PERFORMANCE EVALUATION OF THE FUTURE

NARROWBAND DIGITAL TERMINAL AND

MELP OVER CONCATENATED

INTERNET AND CDMA

CELLULAR DATA

NETWORKS

By

EDWARD JAMES DANIEL

Associate of Applied Science
Spartan School of Aeronautics
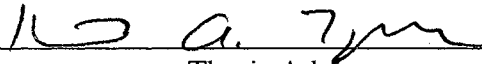Tulsa, Oklahoma
1992

Bachelor of Science
Oklahoma State University
Stillwater, Oklahoma
1997

Master of Science
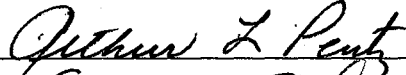Oklahoma State University
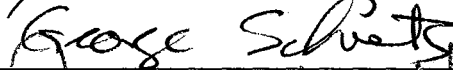Stillwater, Oklahoma
2000

PERFORMANCE EVALUATION OF THE FUTURE

NARROWBAND DIGITAL TERMINAL AND

MELP OVER CONCATENATED

INTERNET AND CDMA
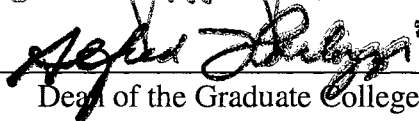
CELLULAR DATA

NETWORKS

Thesis Approved:

_____
Thesis Advisor

_____

_____

_____

_____
Dean of the Graduate College

ii

ACKNOWLEDGEMENTS

This dissertation would not have been possible without the support and guidance of many people. I would like to take this opportunity to acknowledge and thank some of those people.

First, I would like to thank my Lord and Savior, Jesus Christ, who is the author and finisher of my faith. Thank you for blessing the work of my hands, and giving me all that I needed to do this and every good work. It was your commitment to me that enabled me to finish this project. I pray that this dissertation will be a testament to your love and faithfulness.

Second, I want to thank my advisor, Dr. Keith A. Teague, for his technical expertise, wisdom and support that was instrumental in the completion of this dissertation. I am forever grateful for your advice and friendship throughout the years "on and off the court". Additionally, I wish to thank the other members of my advisory committee, Dr. Rao Yarlagadda, Dr. George Scheets, and Dr. Arthur Pentz, as well as the School of Electrical and Computer Engineering for their support throughout the years. I also want to thank the Department of Defense for funding this research, especially Mary Ruhl.

Next, I would especially like to thank my wife, Cherdena Daniel. You have truly been my inspiration. Thank you for your patience, support, sacrifice, and encouragement throughout the years, especially on this project. I could not have completed this

dissertation without you by my side. Thank you. To my precious daughter Cheryl Gabrielle, thanks for your patience while I completed this work. I love you.

Finally, I want to thank my mother Shirley Wallace, my four sisters, Kim, Lynn, Darlene, and Donna, and my father-in-law George Fowler, for the love, support, prayers, and encouragement they have given me throughout the years. I also would like to thank my brother-in-law Randy for his encouragement and particularly for introducing me to the field of engineering. My other family members and friends has also prayed for and encouraged me, and I appreciate them. None of this would have ever been possible without their support.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

NOMENCLATURE

3G	Third Generation
ACK	Acknowledgement
ADPCM	Adaptive Differential Pulse Code Modulation
ARQ	Automatic Repeat Request
ATM	Asynchronous Transfer Mode
BER	Bit Error Rate
BS	Base Station
CDMA	Code Division Multiple Access
CELP	Code Excited Linear Prediction
CRC	Cyclic Redundancy Check
DDVPC	Defense Digital Voice Processor Consortium
DTX	Discontinuous Voice Transmission
EMBE	Enhanced Multi-Band Excitation
EOM	End of Message
F (R)	Forward and Reverse
FEC	Forward Error Correction
FER	Frame Error Rates
FIFO	First- In First- Out
FNBDT	Future NarrowBand Digital Terminal
F_s	Sampling Frequency
GPRS	General Packet Radio Service
GSM	Global System for Mobile Communication
GUI	Graphical User Interface
IMBE	Improved Multi-Band Excitation

IMBE	Improved Multi-Band Excitation
IP	Internet Protocol
ISDN	Integrated Services Digital Network
ITU-T	International Telecommunications Union
IWF	Inter-Working Functions
LAN	Local Area Network
LMR	Land Mobile Radios
LPC	Linear Predictive Coding
LSB	Least Significant Bit
LSF	Line Spectral Frequencies
LTU	Logical Transmission Units
MBE	Multi-Band Excitation
MELP	Mixed Excitation Liner Prediction
MH	Mobile Host
MIL-STD-3005	Military Standard 3005 (MELP)
MoIP	MELP-over-IP
MOS	Mean Opinion Score
MSB	Most Significant Bit
NetPerf	Network Performance Application
NTP	Network Time Protocol
OSI	Open System Interconnect
PC	Personal Computer
PCM	Pulse Code Modulation
pdf	probability density function
PESQ	Perceptual Evaluation of Speech Quality
PO	Percentage Outlier (frames)
PSTN	Public Switched Telephone Network
QoS	Quality of Service
RLP	Radio Link Protocol

RMS	Root Mean Square
RTCP	Real-Time Control Protocol
SDfw	Spectral Distortion frequency weighted
SMF	Sync Management Frame
SNR	Signal to Noise Ratio
SOFM	Self-Organizing Feature Map
SOM	Self-Organizing Map
SOM	Start of Message
STE	Secure Terminal Equipment
STU III	Secure Telephone Unit III
TCP	Telnet Control Protocol
TE	Terminal Equipment
UDP	User Datagram Protocol
VAD	Voice Activity Detection
VAF	Voice Activity Factor
VoIP	Voice-over-IP
WAN	Wide Area Network
XOR	Exclusive-or

CHAPTER I

INTRODUCTION

The government and military have plans for secure interoperable communication using both commercial and proprietary networks and terminal equipment. This vision is known as the Future NarrowBand Digital Terminal (FNBDT). The FNBDT initiative realizes the concept of "anyplace, anytime" communication. The FNBDT system will provide seamless interoperable communication, enabling secure communication regardless of the point of presence of the user in the communications network [1]. The user will experience transparent end-to-end communication, with respect to actual underlying communications network(s) providing the connection.

The FNBDT system will include interoperability between networks (internet, cellular, Public Switched Telephone Network (PSTN), tactical, etc.) and terminal equipment (PC's, mobile radios, phones, and various handheld devices). It will also encompass protocols, media types (data, voice, fax, video, etc), as well as security (encryption). The FNBDT system will replace the existing Secure Telephone Unit (STU III) and other legacy systems to provide secure communications. It will provide secure communication for both narrowband mobile users and broadband wireline (land) users, with the eventual goal of providing an interoperable system that will enable secure communication for mobile and wireline users over mixed networks.

A key component in achieving interoperable voice communication for military and government agencies (including FNBDT) was the adoption of the 2.4k bps Mixed Excitation Liner Prediction (MELP) voice coding algorithm. MELP is the current Military Standard (MIL-STD-3005) for narrowband secure voice coding products and

1

systems [3] [4]. The incorporation of MELP as the voice coding standard into FNBDT

represents the first phase of the FNBDT multimedia communications initiative. Figure

1.1 shows the overall FNBDT concept.
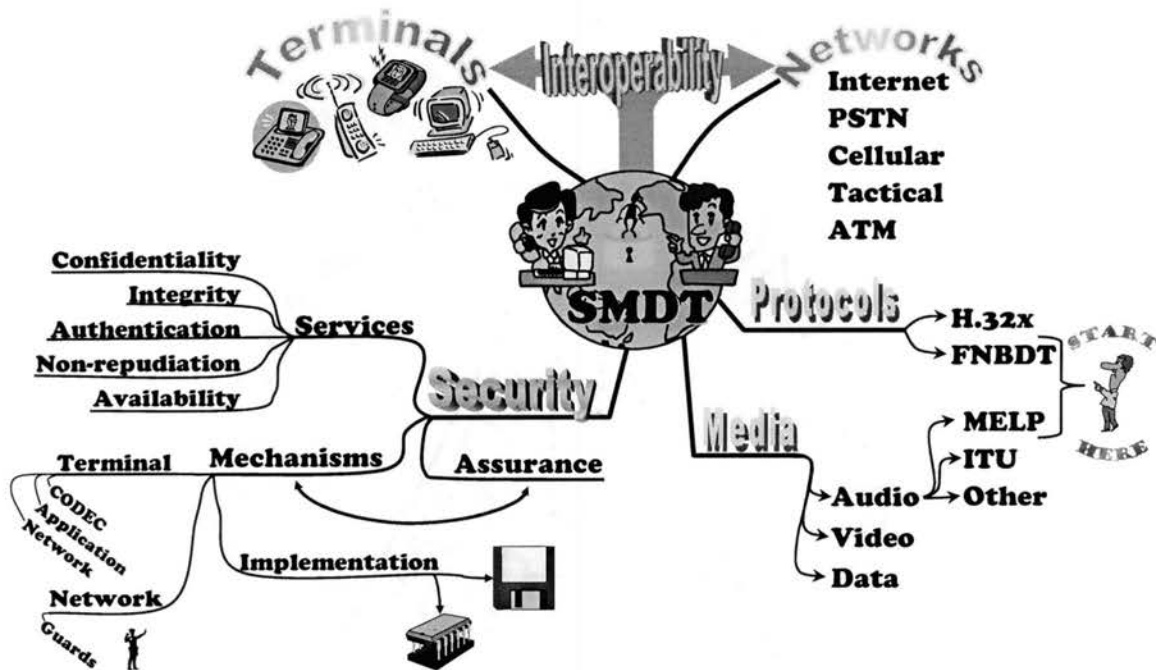
Figure 1.1 FNBDT Concept

Purpose

FNBDT is currently in the initial stages of development, with modifications being

made as development proceeds. There are several entities, government, private industry,

and academia, working on not only the signaling plan but also the terminal equipment

that will make up the system. However, a complete performance study has not yet been

done and is vital to the successful deployment of this system. There are several issues that

must be addressed for the deployment of such a system with its expected functionality

and interoperability. FNBDT will be deployed over not only a single network (i.e. LAN

to LAN or cellular data to cellular data), but also over mixed networks (i.e. LAN to cellular data). The characteristics of networks vary widely, each having its own non-ideal behavior. The concatenation of networks results in more complicated error characteristics. The resulting effect of network behavior on the operation of FNBDT and MELP becomes more difficult to predict. Therefore, the need for modeling and prediction of network behavior becomes important when FNBDT is deployed, especially across high-error rate channels such as packet wireless networks.

