path through the mixer giving the greater complexity is assumed. For example the second fixed codebook algorithm is considered as it is the one with the greater complexity.

Table 5-10 Parametric Mixer Complexity for 20 ms Frame

	Parametric Mixer Component	Operations
Voicing Algorithm		200
LPC Algorithm		
1	Decode ISFs	50
2	Convert ISFs to LPC	3000
3	Denominator coefficients of mixed synthesis filter (M = 33)	545
4	Numerator coefficients of mixed synthesis filter (N = 17)	51
5	Impulse Response of mixed synthesis filter	20200
6	Prony's Method to calculate mixed LPCs coefficients	111747
7	Convert LPCs to ISFs	2500
8	Quantize LPCs and Code Them	3000
Fixed Codebook Algorithm		
1	Decode fixed codebook pulses for S1 and S2	300
2	Add two codebooks	400
3	Code pulses at 18.25 kbps	800
Lags and Gains Algorithm		200
Others		600
	Total	**143593**

All the numbers reported in Table 5-9 consists of the summation of operations such as comparisons, additions, and multiplications. Over 90 % of the parametric mixer complexity is introduced by the calculations for the impulse response of the mixed synthesis filter and the calculations of the mixed LPCs using Prony's method. Given their weight in the overall mixer complexity it is important to understand how these numbers are obtained, for the remaining 10 % the number of operations is calculated by inspecting the software used to implement the algorithms.

The impulse response is calculated by using the *impz* function in Matlab. In this function a filter implemented with a direct form II realization and having the coefficients from steps 3 and 4 of the LPC algorithm (refer to Table 5-9), is used to filter an impulse vector, a vector of zeros except for the first element which is 1, of length *len*. The length of the impulse vector determines the accuracy of the impulse

response and it is directly proportional to the algorithmic complexity of this operation. For a direct form II implementation the number of operations are $N + M + 1$ multipliers and $M + N$ additions for each time delay [44], N and M represent the number of filter coefficients. In this case for an impulse vector of length $len = 200$ there will be $len * (N + M + 1) = 10200$ multipliers and $len * (M + N) = 10000$ additions.

The complexity in finding the mixed LPCs coefficients is also related to the length of the impulse response. Prony's method for time-domain infinite impulse response (IIR) filter design is used to find the coefficients of the synthesis filter given its impulse response [44]. Since the synthesis filter is an all pole filter Prony's method only needs to find the coefficients of the denominator, which for the G.722.2 codec corresponds to 17 coefficients. The *prony* Matlab function is used for this calculation. The LPC coefficients are found by solving the equation $a = inv(H)*h$. H is the non symmetric Toeplitz matrix having the normalized impulse response as the first column and an impulse vector of size 17, the number of LPC coefficients, as the first row, h is the impulse response vector. The QR decomposition is used to solve for a. The complexity of the QR decomposition given in [45] is $(2 * A^2 * (L - A/3))$, where $A = 17$ and $L = len-1$, with len being the length of the impulse response. If $len = 200$ the algorithmic complexity given by Prony's method is 111,747.

The tandem mixer approach performs full decoding of the two speech signals followed by a linear addition and full re-encoding. This signifies that the bridge of the central conferencing architecture would have to be equipped with two full G.722.2 decoders and one G.722.2 encoder for the two speaker mixing case. The

complexity of the components of such a bridge is given in Table 5-10 [41]. The complexity of the linear addition component is ignored as it is insignificant in relation to the complexity of the G.722.2 codec.

Table 5-11 Tandem Mixer Complexity

Tandem Mixer Component	Operations in MOPS
G.722.2 Decoder x 2	15.6
G.722.2 Coder	31.1
Total	**46.7**

In order to compare the parametric mixer to the tandem mixer the computational complexity presented in Table 5-9 is converted to MOPS. This is accomplished by multiplying the total figure by 50 (50 ms frames in one second) which gives a complexity of 50 (143593) = 7.18 MOPS a major saving compared to the 46.7 MOPS of the tandem mixer as shown in Table 5-12.