Little work has been done in characterizing how FNBDT / MELP will perform in IP packet environments, while much work has been published regarding modeling of network behavior. Potential questions may include the following. How can we expect FNBDT to perform under harsh wireless environments? Under unpredictable packet environments such as the Internet? Under concatenated Internet and wireless environments? How will FNBDT react to a single lost voice frame? Multiple lost voice frames? How will FNBDT react to a lost packet that contains n number of FNBDT frames? Will certain FNBDT frames have a bigger impact on communication quality or performance if they are lost or corrupted? If so, which ones? The answer to these and several others questions are vital to the successful deployment of FNBDT and represent the thrust of this research.

Prediction of FNBDT and MELP system performance over various types of networks is possible using published network performance data, statistical models and collected empirical network statistical data. Knowledge of how FNBDT and MELP react to particular non-ideal network behavior will also contribute to the prediction of system performance.

Research Goals

As part of an ongoing funded research project, the Speech and Audio Communications Laboratory of the Oklahoma State University School of Electrical and Computer Engineering developed a voice-over-IP (VoIP) communications system, using the MELP audio codec that simulated one aspect of the FNBDT protocol [5]. The development of this system explored the issues associated with providing real-time voice communication over a packet switched network, namely the Internet [6]. This research addresses a continuation of that project to include the deployment of FNBDT (and MELP) over mixed networks, Internet and CDMA cellular data. Specifically, this research considers how the characteristics of different packet networks will affect the operation of FNBDT and MELP, particularly how they affect the Quality of Service (QoS) and voice speech quality, of the end-to-end communication.

In order to a provide performance evaluation of the FNBDT communication system operating over mixed Internet and CDMA Cellular data networks, this research has the following goals and objectives: 1) identify network characteristics and limitations that will degrade FNBDT system performance, 2) project network limitations on the FNBDT system (operation), 3) evaluate the performance of the FNBDT system, in terms of overall system quality or QoS, 4) present and assess FNBDT system quality enhancements, 5) evaluate FNBDT performance with improvements, and finally 6) make recommendations for a robust FNBDT system for mixed packet network environments.

Outline

In this dissertation FNBDT, MELP, and network characteristics are studied, and network models are chosen and developed based on the findings. The network and FNBDT models are then used to simulate FNBDT communication over mixed packet networks. FNBDT communication simulations are performed over a wide range of network conditions. The performance of the raw FNBDT system is assessed and quality enhancements are recommended and evaluated. Finally, results of the FNBDT system performance evaluation are presented.

This dissertation is organized in the following manor. Chapter 2 provides a background on low rate voice coding, the MELP voice coder, and a thorough discussion of FNBDT. To understand the application of sending FNBDT/MELP data the reader is given a brief introduction of transmitting voice over packet switched networks, the associated QoS challenges, and queuing theory, chapter 3. Chapter 4 details properties, statistics, and characteristics reported in the research literature for Internet and CDMA cellular data networks. Chapter 5 shows results of empirical "real-life" collected network statistics to verify characteristics reported in the research literature outlined in chapter 4 and to gain additional insight of network behavior for modeling. The network properties found in the literature, collected network statistics, and properties of FNBDT secure communication are used to identify literature based network models found that characterize network behavior, and aid in the development of new network and FNBDT models. Chapter 6 presents network packet loss models for Internet and CDMA cellular data and introduces new models for FNBDT and packet network inter-arrival delay jitter. The models presented in Chapter 6 are used to simulate network communication with an

FNBDT system. However, in order to do a performance evaluation, quality measures are needed. Therefore, chapter 7 introduces quality measures for evaluation of speech quality resulting from FNBDT / MELP communication simulations. Chapter 7 also presents a self-organizing map neural network that enables the overall system quality evaluation, providing a means of determining the impact of network characteristics on FNBDT system output quality. With an understanding of how networks behave and how FNBDT / MELP operate (what network characteristics have potential for degradation in FNBDT / MELP operation), quality enhancements are recommended in chapter 8 to improve overall system communications. Chapter 9 provides a simulated FNBDT performance evaluation using model simulations that explore varying network characteristics (model parameter space). Performance enhancements are implemented and simulated to demonstrate system improvement. From the results of Chapter 9, findings, recommendations, future work and other applications of this research is presented in Chapter 10. Finally, the contribution of this research is summarized in Chapter 11, along with conclusions.

CHAPTER II

BACKGROUND

Low Rate Voice Coding

Speech coding entails the conversion of analog speech into an appropriate digital representation that preserves important characteristics of the speech signal, such as intelligibility and naturalness, while producing a representation that is suitable for transmission or storage. The primary goal of speech coding is usually to achieve a level of compression for use in low bandwidth (or bandwidth constrained) communication and storage applications. This is the goal for applications that transmit real-time voice over packet networks.

Speech coders are typically classified into two broad categories based on their method of operation: waveform coders and vocoders. Waveform coders such as Pulse Code Modulation (PCM) [7] and Adaptive Differential Pulse Code Modulation (ADPCM) [7] attempt to preserve the original shape of the speech waveform by encoding the signal on a sample-by-sample basis, using temporal and/or spectral techniques [8]. Waveform coders operate by attempting to remove predictable (correlated) portions of the signal or its spectrum. Signal to noise ratio (SNR), based on the closeness of the coded and the original speech signals, is the most often applied measure of performance.

Vocoders such as Mixed Excitation Linear Prediction (MELP) ([4] [9] [10] [11]), Multi-Band Excitation (MBE) ([12] [13]), Code Excited Linear Prediction (CELP) [14], Improved Multi-Band Excitation (IMBE) [15], and Enhanced Multi-Band Excitation (EMBE) 2.4k bps [16], estimate a set of parameters based on a vocal tract model. Unlike

7

waveform coders, vocoders do not attempt to preserve the speech waveform, but attempt to preserve the perceptual qualities of the speech signal through modeling and parameter estimation techniques. Once the parameters are estimated, the original speech waveform (signal) is discarded and a synthetic speech signal is generated using the estimated parameters. The fidelity of a vocoder is based on observed naturalness and intelligibility – both subjective criteria that are in stark contrast to SNR.

Both vocoders and waveform coders have limitations. The performance of waveform coders drops off dramatically at rates below about 16 kbps and is thus not appropriate for applications where significant compression is desired. Vocoders are better suited to lower rates (typically 16 kbps and below), offering opportunities for substantial compression with a corresponding tradeoff in intelligibility and naturalness.

Basic Vocoder Operation

A speech coder, or vocoder, consists of two operational parts: an analyzer (encoder) and a synthesizer (decoder), Figure 2.1. The analyzer is responsible for determining, on a short time basis, certain characteristics of the input speech signal, given a specific speech model.

In most cases pitch (fundamental frequency), a single voicing decision, gain, and vocal tract spectrum representation comprise the desired parameters estimated from the input speech signal [16]. The estimation techniques as well as the actual parameters vary from vocoder to vocoder.
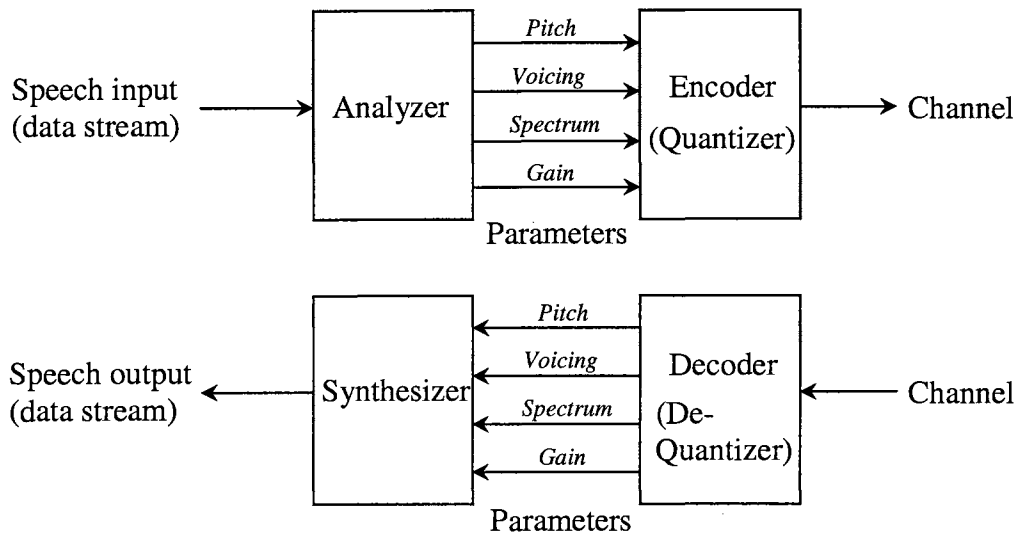
Figure 2.1 Typical Vocoder.

Once estimated, the parameter vectors are properly coded and sent to the decoder

for reconstruction. In the synthesizer, the analysis model is applied in reverse. The goal

of the synthesizer is to produce the best-sounding speech signal without regard for how

the sample-to-sample values correspond to the actual input waveform [12].

Mixed Excitation Linear Prediction (MELP)

MELP was specifically developed by Texas Instruments under the Defense

Digital Voice Processor Consortium (DDVPC) to become a new standard speech coder at

2,400 bps. MELP is currently the new military standard (MIL-STD-3005) and is

proposed to become the federal standard for 2,400 bps high quality speech, replacing

Federal Standard FS-1015 (LPC-10) [4], which, by modern standards, produces very low

quality speech. MELP at 2,400 bps performs as well as or better than Federal Standard

FS-1016 (CELP) at 4,800 bps [9], which is the current benchmark system for low rate

9

speech, making MELP an excellent candidate for most low rate secure voice applications in the government and military.

MELP is a linear predictive coding (LPC) vocoder; specifically, MELP is a mixed excitation LPC vocoder. MELP differs from LPC ([17]) in that it has some added features that improve its performance (allowing the synthesizer to better mimic characteristics of natural speech). These enhancements include: mixed voiced and noisy excitation, periodic and aperiodic pulses, adaptive spectral enhancements, pulse dispersion, and Fourier magnitude modeling [9]. Figure 2.2 shows the MELP synthesizer.
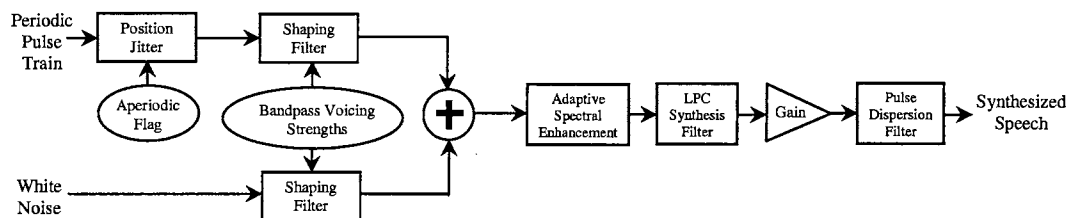
Figure 2.2 MELP Synthesizer

MELP produces the following parameters to represent speech frames: Line Spectral Frequencies (LSF), Fourier magnitudes, gain (2 per frame), pitch, overall voicing, bandpass voicing, aperiodic flag, error protection, and sync bit [4]. MELP requires an input of 180 samples (16-bits per sample) of raw speech at 8,000 samples per second, for an effective frame length of 22.5ms. To achieve 2,400 bps the parameters mentioned above are packed into 54 bits per frame giving approximately a 53:1 compression ratio. The bit allocation for MELP's output parameters is illustrated in Table 2.1 [4].