Table 5-12 Algorithmic Complexity Reduction for Parametric Mixing

Mixer Architecture	Operations in MOPS
Tandem Mixing	46.7
Parametric Mixing	7.18
Algorithmic Complexity Reduction	**39.52 MOPS or ~ 85%**

6 Conclusion and Future Work

A novel approach of mixing two speech signals in the bridge of a centralized voice conferencing architecture has been investigated in this thesis. The compression algorithm used in this investigation is the ITU-T Recommendation G.722.2 Adaptive Multi Rate Wideband encoder/decoder [25].

The new mixing approach introduced by this research consists in mixing the speech signals at the bridge at the parametric level. That is, by extracting the parameters used to encode the speech from the bit streams of the two signals and mixing these parameters instead of going through the full decoding, linear addition, and coding of the tandem mixing approach. This approach greatly reduces the complexity of the bridge, thus decreasing the codec processing delay, while maintaining acceptable speech quality.

The parametric mixing is achieved with the implementation of the following algorithms:

- Voicing algorithm;
- LPC mixer algorithm;
- Pitch mixer algorithm;
- Pitch gain algorithm;
- Fixed codebook mixing algorithm at 12.65 or 12.65/18.25;

Simulations have shown that the parametric mixing method can achieve satisfactory results. A DSLA was used to compare the speech quality obtained through the parametric mixer with the speech quality obtained through tandem mixing. Further to a black box testing of the parametric mixer, each algorithm was tested individually as well, in order to understand their impact on the overall speech quality. Over all simulations run, the average MOS recorded for the tandem mixing approach was 4.03. For the parametric mixer approach Table 6-1 shows the results for each algorithm and for the overall mixer when compared to the tandem mixing.

Table 6-1 MOS Results Summary

Algorithm	Speech Degradation in MOS
Voicing	0.05
LPC	0.11
Pitch Lags	0.07
Gains	0.22
Fixed CB 1	0.05
Fixed CB 2	0.16
Parametric Mixer	
Full 12.65	0.33
Full 12.65/18.25	0.4

In the case when the parametric mixer is operating at 12.65 kbps the overall MOS is 3.7 while when operating at 12.65/18.25 the overall MOS is 3.63. With a score of 3 meaning "fair" and a score of 4 meaning "good" in the MOS rating system, it can be concluded that the speech quality of the parametric mixer is acceptable.

The above analysis has helped identify possible areas of improvements in the parametric mixer that could be investigated in future work. For example the gains mixing algorithm has been identified as the one introducing the most distortion, further analysis is needed in this case to understand a better approach of mixing this parameter.

Also, while the second fixed codebook mixing algorithm has a worse average degradation compared to the first one, in some cases it has a better performance. The parametric mixer could be modified by introducing some intelligence to always choose the fixed codebook algorithm that gives the best speech quality. Again in this case further studies would have to be done to understand how the choice of algorithm could be carried out.

The main advantage of the parametric mixer compared to the tandem mixer is the reduction in algorithmic complexity. The complexity of the parametric mixer has been calculated at approximately 7.18 MOPS which compared to the 46.7 MOPS of the tandem mixer represents a complexity reduction of almost 85 %. This means faster mixing and less costly hardware to implement it on. Of course, as always in

engineering, this does come at a cost which in this case translates to higher distortion in the speech signal obtained through the parametric mixer when mixing is required.

Parametric Mixer Evaluation through the E-Model

The evaluation of the parametric mixer was performed using the PAMS approach. PAMS measures the audible distortions based on the perceptual domain representation of two signals, a reference signal and a degraded signal.

While this is a valid approach in comparing the performance of parametric mixing vs tandem mixing, it does not take into consideration other factors that affect the voice quality in a VoIP end to end transmission, such as packet loss, delay, and delay jitter.

Packet loss and delay jitter, under the same testing conditions, would be identical for both parametric and tandem mixers. The end to end delay, however, would not be equal in this scenario. The parametric mixer, thanks to its lower algorithmic delay, would have a lower end to end delay than the tandem mixer.

The E-model [46], unlike PAMS, is a computational model combining all the impairment parameters of end to end transmission into a total value. The E-model output, as PAMS, can be transformed into a MOS scale for prediction.