These parameter bits are scrambled (reordered) to accommodate limited channel

bit errors. The parameters are scrambled in a way that enables recovery of the important

parameters in case of localized bit errors in the bit stream. The bit ordering matches the

LPC-10 Federal Standard [4]. MELP also includes forward error correction (FEC),

embedded within unvoiced frames. FEC bits are sent in place of parameters not sent in

unvoiced frames, which include Fourier magnitudes, bandpass voicing, and an aperiodic

flag.

Table 2.1 MELP Bit Allocation

PARAMETERS	VOICED	UNVOICED
LSF parameters	25	25
Fourier magnitudes	8	-
Gain (2 per frame)	8	8
Pitch, overall voicing	7	7
Bandpass voicing	4	-
Aperiodic Flag	1	-
Error protection	-	13
Sync bit	1	1
Total Bits / 22.5 ms frame	54	54

An enhancement to the MELP 2.4k speech coding algorithm has been developed

to accommodate the need for support of transmission channels that do not support robust

speech coding at bit rates of 2.4k bps and below [2]. The enhanced algorithm is based on

the MELP 2.4k (MIL-STD-3005) algorithm, and is known as MELPe, the new

1.2kbps/2.4kbps MELP codec algorithm. In addition to the lower rate of 1.2 kbps, other

enhancements to the original MELP algorithm include a noise pre-processor that provides

high performance in harsh acoustic noise environments, an improved pitch enhancement algorithm in the transmitter, and an adaptive post filter in the receiver ([3][18]). For more information on the new MELPe coder refer to the above references as well as ([19] [20] [21] [22] [23]).

Future NarrowBand Digital Terminal (FNBDT)

FNBDT and its corresponding signaling plan [24] (referred to jointly as FNBDT in the following sections), currently under development by the government, provide point-to-point secure communications between end-to-end terminals potentially operating over a variety of narrowband, wideband, and protected networks. FNBDT is a digital solution that incorporates emerging commercially available digital technologies while providing a bridge to the existing analog infrastructure [25]. FNBDT allows communication continuity between users without regard to the communication platform. Further, although FNBDT suggests a "digital terminal", FNBDT encompasses the entire end-to-end communication system. It involves the end system hardware, the terminal equipment (TE), the protocol suite, and the network (unspecified). FNBDT includes the interoperability of the end-to-end hardware such as secure terminal equipment (STE), multimedia PC's, land mobile radios (LMR), cellular phones, etc. The protocol suite involves the data format (data, voice, fax, video, image, etc.), the signaling and control, and the communications modes.

FNBDT provides interoperability through its software configurable hardware, common mode negotiation between peer terminals, and its "network independent" protocol (signaling plan). FNBDT has a common set of functions that are either fixed or defined for all terminals at any given time [1]. These functions are used by FNBDT for

12

mode negotiation between terminals to establish the communication modes for the connection. For network interoperability, FNBDT only defines the higher-layer end-to-end protocols leaving the lower-layer network protocol open to accommodate different networks. The FNBDT signaling and control functions are intended to be independent of the network medium. FNBDT depends on a network connection being available, but is not dependent on the native protocol or formatting being used. Figure 2.3 illustrates the structure of a typical voice over packet network application using FNBDT. Specifically, the application data into and out of FNBDT is assumed to be a simple bit stream. Once a native network connection is established, the connection is turned over to FNBDT for call completion, security negotiation, and data communication. With regard to MELP, two aspects of the FNBDT protocol suite are important: the voice communication modes and the high-layer protocol architecture.
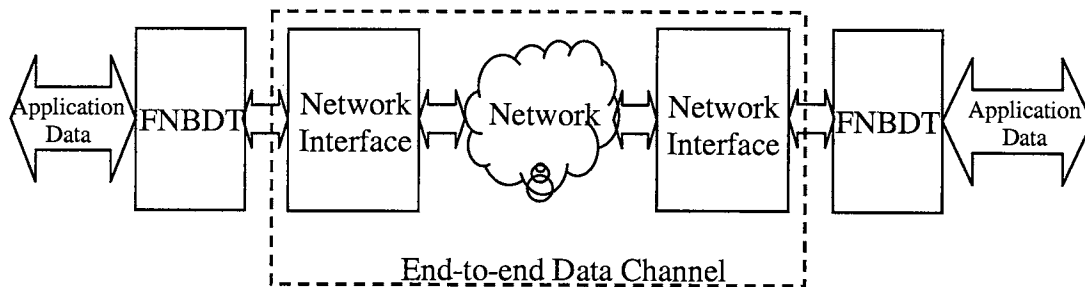
Figure 2.3 Relationship between FNBDT and the Network.

FNBDT Signaling Plan

FNBDT is being developed by a number of entities, including government, military, private industry, and academia. Issues currently being addressed range from network compatibilities to future world wide standards compliance. FNBDT will

13

ultimately support a variety of data types, however secure voice communication is the first phase of FNBDT development. To provide interoperability over narrowband, wideband, and protected networks, FNBDT specifies the new military standard MELP 2.4 kbps speech coder (MIL-STD-3005) for low rate voice communications [4] [75].

FNBDT provides a secure overlay to the existing commercial and military communications infrastructure to provide seamless secure communication between *like* and *unlike* networks that make up the underlying end-to-end connection. Specified within the FNBDT signaling plan is a minimum set of requirements to enable compatibility between FNBDT enabled devices.

The open signaling plan defines signaling requirements for end-to-end terminals. This signaling includes the call setup and negotiation required to establish an FNBDT connection as well as the FNBDT operational modes for secure voice and data communication. FNBDT signaling is defined for terminals that operate over digital narrowband channels such as commercial digital cellular and mobile satellite, and military and tactical channels. However, the signaling is also compatible with wideband channels such as the Internet and ATM. The signaling plan is intended to be flexible to accommodate future networks and additional data types (i.e., video).

FNBDT defines, independent of the communication network, how point-to-point FNBDT enabled terminals interoperate, including:

- The secure control signaling required to initiate, maintain, and terminate a secure mode of operation.

- The exchange of proprietary secure voice or data traffic (key exchanges, certificates, etc.) between two terminals.

- The transmission of secure voice traffic using the military standard MELP vocoder at 2,400 bps.

- The transmission of secure data traffic.

FNBDT Interoperability

The "network independent" FNBDT protocol (signaling plan) attains network interoperability between dissimilar networks by not defining lower layer network protocols. FNBDT relies on the application developer to specify the lower network layers protocols to provide the end-to-end delivery of data. This provides the greatest flexibility for inclusion of a variety of different network, and even concatenations of dissimilar networks. This does imply that for mixed network scenarios, FNBDT and the underlying application must rely on existing Interworking Functions (IWF) to translate (or connect, interface) data between the given networks. Finally, the specification of the MELP vocoder for secure and unsecure voice at 2400 bps insures interoperability amongst wide band and narrowband digital networks, including legacy networks.

FNBDT Protocol Architecture

FNBDT uses Open System Interconnect (OSI) layer 5 (session layer) and higher (see [26] for OSI 7 layer model), Figure 2.4, for its end-to-end protocol definition. Leaving the lower network layers "open-ended" allows developers the ability to define the communication platform to deliver the end-to-end data. FNBDT provides a service to the higher layers (layers 6-7) for the delivery of messages. As a layer 5 protocol, FNBDT calls on the lower network layers (layers 1-4) for the transmission of messages. FNBDT provides layer 4 with a data stream for transfer across the network, relying on the lower

layers to transparently deliver the message. By not defining these lower layers, the
network of choice becomes transparent to FNBDT. The FNBDT protocol sees the lower
network layers as a "bit pipe". The input and output, with regard to layer 4 and below,
appears as an input/output (I/O) data stream in relation to the FNBDT protocol.
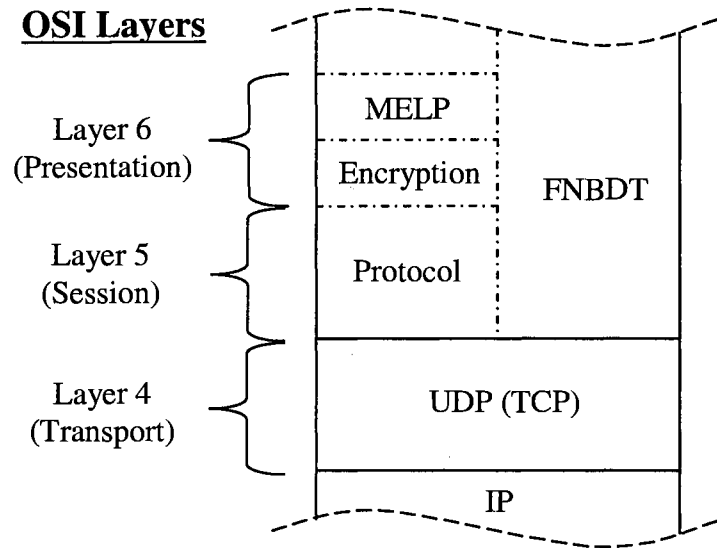
Figure 2.4. FNBDT Layered Architecture

In the high-layer protocol, FNBDT defines the proposed federal standard MELP
2.4 kbps voice coder for the compression and decompression of voice. MELP resides in
layer 6 (presentation layer) where FNBDT encryption is also performed for the secure
transfer of voice data.

FNBDT Call Setup Signaling and Control

An application following the FNBDT signaling plan must first perform its own
call setup by establishing a native end-to-end data connection using lower-layer network

communication protocols (TCP, UDP, RTP, etc.). Figure 2.5 gives an example of how a

typical voice-over-IP application might establish a native connection prior to invoking

FNBDT. Once the native end-to-end data channel is established, control (of the channel)

is passed to FNBDT. FNBDT then proceeds to perform its higher-layer signaling to

establish a point-to-point FNBDT session, Figure 2.6. The communication network

appears as a simple data pipe to FNBDT.

To insure the successful exchange of signaling and control data, FNBDT uses

reliable transport messaging. Reliable transport utilizes several error control mechanisms,

which include framing, Forward Error Correction (FEC), Cyclic Redundancy Check

(CRC), retransmits, and a combination of positive acknowledgements and selective

rejects [24].
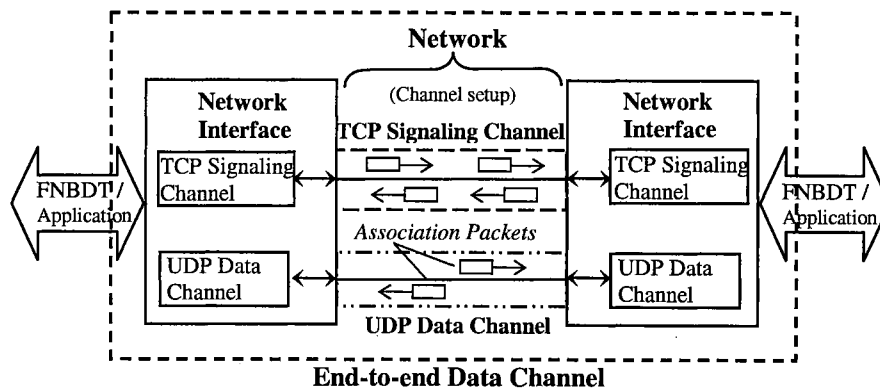
Figure 2.5 Native End-to-End Packet Network Connections.
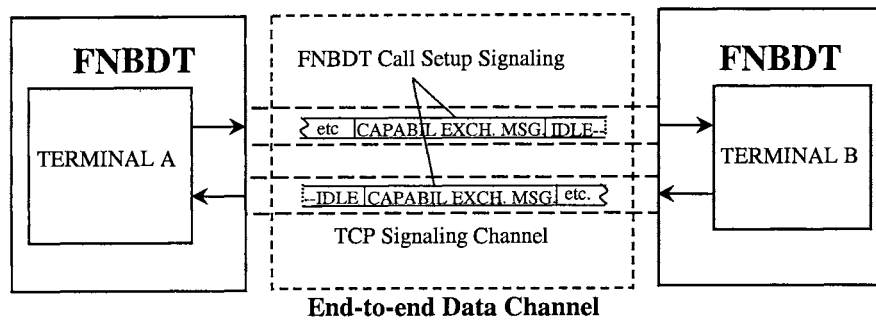
Figure 2.6 FNBDT Signaling and Connection.

Prior to secure (or clear) voice or data transfer, both FNBDT terminals initiate

point-to-point call setup signaling that includes negotiation, security, and control. A

Capabilities Exchange message is followed by Parameter/Certificate messages, F (R)

messages (Forward and Reverse messages), and Cryptosync messages [24]. The

Capabilities Exchange messages are used to negotiate the Operational Mode (and select a

Keyset) common to both terminals. If a *clear* Operational Mode is agreed upon by both

terminals, clear application signaling will begin. If a *secure* Operational Mode is agreed

upon by the end terminals, call setup proceeds by exchanging Parameter/Certificates and

F (R) messages. Upon receipt of these messages the terminals will use the Certificate and

F (R) for the selected Keyset to generate a common traffic key [24]. The terminals will

then encode and encrypt a common set of data, and exchange the encrypted data by

placing it into the Cryptographic Exchange messages. During the exchange of the above

signaling messages, reports are sent between end terminals. The reports indicate

successfully and unsuccessfully received messages. Once the signaling is complete, the

terminals proceed with data transfer according to the selected Operational Mode.

FNBDT Operational Modes

Figure 2.7 illustrates the transfer of secure voice data frames with FNBDT over a packet network.
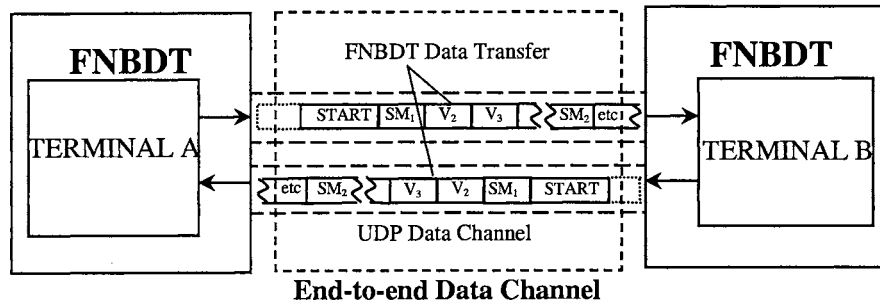
Figure 2.7 FNBDT Data Transfer.

FNBDT defines a variety of modes for transferring both secure and clear data. The modes will eventually include multimedia sources. For now, emphasis is on secure and clear voice using the MELP 2.4 kbps voice coder. The various communication modes describe the data stream – in this case the packet stream. Each mode utilizes a superframe structure that involves the placement of a sync management frame followed by a number of data or MELP voice frames. In secure mode the sync management frame contains information that allows cryptographic synchronization for the corresponding superframe data. Four of the possible Operational Modes include, *Clear* MELP, secure MELP *Blank and Burst*, secure MELP *Burst without Blank*, and secure *Asynchronous Data*. FNBDT relies on the underlying connection to reliably deliver its data.

Clear 2,400 bps MELP is a raw voice mode with a superframe structure that includes a 54 bit sync management frame followed by 23 MELP voice frames. The sync

management frame replaces the first MELP voice frame in the superframe structure, adding no additional overhead to the 2,400 bps data stream. At the receiver, the missing voice frame must be compensated for using a frame replacement strategy upon receipt of the superframe.

The secure 2,400 bps MELP *Blank and Burst* mode superframe contains a 54 bit sync management frame that replaces the first MELP voice frame in a manner similar to the *Clear* MELP mode. As before, the first frame is discarded prior to the insertion of the sync management frame into the superframe. The *Blank and Burst* superframe includes a sync management frame followed by 23 MELP voice frames.

The secure *Burst without Blank* MELP mode superframe contains a 56 bit sync management frame followed by 24 MELP voice frames. The extra 2 bits in the sync management frame are required for octet alignment of the superframe due to the additional MELP frame. In this case, the sync management frame does not replace the first MELP voice frame; the sync management frame is inserted in between the voice frames. Thus, no correction is needed in the receiver. By inserting the additional sync management frame into the superframe the resulting bit stream is more than 2,400 bps.

The secure *Asynchronous Data* mode is structured like the secure MELP *Burst without Blank* mode. Sync management frames are inserted between data frames, without replacing data frames. Superframes consist of a 64 bit sync management frame followed by 14 asynchronous data frames, 11 bytes each. Each byte in a data frame has start and stop bits. Therefore, to achieve 2,400 bps with the additional sync management frame, start and stop bits are removed [24].

An illustration of the Clear MELP mode is found in Figure 2.8.
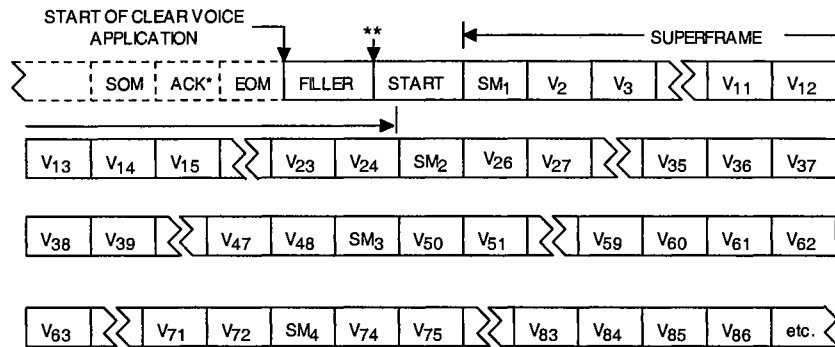
Figure 2.8 Clear MELP Voice Transmission Format [24].

The Start of Message (SOM), Acknowledgement (ACK), and the End of Message (EOM) frames noted in Figure 2.8 are components of FNBDT's signaling scheme to provide error correction [24]. The FNBDT protocol requires that all full bandwidth traffic is preceded by the FILLER and START pattern.

FNBDT Counter-Mode Encryption

To an input data stream FNBDT can be viewed as a black box. For a serial bit stream input it produces a formatted serial bit stream output. The input to FNBDT for voice operational modes will consist of a sequence of encoded MELP frames, 54 bits each, arriving at a rate of approximately 44 frames per second. The resulting output is a stream of formatted FNBDT superframes each consisting of multiple voice frames and additional data based on the current operational mode. For secure modes the formatted superframe data are a result of the FNBDT encryption scheme.

21

The proposed encryption scheme for FNBDT is counter-mode cryptography [76].

Encryption via counter-mode cryptography is achieved by taking a plaintext M of

arbitrary length, a key K, and a counter ctr. First, the keypad $E_k(ctr)$ is created by

encrypting the counter with a block cipher such as Rijndael using the key. To encrypt the

plaintext, resulting in ciphertext C, the plaintext is XOR (exclusive-or) with the first $|M|$

bits[1] of the keypad [76]. To decrypt the ciphertext, which is now a function of the

counter, the reverse transformation is accomplished by XOR-ing the ciphertext and the

first $|M|$ bits of the keypad. Figure 2.9 shows counter-mode encryption and decryption.
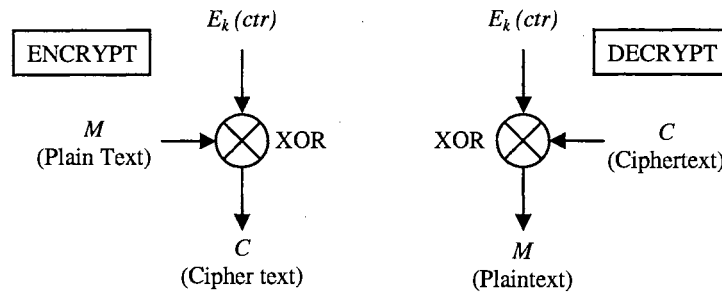
Figure 2.9 Counter-mode Encryption and Decryption.

The application of counter-mode encryption for FNBDT/MELP implementation is

as follows. The FNBDT transmitter terminal generates a keypad by encrypting a

transformed counter as described above. Each outgoing 54-bit MELP frame is XOR-ed

with the first 54-bits of the next available key in the keypad. FNBDT sends the encrypted

MELP data by stripping any padding bits and transmitting it as part of a superframe. At

the receiver FNBDT reformats the incoming data stream by first removing the sync

management frame and encrypted MELP frames from the superframe structure. Then the

---

[1] $|M|$ regards M as a nonnegative number, written in binary, most significant bit first.

sync management frame's counter value is used to perform the reverse encryption

operation. The encrypted MELP frame is XOR-ed with the appropriate key in the keypad

to produce the corresponding clear (unencrypted) MELP frame. Figure 2.10 illustrates

the application of counter-mode encryption for an FNBDT implementation.