By using the E-model in evaluating the speech quality of the two mixers the difference in end to end delay can be taken into account.

In this case, the parametric mixer is expected to narrow the MOS gap between itself and the tandem mixer, or even outperform it. However, this would have to be proven by further analysis.

Parametric Mixing for Decentralized VoIP Conferencing

Although the focus of this research has been on simplifying the bridge in centralized VoIP conferencing architecture via parametric mixing, the same design can be applied to an end unit of a decentralized VoIP conferencing topology.

The implementation of decentralized conferencing depends on two key points; low cost end terminals and, in the case where the end terminal is a mobile unit, low power consumption.

Parametric mixing, thanks to its low algorithmic complexity, provides a solution that allows end terminals to mix multiple voice streams while addressing both of these points.

References

[1] "Packet Based Multimedia Communications Systems," *ITU-T Rec. H.323*, February 1998.

[2] H. Liu, P. Mouchtaris, "Voice over IP Signaling: H.323 and Beyond," *IEEE Communications Magazine*, October 2000, pp. 142-148.

[3] M. Handley, H. Schulzrinne, E. Schooler, J. Rosenberg, G. Camarillo, A. Johnston, R. Sparks, J. Peterson, "SIP: Session Initiation Protocol," *IETF RFC 3261*, June 2002.

[4] M. Handley, V. Jacobson, "SDP: Session description protocol," *IETF RFC 2327, April 1998.*

[5] Y. Zhang, "SIP-based VoIP Network and its Internetworking with the PSTN," *Electronics & Communication Engineering Journal*, Vol. 14, Issue 6, December 2002, pp. 273-282.

[6] F. Andreasen, B. Foster, "Media Gateway Control Protocol (MGCP)," *IETF RFC 3435*, January 2003

[7] T. Taylor, "Megaco/H.248: A New Standard for Media Gateway Control," *IEEE Communications Magazine*, Vol. 38, Issue 10, October 2000, pp. 124-132.

[8] H. Schulzrinne, S. Casner, R. Frederick, V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications," *IETF RFC 1889*, January 1996.

[9] "One-way Transmission Time," *ITU-T Rec. G.114*, May 2003

[10] J. R. Dellar Jr., J. H. L. Hansen, and J. G. Proakis, "Discrete-Time Processing of Speech Signals," New York, NY, *IEEE Press*, 2000.

[11] J. D. Tardelli et al., "The Benefits of Multi-Speaker Conferencing and the Design of Conference Bridge Control Algorithms," *Proc. IEEE Int'l Conf. Acoustics, Speech, Sig. Processing*, Minneapolis, MN, vol. 2, April 1993, pp. 435-438.

[12] P. J. Smith, P. Kabal, M. L. Blostein, "Tandem-Free VoIP Conferencing: A bridge to Next-Generation Networks," *IEEE Communications Magazine*, May 2003, pp. 136-145.

[13] J. Forgie, C. Feehrer, P. Weene, "Voice Conferencing Technology Final Report," *Tech. rep. DDC ADA074498 MIT Lincoln Lab.*, March 1979.

[14] T. G. Champion, "Multi-speaker Conferencing Over Narrowband Channels," *Proc. IEEE MILCOM*, Washington, DC, Nov. 1991, pp 1220-1223.

[15] P. J. Smith, P. Kabal, M. Blostein, R. Rabipour, "Tandem-Free Operation for VoIP Conference Bridges," IEEE International Conference on Communications, vol. 2, May 2003, pp 794-798.

[16] P. J. Smith, P. Kabal, R. Rabipour, "Speaker Selection for Tandem-Free Operation VoIP Conference Bridges," *Speech Coding, 2002, IEEE Workshop Proceedings*, October 2002, pp 120-122.

[17] ITU-T Rec. P.59, "Artificial Conversational Speech," Mar. 1993.

[18] ITU-T Rec. P.800, "Methods for Subjective Determination of Transmission Quality," Aug. 1996.

[19] PsyTechnics Group, British Telecom, "PAMS Usage Guidelines," Feb. 2000.