Sending Counter   **Receiving Counter**

| $T(0) \rightarrow T_0$ |
| $T(1) \rightarrow T_1$ |
| $T(2) \rightarrow T_2$ |
| $T(3) \rightarrow T_3$ |
| ... |

Sending Key Buffer

| EB C5 49 EC 24 D1 ... |
| 64 05 12 7E 24 2D ... |
| CE 28 39 5F 73 23 ... |
| 52 5A87 FO 74 4F ... |
| ... |

Receiving Key Buffer

| EB C5 49 EC 24 D1 ... |
| 64 05 12 7E 24 2D ... |
| CE 28 39 5F 73 23 ... |
| 52 5A87 FO 74 4F ... |
| ... |

| $T(0) \rightarrow T_0$ |
| $T(1) \rightarrow T_1$ |
| $T(2) \rightarrow T_2$ |
| $T(3) \rightarrow T_3$ |
| ... |

Block Cipher   **Block Cipher**

Key: K   **Key: K**

| 12 25 |
| D7 BE B6 03 |
| 36 48 97 |
| 5E FA D8 |
| ... |

XOR

| F9 EO |
| B3 BB A4 7D |
| F8 60 AE |
| 0C A0 5F |
| ... |

XOR

| 12 25 |
| D7 BE B6 03 |
| 36 48 97 |
| 5E FA D8 |
| ... |

Clear Data   **Encrypted Data**   **Clear Data**

Figure 2.10 Application of Counter-mode Encryption for FNBDT Implementation.

It becomes important with counter-mode encryption to protect how the counter is

formed or how the keypad $E_k(ctr)$ relates to the counter. Protection is also required for

how this information is communicated between sender and receiver. FNBDT

accomplishes this by using a codebook (containing the counter) and maintaining a state

vector (to increment the counter) at both the transmitter and receiver.

FNBDT uses sync management frames as the vehicle for transmitting the state

vector information for FNBDT counter-mode cryptography. The state vector includes a

binary counter which increments every voice frame. The binary counter is arbitrarily

divided into a long term (MSBs) and short-term (LSBs) components. The long-term

component is distributed over three superframes; split into three consecutive sync

management frames, while the short-term component is included in each sync

management frame. So, each sync management frame includes the short-term component

and a partial long term component of the state vector, Figure 2.11.

Header (PN)	Partial Long-term Component	Short-term Component	CRC

|←—— 16 bits ——→|←——— 16 bits ———→|←—— 14 bits ——→|←— 8 bits —→|

|←——————————————— 54 bits ———————————————→|
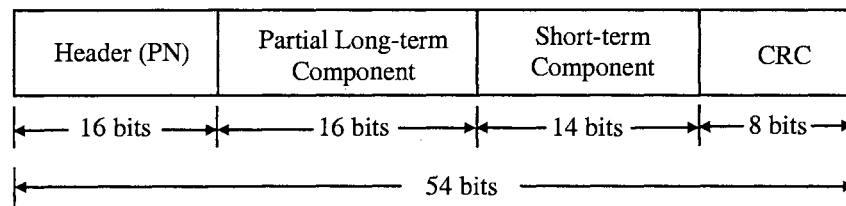
Figure 2.11 54-bit *Blank and Burst* Sync Management Frame.

There are numerous benefits of using a counter-mode cryptographic scheme with

FNBDT. First, there is no error extension due to bit or frame errors. For example, if a

frame is lost within a given superframe (assuming the lost frame is identified) it is

possible to decrypt the remaining frames in the superframe by simply incrementing the

counter to reflect the lost frame. If synchronization is lost, recovery is possible as early as

the start of the next superframe (next sync management frame) and never worse than

three superframes (3 sync-management frames) later. There is no additional delay

imposed by the encryption. Finally, counter-mode encryption is easily adapted to

vocoders with different frame sizes and frame data of arbitrary length.

FNBDT Discontinuous Voice Transmission

Discontinuous voice transmission (DTX) voice involves the transmission of talkspurts only in a voice conversation. The FNBDT specification describes some general requirements for DTX. Like other voice communication applications, FNBDT's DTX distinguishes between voice (active) speech and silence speech. When voice speech is detected it is transmitted to the destination, however, when silence is detected transmission is halted and silence/comfort noise is played out at the receiver. Features defined in the FNBDT signaling plan for DTX voice operation include, Voice Activity Detection (VAD), Grace Period, Blank Period, Comfort Noise, and Re-Start, [24]. Figure 2.12 shows an example of the DTX voice operation, highlighting the features listed above.
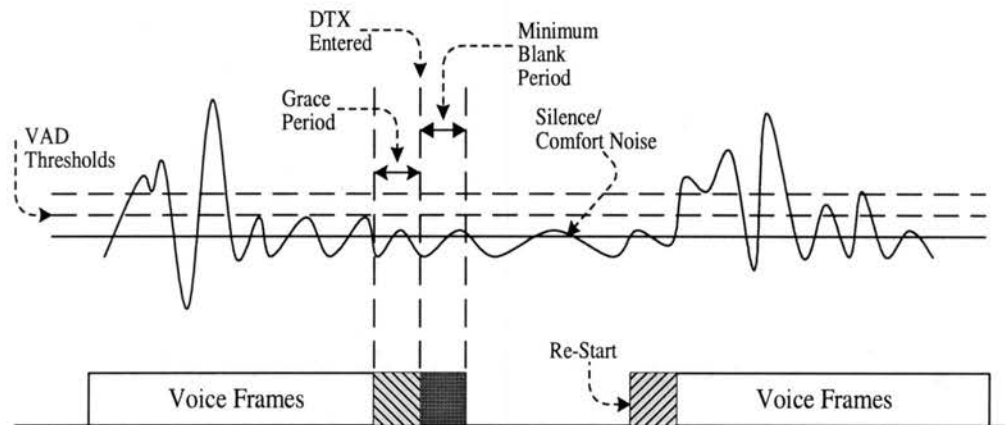
Figure 2.12 FNBDT DTX Voice [24].

The upper portion of figure 2.12 shows a sample speech signal, while the lower half of the figure 2.12 illustrates the corresponding voice activity classification and DTX state of the speech signal above. For the bottom portion of the figure the regions within the areas labeled Voice Frames correspond to portions of the speech signal that is considered

active voice segments (frames), determined by the DTX algorithm using VAD

thresholds. The shades regions that have backward and forward slashes indicate grace

period frames indicating the end of a voice region and the start of a voice region,

respectively. The more solid shaded region provides an example of an additional frame

that is required to achieve a minimum number of grace period frames before active voice

transmission can cease. All other regions on the lower portion of the figure refer to

inactive voice regions or silent region where voice transmission is discontinued and

comfort noise is to be played out at the receiver. A more detailed discussion of DTX is

given below.

The FNBDT signaling plan has a default Voice Activity Detection (VAD)

algorithm that determines if the input audio signal is active speech or silence. The

algorithm applies an energy measure to make this determination. The energy is calculated

for each frame using the input frame's amplitude, where A is a vector of one frame of

input data, A^H is the complex conjugate transpose of A, equation (2.1),

$$Energy = \sqrt{\left(A^H \times A\right) \Big/ \left(FrameSize\right)},$$ (2.1)

and *FrameSize* is the number of samples per vocoder (MELP) frame. Based on the

energy of the input signal, minimum and maximum energy levels are set. These values,

seen as *LowRMS* and *HighRMS* in equation (2.2), are used to set the threshold for

comparison with the current frame's energy level. The algorithm establishes that silence

is present if more than 4 consecutive frames have energy below the threshold. At the

receiver, comfort noise is played out during silence periods. This continues until the

energy level of the input speech goes above the threshold signaling the end of silence and

Re-Start of voice traffic. FNBDT accounts for low energy anomalies by slowly

increasing the *LowRMS* energy level with each frame until the *LowRMS* energy value is

reset.

$$Threshold = (0.07 \times HighRMS) + (K \times LowRMS) \tag{2.2}$$

After silence is detected and before DTX mode is entered a mandatory Grace

Period of at least two vocoder frames must have elapsed. This variable period is made up

of silence/background noised, transmitted as parameterized silence frames recognizable

by the vocoder. For MELP *Blank and Burst* operational mode the format of these Grace

Period MELP vocoder frames are comfort noise frames, with all parameters set to zero

except MSVQ1, Gain2, and Sync, shown in table 2.2.

Table 2.2 MELP Comfort Noise Parameter Values [24]

MELP Parameter	Value
MSVQ1 (line spectral frequencies)	Default value of previous vocoder frame or average of some number of previous frames
MSVQ2 (line spectral frequencies)	Set to 0
MSVQ3 (line spectral frequencies)	Set to 0
MSVQ4 (line spectral frequencies)	Set to 0
Fsvq (Fourier magnitudes)	Set to 0
Gain1 (gain)	Set to 0
Gain2 (gain)	Default value of previous vocoder frame or average of some number of previous frames
Pitch (pitch – overall voicing)	Set to 0
Bp (bandpass voicing)	Set to 0
Sync (sync bit)	Continue Alternations

After the grace period there is a minimum Blank Period of silence. The Blank Period is variable, defined as *n* vocoder frames, and defines the minimum number of silence vocoder frames or period that must elapse before voice traffic can proceed. For MELP *Blank and Burst* mode *n* is equal to 2.

The FNBDT signaling plan encourages the use of comfort noise during silent periods to alleviate the abrupt change from background noise to silence during the conversation. Comfort noise is suggested as replacement for silence played out at the receiver. The MELP *Blank and Burst* mode uses the comfort noise parameters shown in table 2.2, with these frame parameters being replications of the Grace Period frames received from the transmitter. Average vocoder parameter values from the transmitter are used for the Grace Period frames. Once voice activity is again detected and the minimum Blank Period is achieved, a Re-Start message is sent to the receiver to signal the beginning of voice traffic. The Re-Start message is the sync management frame for MELP *Blank and Burst* mode.

The FNBDT signaling plan defines a Voice Activity Factor (VAF) used to verify the proper operation of implementations of the Voice Activity Detection algorithm. The VAF is a performance criterion that indicates the percentage of voice activity present as detected by the DTX algorithm. The FNBDT signaling plan defines specific test speech files, available through the signaling plan developers, for application developers to test their respective Voice Activity Detection algorithm. A VAF of ≤ 0.6 is required for the respective test vectors using the DTX algorithm for MELP *Blank and Burst* mode.