[20] Y. Ota et al., "Speech Coding Translation for IP and 3G Mobile Integrated Network," *IEEE International Conference on Communications*, vol. 1, May 2002, pp. 114-118.

[21] S. Lee, S. Seo, D. Jang, C. D. Yoo, "A Novel Transcoding Algorithm for AMR and EVRC Speech Codecs via Direct Parameter Transformation," *IEEE ICASSP*, vol. 2, April 2003, pp. 177-180.

[22] S. Tsai, J. Yang, "GSM to G.729 Speech Transcoder," *The 8th IEEE International Conference on Electronics*, vol. 1, Sept. 2001, pp. 485-488.

[23] K. R. Pankay, "A Novel Transcoding Scheme from EVRC to G.729AB," *Conference Record of the Thirty-Seventh Asilomar Conference on Signals, Systems and Computers*, vol. 1, Nov. 2003, pp. 533-536.

[24] H. Kang, H. Kim, R. V. Cox, "Improving the Transcoding Capability of Speech Coders," *IEEE Transactions on Multimedia*, vol. 5, no. 1, March 2003, pp. 24-33.

[25] ITU-T Rec. G.722.2, "Wideband Coding of Speech at Around 16 kbit/s Using Adaptive Multi-Rate Wideband (AMR-WB)," July 2003.

[26] IMTC Liaisons Documents, "Liaison to multiple SDOs requesting input for "Media Coding Summary Database" project," http://www.imtc.org/about/liaisons.asp, Geneva, Oct. 2002.

[27] R. Salami et al., "The Adaptive Multi-Rate Wideband Codec: History and Performance," *Speech Coding, 2002, IEEE Workshop Proceedings*, Oct. 2002, pp. 144-146.

[28] M. Perkins, K. Evans, D. Pascal, L. Thorpe, "Characterizing the Subjective Performance of the ITU-T 8 kb/s Speech Coding Algorithm – ITU-T G.729," *IEEE Communications Magazine*, Sept 1997, pp. 74-81.

[29] R. Salami, C. Laflamme, B. Bessette, J-P. Adoul, "Description of ITU-T Recommendation G.729 Annex A: Reduced Complexity 8 kb/s CS-ACELP Codec," *IEEE ICASSP-97*, vol. 2, Apr. 1997, pp. 775-778.

[30] TIMIT Acoustic-Phonetic Continuous Speech Corpus http://www.ldc.upenn.edu/Catalog/docs/TIMIT.html.

[31] Malden Electronics Ltd, http://www.malden.co.uk/

[32] ITU-T Rec. P.861, "Objective Quality Measurement of Telephone-band (300-3400 Hz) Speech Codecs," Feb. 1998.

[33] ITU-T Rec. P.862, "Perceptual Evaluation of Speech Quality (PESQ), An Objective Method for End-to-End Speech Quality Assessment of Narrowband Telephone Networks and Speech Codecs," Feb. 2001.

[34] Z. Qian, "Audio Mixers for Centralized VoIP Conferencing," Master of Applied Science Thesis, *Carleton University*, May 2003, pp. 58-64.

[35] J. Davidson, J. Peters, "Voice over IP Fundamentals," Indianapolis, IN, *Cisco Press*, 2000.

[36] T. N. Yensen, R. A. Goubran, I. Lambadaris, "Synthetic Stereo Acoustic Echo Cancellation Structure for Multiple Participant VoIP Conferences," *IEEE*

Transactions on Speech and Audio Conferencing, vol. 9, issue 2, February 2001, pp. 168-174.

[37] J. D. Gordy, R. A. Goubran, "A Perceptual Performance Measure for Adaptive Echo Cancellers in Packet-Based Telephony," *IEEE International Conference on Multimedia and Expo*, July 2005, pp. 157 – 160.

[38] L. Zheng, L. Zhang, D. Xu, "Characteristics of Network Delay and Delay Jitter and its Effect on Voice over IP (VoIP)," *IEEE International Conference on Communications*, vol. 1, 2001, pp. 122-126.

[39] F. P. Zhang, O. W. W. Yang, B. Cheng, "Performance Evaluation of Jitter Management Algorithms," *Canadian Conference on Electrical and Computer Engineering*, vol. 2, May 2001, pp. 1011-1016.

[40] C. Perkins, O. Hodson, V. Hardman, "A Survey of Packet Loss Recovery Techniques for Streaming Audio," *IEEE Network*, vol. 12, no. 5, Oct. 1998, pp. 40-48.

[41] "Wideband Speech Coding Standards and Applications," White Paper, *VoiceAge Corporation*, Montreal, Canada, Sept. 2001.

[42] C. Laflamme, J-P. Adoul, R. Salami, S. Morissette, P. Mabilleau, "16 KBPS Wideband Speech Coding Techniques Based on Algebraic CELP," *ICASSP-91*, vol. 1, April 1991, pp. 13-16.

[43] D. Nahumi, "Conferencing Arrangement for Compressed Information Signals," *U.S. Patent 5390177*, Feb. 1995.

[44] J. G. Proakis, D. G. Manolakis, " Digital Signal Processing," *Prentice-Hall*, New Jersey, NJ, 3rd edition, 1996.

[45] G. H. Golub, C. F. Van Loan, "Matrix Computations," *Johns Hopkins University Press*, Baltimore, MD, 3rd edition, 1996.

[46] ITU-T G.107, "The E-model, a computational model for use in transmission planning," 2000.

Appendix A

Corrections Submitted to ITU-T for Recommendation G.722.2 (07/2003)

In 5.6, Target signal computation, page 23, equation (36)
It reads:

$$r(n) = s(n) = \sum_{i=1}^{16} \hat{a}_i s(n-i), \quad n = 0,\ldots,63$$

It should read:

$$r(n) = s(n) - \sum_{i=1}^{16} \hat{a}_i s(n-i), \quad n = 0,\ldots,63$$

In 5.7, Adaptive codebook, page 24, line 29
It reads:

"Thus, for 8.85-, 12.65-, 14.25-, 15.85-, 18.25-, 19.85-, 23.05-, or 23.85-Kbit/s, there are two possibilities to generate the adaptive codebook *v(n)*..."

For the rate 8.85 as well as the rate 6.60 there is only one way of generating the adaptive codebook. Therefore 8.85 should not be included in the sentence above. This is also reflected by the table on page 50, there is no LTP-filtering-flag, and again by the table on page 13, no bits transmitted for LTP-filtering in 8.85 rate. I also confirmed this with the ITU-T c-code implementation for G.722.2.

In 5.8.3, Codebook search, page 33, last equation on the page

It reads:

$$R = \sum_{i=0}^{N_p-1} d'(i)$$

It should read:

$$R = \sum_{i=0}^{N_p-1} d'(m_i)$$

In 6.1, Decoding of speech synthesis, page 38, step 1

It reads:

"... The adaptive codebook vector *v(n)* is found by interpolating the past excitation *u(n)* (at the pitch delay) using the FIR filter described **in 5.6**. ..."

*It should be in **5.7**.*

In 6.1, Decoding of speech synthesis, page 39, step 6, equation (65)

It reads:

$$\hat{g}_c = \theta g_0 + (1 + \theta)\,\hat{g}_c$$

It should read:

$$\hat{g}_c = S_m g_0 + (1 - \theta)\,\hat{g}_c$$

I validated the above equation after analyzing the G.722.2 C-code implementation.

In 6.1, Decoding of speech synthesis, page 40, step 7

The line after equation (66) reads:

"... where $c_{pe} = 0.125(1 - r_v)$, ..."

It should read:

"... where $c_{pe} = 0.125(1 + r_v)$, ..."

Again this was confirmed after analyzing the C-code implementation.

In 5.7, Adaptive codebook, page 24, equation (39)
It reads:

$$g_p = \frac{\sum_{n=0}^{63} x(n)y(n)}{\sqrt{\sum_{0}^{63} y(n)y(n)}}, \qquad \text{bounded by} \quad 0 \le g_p \le 1.2,$$

It should read:

$$g_p = \frac{\sum_{n=0}^{63} x(n)y(n)}{\sum_{0}^{63} y(n)y(n)}, \qquad \text{bounded by} \quad 0 \le g_p \le 1.2,$$