FNBDT will provide real-time voice communication over packet networks therefore, the next chapter provides a brief introduction of voice transmission across

packet networks. Qualities of Service challenges are also presented that identify areas

that may potentially degrade the performance of FNBDT and other voice over IP

systems. Also an overview of network queuing behavior is provided, which make up the

mechanisms for forwarding data over packet networks.

CHAPTER III

TRANSMITTING VOICE OVER PACKET NETWORKS

Voice is traditionally transmitted over the Public Switched Telephone Network (PSTN). The telephone network is a circuit switched network that allocates a channel for each two-way conversation. When a call is initiated, a circuit is assigned for the duration of the call, giving the talkers a reliable, low latency connection. Other circuit switched communication channels that carry voice traffic include radio, wireless, and cellular. In contrast to circuit switched networks, packet switched networks were not designed for voice traffic; they were specifically designed for data traffic where each channel is shared by all users. Data traffic is bursty, sent in short spurts; therefore, sharing a channel with multiple data connections provides efficient utilization of the channel, but also provides for bandwidth limitations. Moreover, information is transmitted as packets in contrast to a continuous data stream as with the other communications channels.

Transmission of voice data over packet switched networks, particularly those supporting the Internet Protocol (IP), is possible primarily because of the "good fit" between low bandwidth audio codecs (audio coder/decoder) and the characteristics of a packet network. The frame oriented output of the audio codec provides a good match to the packet format of the packet network. The high compression ratio achievable by many low rate codecs makes low bandwidth transmission across packet networks possible.

As the previous discussion suggests, transmitting voice over a packet network involves two basic parts, the audio codec and the packet network. The primary goal, as with any communication system, is to maintain a continuous audio stream between end

users (the connection). In order to provide this continuous stream of voice data at both

ends, the system must handle the transition at the transmitter from a continuous speech

data stream to an intermediate compressed and packetized format suitable for

transmission, and back to a continuous speech data stream at the receiver. The

mechanism that provides that bridge between the continuous data stream and the

packetized, and usually compressed, format is the audio codec as shown in Figure 3.1.
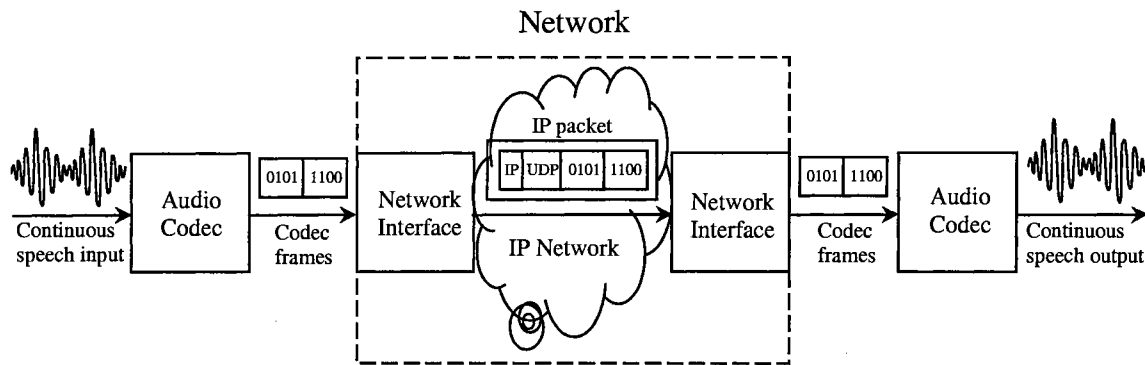
Figure 3.1 Audio Codec and Network. Illustration of the stages of an end-to-end IP network voice communication stream using an audio codec.

Quality of Service (QoS)

In any communications system, especially the PSTN, one does not expect large

end-to-end delay, missing parts of speech, and talker overlap (one talker suppressing the

other speaker's speech). These are all major Quality of Service (QoS) issues that arise

when attempting to transmit voice over a packet network. QoS describes not only the

quality of communication exhibited by the system, but also the reliability of the

connection (communication link). The major factors affecting QoS of a voice over packet

switched system are delay, packet inter-arrival time, and loss rate [28]. Other factors

include out-of-order packets, packet overhead, and network efficiency. These factors

describe the transmission and receipt of packets (the mechanism by which voice data is transmitted) as they traverse the network. Delay, packet inter-arrival time, and loss rate affect the quality of audio received by the end users of the system. The reference for audio quality is the performance of the audio codec - the naturalness and intelligibility of synthesized speech. However, the perceived end-to-end system audio quality lies in the QoS provided by the system (in response to the communication link characteristics). For instance, an end-to-end system with excessive delay, or extended periods of signal "drop-outs" due to excess packet loss, would be considered a low quality audio system regardless of the performance of the audio codec.

End-to-end delay, also referred to as latency, has a direct impact on quality of communication of any two-way communication system. Delay affects the amount of talker confusion and difficulty in maintaining a normal conversation. Delay also gives rise to talker overlap and mutual silence. With increasing delay, talkers are less likely to respond to questions, pauses or interruptions. Talker interruptions or talker overlap also tends to increase with delay. Further discussion on the effects of delay on user conversations can be found in [29] and [30].

Variation in the packet inter-arrival time, the delay between successive packets, is referred to as jitter. If the packet inter-arrival time is too long there can be gaps between words or syllables of the output speech, thus being a nuisance to the listeners. This may also cause problems in the audio playback by not being able to supply a continuous stream of data for the audio playback device. Therefore, adequate buffering is required to supply the audio playback device with continuous data.

Finally, in a real-time voice communication application, when loss is excessive there is degradation in QoS ([31] [32]). Lost and out-of-order packets can cause confusion, and depending on the seriousness of the conversation, can be costly.

These and other QoS factors pose a challenge when developing a high quality voice over packet network system. There are several tradeoffs in developing a system that reduces the adverse effects of these factors [5] [6]. There are a number of inter-related design tradeoffs to consider, with delay being the most constraining factor. End-to-end delay has the greatest impact on a voice communication system. For instance, end-to-end delays less than 300ms are considered good quality by general users, and delay greater than 450ms is considered unacceptable for voice applications [6]. Tradeoff challenges for voice over packet applications, shown in figure 3.2, include robustness to packet loss versus delay, packet overhead versus network efficiency versus system delay, jitter buffering versus delay, etc.
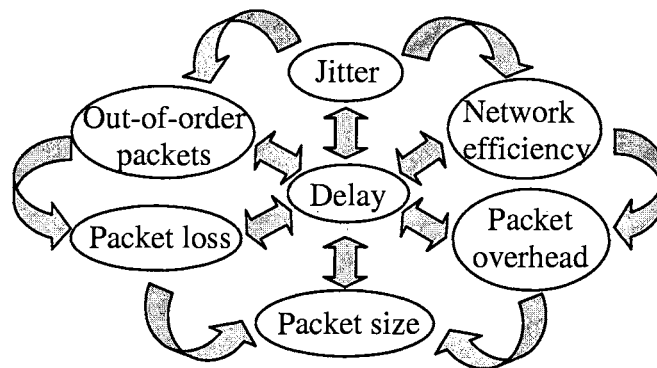
Figure 3.2 QoS Tradeoffs

Delay, packet inter-arrival time, and loss rate are a few of the many factors that influence the QoS of a voice over packet system. This research will examine some of

these QoS factors and others that are related to the Internet and CDMA cellular data networks. An attempt will be made to characterize these factors through the use of empirical statistics and network models found in the literature in order to determine and predict how they will affect FNBDT and MELP.

Models Based on Queuing Theory

As packets are sent from source to destination in a packet switched network they will potentially be routed through several intermediate network nodes or routers. The delay, loss, and jitter experienced by packets will largely be due to queuing and congestion at routers. A packet may experience several different delay scenarios as it makes its way to its destination, including processing, queuing, and transmission delay at each node as well as a propagation delay through the link between nodes. If there is network congestion at the router, packets may encounter a large delay or even be dropped (by the router). Large delays may cause packets to arrive at their destination too late, after the frame's scheduled playback time, and thus be considered lost. Packets may experience varying delays and arrive at their destination with jitter (varying inter-arrival time). Sequential packets may take different routes (to avoid congested links) and experience different delays, arriving at their destination out-of-order. Packets may encounter random bit errors at routers or on the communications link rendering them unusable at the receiver (and discarded).

Mathematical models have been traditionally used to evaluate network delay and congestion. Simplified queuing models based on statistics are the most common techniques used. Due to the great complexity, vastness, and heterogeneity of networks such as the Internet, many of the QoS statistics mentioned above are obtained using

34

empirical studies. The statistics derived from these studies in turn are used to develop better statistical models. Queuing models in general remain the basis for network characteristics approximations and provide valuable quantitative results and worthwhile insights [33].

A typical queuing system is made up of a receiver, buffer, server and a transmitter as shown in Figure 3.3. Arriving packets are received and placed in a buffer. When the packet reaches the head of the buffer it is processed by the server and transmitted.
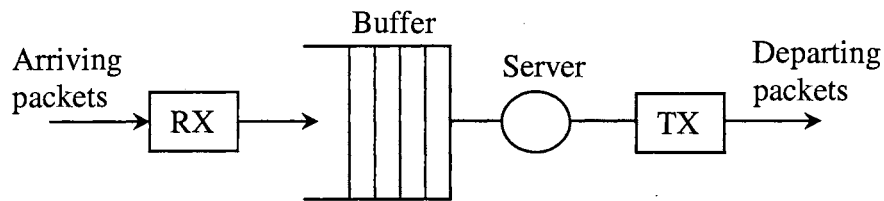
Figure 3.3 Simple Queue System.

The basic components of a queuing model are arrival process, service process, number of servers, queue capacity, and the queuing discipline. The arrival process is randomly distributed and describes the manner in which customers[2] arrive at the queue. The service process is randomly distributed and specifies how service is rendered. It is specified by the time taken to service a customer. The number of servers represents a constant value, a system can have a single server or a number of parallel servers. The queue capacity can be finite or infinite and gives the maximum places available in the

---

[2] Customer is a general term used for the unit that needs the service provided by the queuing system. In this discussion the customers are packets.

queue for waiting customers. The queuing discipline is the manner in which the units are taken for service (priority for servicing customers).

A simple queuing system that consists of a single queue with a single server is the M/M/1 queuing system. The notation (arrival process/service process/number of servers) is common among queuing nomenclature. Some common queuing models are described by varying the arrival and service processes according to the following distributions: (M) for memoryless with an exponential distribution, (G) for general distribution, and (D) for deterministic. For example, in the above M/M/1 queuing model the arrival and service processes are both memoryless with an exponential distribution. An M/D/1 model would have a memoryless arrival process that is exponentially distributed with a constant service time distribution. Likewise, a M/G/1 model would have a memoryless arrival process but with a general service time distribution (not necessarily exponential).

The M/M/1 model may theoretically have an infinite queuing capacity with a first-in-first-out (FIFO) queuing discipline. The arrival process (signified by the M) is Markovian (memoryless). Packets arrive according to a Poisson process with rate λ. The inter-arrival times are independent and exponentially distributed with probability,

$$P\{t_{n+1} - t_n \le t\} = \lambda e^{-\lambda t}, \quad t \ge 0 \tag{3.1}$$

where λ is the average (mean) number of arrivals per second.

The service process for this model has an exponential distribution with a service rate of μ and a probability density function,

$$P\{S_n \ge t\} = \mu e^{-\mu t}, \quad t \ge 0 \tag{3.2}$$

where μ is the mean service rate (in customers served per unit time) and S_n is the service time for the nth customer. The service time is the packet transmission time given by

$S_n = L/C$, where L is the packet length (in bits) and C is the link transmission capacity (in bits/sec).

An important variable in queuing models is the system utilization factor, ρ, that uses the mean arrival rate λ and the mean service rate μ as shown in the calculation

$$\rho = \frac{\lambda}{\mu}. \tag{3.3}$$

When $\mu > \lambda$, the service rate exceeds the arrival rate and $\rho < 1$. This implies that the server is in an uncongested state, so packets are serviced before the queue reaches its capacity. However, if $\mu < \lambda$ then $\rho > 1$, the server cannot keep up with the arrival rate, and therefore the server is in a congested state. While in a congested state the queue will reach its capacity before packets can be serviced. This may cause the queue to drop packets due to buffer overflow.

For the uncongested M/M/1 queue, the average number of customers in the queue (queue length), N, is given by

$$N = \frac{\rho}{1-\rho} = \frac{\lambda}{\mu - \lambda}. \tag{3.4}$$

Little's Theorem relates N, the average number of customers in the queue (queue length) with the average customer time in the system (average delay) T, by

$$N = \lambda T. \tag{3.5}$$

Accordingly, the average delay can be derived using Little's Theorem and N (queue length), shown in (3.6).

$$T = \frac{1}{\mu - \lambda} \tag{3.6}$$

Further, the average waiting time in the queue, W, is the average delay, T, less the service

time $1/\mu$ given by

$$W = \frac{1}{\mu - \lambda} - \frac{1}{\mu} = \frac{\rho}{\mu - \lambda} \qquad (3.7)$$

In (3.6) for average delay, as long as $\mu \gg \lambda$ there will be minimal delay

experience by customers because the server will be able to service customers faster than

they are arriving. As $\mu \to \lambda$ the delay experienced by the customers increases. When

$\mu < \lambda$, customers may be discarded or experience infinite delays because the server has

reached a congested state. As the delays increase or vary at the queue, packets can arrive

at their destination with unwanted jitter between successive packets. If congestion occurs

at nodes (queues), successive packets may be re-routed to avoid the congested areas.

These packets may arrive at their destination out-of-order with respect to their

predecessor(s) that may be caught in a congested or highly delayed queue.

To gain an understanding of particular networks that will be used in this

investigation and to identify potential behavior that might degrade the performance of

FNBDT, the next chapter provides properties and characteristics sited in the research

literature for Internet and CDMA networks. This information along with "real-life"

collected network statistics (chapter 5) will be used to identify and develop models to

simulate network behavior.

CHAPTER IV

NETWORK PROPERTIES AND CHARACTERISTICS

Channel Errors

Channel errors are dependent on the physical medium and equipment used to transport data. Physical media can range from copper wire to optical fiber to air for wireless communication systems. Optical fiber channels provide reliable connections with typical bit error rates of 10^{-9} to 10^{-12}. Other wireline (non-wireless) channels (analog and digital) offer bit error rate that range from 10^{-3} to 10^{-7}, with purely digital channels achieving rates closer to the upper bound. In contrast to wireline channels, wireless channels are very unreliable and characterized by high bit error rates (reported as high has 10^{-2}).

Wireless channels are more complex than wireline channels due to their susceptibility to noise, interference, obstructions, multipath, etc. Signal propagation in wireless channels can be attenuated or corrupted with bit errors due to long range or short range propagation effects. Long-range propagation, also called free space propagation, corresponds to unobstructed line-of-sight paths where path loss, earth's curvature, terrain, and the ionosphere affect the signal. Short-range propagation typically corresponds to urban environments where there is potentially high mobility and no line-of-sight paths from the mobile host to the base station. The effects on short-range propagation include interference (co-channel), shadowing, and multipath fading.

For terrestrial mobile radio systems, such as CDMA cellular data that operate over a frequency range of 0.3 to 30G Hz, long-range propagation factors (path loss,

ionosphere, etc.) have no effect on the signal propagation [34]. Moreover, in this study the focus will be on signal propagation in cellular data networks over a short distance, from base station (BS) to mobile host (MH), thus further nullifying the long-distance propagation effects as well as slowly varying shadow losses.

Short-range wireless systems typically yield high bit error rates that tend to be bursty in nature, meaning that errors occur in large sequential groups. These bursts are synonymous with multipath fading that follows a Rayleigh distribution ([35]), which has the probability distribution function given by (4.1)

$$p(r) = \frac{r}{\sigma^2} e^{-\frac{r^2}{2\sigma^2}} \quad (0 \le r \le \infty) \tag{4.1}$$

For simplicity of simulation and modeling, effects of interference[3], and shadowing are often ignored.

Internet Properties

Packet loss can occur as a result of buffer overflows at network nodes due to heavy network loads. It can also occur due to bit errors (caused by noise) incurred by packets as they make their way through the network. There have been numerous studies on Internet packet loss statistics including ([36] [28] [37] [32] [83] [95] [96]). Packet loss is known to have some correlation with packet size, network congestion[4] and network delay. According to [38] and [39] packet loss of 10% is not uncommon, and losses of up to 40% are possible [40] [83]. Internet packet loss is bursty and correlated, meaning that packets are usually lost in sequential groups, and if packet n is lost then there is high

---

[3] For this study of CDMA cellular data, CDMA is theoretically unaffected by interference because of its spread spectrum coding scheme.
[4] Network congestion is also correlated with the time of day [42]. Curtain times of day have high network usage, thus network congestion occurs.

probability that packet $n+1$ will be lost ([32] [36] [28] [41] [42]). Borella, Swider, Uludag, and Brewster ([32]) found that packets lost in bursts account for the majority of the packets lost over a long period of time. But, overall packet loss gaps are usually close to one or two packets ([43] [36][44]).

Internet delay describes the delay a packet encounters as it is transmitted form source to destination across the Internet. This delay is highly variable; Internet delays can range from a few milliseconds to infinity (never reaching its destination). Delay is not only dependent on the total distance (geographical) between source and destination, [42], but also on the number of network nodes (hop count) encountered by packets between source and destination. Kostas et al. ([31]) reports that observed delays are more correlated to actual "hop distance" between source and destination than geographical distance (due to potentially higher levels of congestion at the shorter links). Delay is also dependent on the time of day as it affects network congestion ([31][42]). High levels of congestion yield higher delays (as well as loss). These factors make predicting exact delays from a particular source and destination difficult (if not impossible). There have been several empirical studies that have monitored delay over various network paths, which include interstate, cross-country, and international Internet connections. Kostas et al. ([31]) obtained delay measurements from a 2000-mile Chicago-to-California link. Under poor network conditions they measured round-trip delays between 75 - 300ms. Yet, most of their round-trip delay measurements were between 70 and 160ms, with unidirectional delays of 30 – 100ms. Su, Srivastava, and Yao ([28]) measured delays from the University of Manitoba in Canada to the University of Minnesota in the U.S.A

and found less than 2% of data had delays between 70 – 200ms, with most delays under 70ms.

As packets make their way through the Internet they may reach their destination out-of-order. This phenomenon occurs because of the varying delays that packets may encounter along the paths to their destination. The out-of-order packet behavior of the Internet has been looked upon as unusual, but Paxson ([41]) found that out-of-order packets are quite common in the Internet, occurring usually in groups of one or two packets, but sometimes as large as dozens of packets. Analyzing packet traces of 20,800 TCP connections between 35 Internet sites, dividing them into 2 data sets, Paxson observed 36% of the connections in one data set and 12% in the other had at least one packet delivered out-of-order. Paxson attributed these findings of out-of-order packets to "route fluttering", where a packet takes multiple routes to reach the destination. Paxson also states that out-of-order delivery of packets varies from site-to-site. Bennett, Partridge, and Shectman ([45]) on the other hand attribute a higher amount of out-of-order packets to parallelism of Internet components and links, where packets can follow multiple paths within a device or logical link and thus experience varying delays, based on load and configuration of these devices or logical links. Bennett *et al.* performed a test to determine how many sites experienced out-of-order delivery of packets. They "pinged" 140 Internet sites and concluded that the probability of a session experiencing out-of-order packets was 90%. Contrary to the above statistics on out-of-order packet delivery over the Internet, experiments conducted using a Network Performance application (described later) shows a very low probability of out-of-packet delivery. Several experiments of packet trace runs over both Internet and CDMA links, of varying

durations, at different times of day were conducted and out-of-order packet delivery were rare or non existent in most cases. Nevertheless, it is clear from the results of Paxson and Bennett *et al.* that out-of-order delivery of packets is not an anomaly but a characteristic that must be considered for any application using the Internet.

These measurements of delay, loss, and out-of-order packets seem reasonable, but extreme caution must be taken when labeling them as "typical" values. Because of the vastness, heterogeneity, and constantly changing conditions over time (hardware, traffic levels, links, etc.) of the Internet, Floyd and Paxson ([46]) suggest that the focus should remain on the *invariant*[5] properties of the Internet. One particular and important invariant is the Internet's *self-similar* nature with respect to traffic characteristics. Self-similarity means that its characteristics are similar on a wide range of time scales. It represents one property of fractals – an object whose appearance is unchanged regardless of what scale it is viewed, figure 4.1. For a time series or distribution, self-similarity implies that an object's correlation structure remained unchanged when viewed at varying scales. Thus, it can exhibit long-range dependence (long-term correlation), often characterized by heavy-tailed distributions. Heavy-tailed refers to the *upper tail* or extreme values of the distribution, where an arbitrarily large amount of the probability mass may be present in the tail of the distribution.

---

[5] Floyd and Paxson ([46]) define the term "invariant" as "some facet of behavior which has been *empirically* shown to hold in a very wide range of environments".
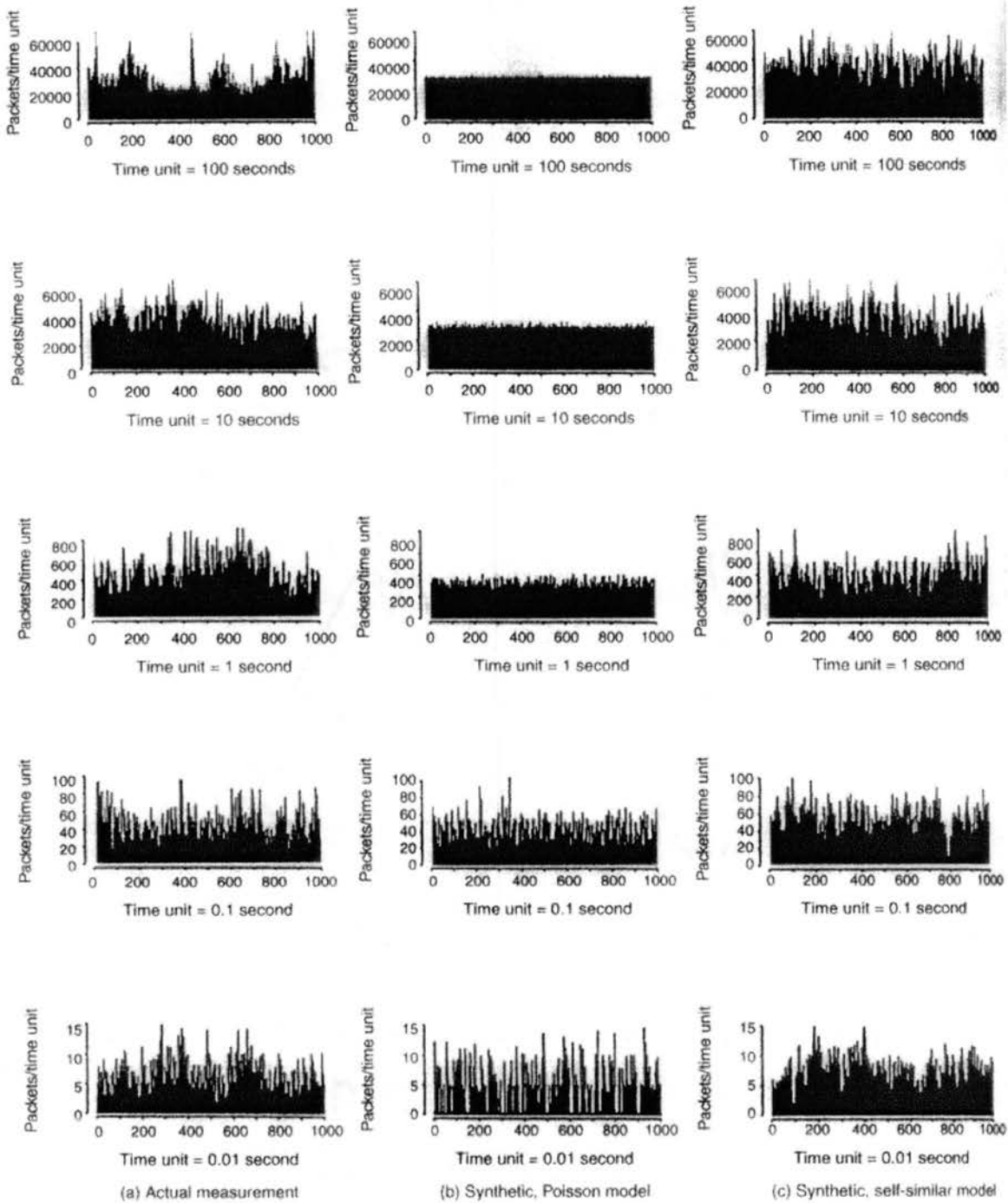
Figure 4.1 Comparison of actual and synthetic Ethernet traffic [77]

For a stochastic process, $x(t)$, to be defined as self-similar, the following

conditions for the mean, variance, and the autocorrelation must exist (for any $n > 0$).

44

$$E[x(t)] = \frac{E[x(nt)]}{n^H} \qquad (4.2)$$

$$\text{var}[x(t)] = \frac{\text{var}[x(nt)]}{n^{2H}} \qquad (4.3)$$

$$R_{xx}(t,s) = \frac{R_{xx}(nt,ns)}{n^{2H}} \qquad (4.4)$$

where H is the *Hurst* parameter, also known as the self-similarity parameter. It indicates the degree of self-similarity, or the measure of long-range dependence of a stochastic process. H varies from 0.5 to 1, with a value of 0.5 indicating few self-similar characteristics. As H approaches 1, the degree of self-similarity (long-range dependence) increases.

It has been reported by ([47] [48] [49] [50]) that Wide-Area (WAN), Ethernet (LAN), World Wide Web, and multimedia network traffic each exhibits self-similar behavior. Through empirical studies they observed that traffic patterns are bursty, showing correlation between arrivals, with the observed correlation having long-range dependence, remaining unchanged regardless of time scale viewed. Paxson ([41]) as well as Borella, et al. ([51] [52]), and Li and Mills ([53]) found packet delay to be extremely bursty and self-similar. Paxson and Borella *et al.* also suggest that the degree of self-similarity, and thus packet delay, is correlated with network congestion, which spans a wide range of time scales. Borella *et al.* ([51]) goes on to suggest that the degree of self-similarity on a round trip path is positively correlated with the packet loss for that path. In [32], Borella *et al.* finds that packet loss is correlated and exhibits extreme heavy-tailed behavior (long-range dependence).

CDMA Cellular Data Properties

In this study one focus will be on CDMA cellular data properties at the packet level. CDMA cellular data networks have poor connection quality and are associated with higher error rates than that of wireline networks. Typically in wireline networks like the Internet, the bit error rate (BER) for the physical medium is low enough, approximately 10^{-12} for fiber, that the focus of network performance is primarily on the network nodes and equipment (queuing theory). In the analysis of wireless networks, channel (air – physical layer) properties require more attention because of their high error rates and unreliable characteristics. The focus of this research is on the short-range properties of cellular data networks, from BS to MH, therefore the properties and assumptions in this paper will neglect long-range effects such as path loss, earth curvature, terrain, etc.

The CDMA cellular data network is part of the IS-95 ([54]) standard, the IS-99 ([55]) and IS-707 ([56]) (extended version of IS-99), as well as the IS-2000 3G standard [78]. The IS-95 CDMA cellular system was originally established for voice communication, but its digital system architecture and its capability of handling variable data rates made it well suited for either circuit or packet switched data [57]. The IS-2000 (cdma2000) cellular system however was designed to accommodate data traffic. The CDMA cellular system is connection-oriented, but the data (packet) service is connectionless, provided by the Radio Link Protocol (RLP). RLP brings up and drops CDMA packet channels as needed with a 1 to 2 second channel setup time. RLP is the mechanism for transporting variable-length packets across the IS-95 and IS-2000 radio channels. The IS-2000 standard (cdma2000) provides a physical layer protocol that further divides RLP packets into smaller Logical Transmission Units (LTU's) for

delivery of large RLP frames at high data rates [79]. The RLP information bits range from 172 bits at 9600 bps to 20712 bits at 1.0368M bps for IS-2000 [78]. These packets can contain datagrams from IP or any other desired datagram protocol suite [57].

The CDMA cellular data networks have a packet loss rate of 10^{-2} under full-load conditions, often given in terms of frame error rates (FER) (ie. FER of 1-2%) ([58] [57]). Data is transmitted along the CDMA traffic channel with a packet inter-frame time of 20ms ([54] [58] [57]). The end-to-end data transportation delay of a CDMA channel is "typically" 100ms [58].

Packet loss is bursty and is characterized by Rayleigh distributed fading (shadowing and handoff effects are ignored in this case for simplification). In order to combat the high error rates associated with wireless links at the physical layer, CDMA (IS-95 and IS-2000 standard) employs non-transparent RLP ([56]) at the data link layer. Non-transparent RLP is used to improve the link quality as seen by higher layers, with the goal of improving the link quality to that of wireline links (close to 10^{-8}) [58]. In actuality, non-transparent RLP has been reported to improve the link quality from 10^{-2} to 10^{-4} ([57] [59]). The down side to non-transparent RLP is the additional delay it places on the system, through the use of negative acknowledgements, packet retransmissions and re-ordering to correct for lost and out-of-order packets.

Network Delay Jitter Properties

The packet send rate of a voice over IP (VoIP) application is typically set (determined) by the codec frame accumulation rate. Under ideal network situations, packets will be received at their destination at a constant inter-arrival rate, equal to the send rate. However, as packets are transmitted on high-speed packet networks they may experience different delays as they travel from source to destination. As a result the packets arrive at the destination with varying delays (between packets) referred to as 'jitter'. In a VoIP application, when the inter-arrival jitter is too large it leads to starvation of the audio playback system. If the inter-arrival time between packets is too small, it could result in buffer overrun, where the application can not service the packet in time. This leads to packet loss and correspondingly a loss of voice data (seen at playback).

On packet-switched networks, jitter may result from packets taking different paths to their destination to avoid congested areas or failed links. However, jitter is primarily caused by varying queuing delays encountered by packets at routers (nodes). Network packets compete with other networks traffic at routers. Routers statistically multiplex incoming packets, which results in the varying delay. When several packets arrive at the node during a given interval (hypothetically at the same time), the node enters a busy state.

In a wide area network such as the Internet, when an application such as FNBDT transmits a stream of data, referred to as a tagged stream, it is multiplexed at network nodes (routers) with the other traffic on the network called background traffic. Internet IP network nodes have no priorities so they service network traffic on a FIFO basis. Therefore, the queuing delays and jitter experienced by packets are a direct result of wait

and service times of the packets in the queue. Jitter delays of tagged streams on packet networks (including ATM) are a sample of the queuing delays of the background traffic at the nodes [80] [81]. The relative size of the tagged stream is small compared to the background traffic, so it does not significantly contribute to the queuing delays. Li and Mills [82] further explain that the tagged stream samples the busy periods of the node. The tagged stream can capture the delay components as they relate to the queuing delays at network nodes. If the send rate, time between packets, of the tagged stream is smaller than the busy periods of the node, the tagged stream gets a sample of the queuing delays within the busy periods (more than one sample from the same busy period). However, when the send rate is larger than the busy periods then the tagged stream samples delays of different busy periods (many different busy periods). Essentially, larger send rates sample the busy periods while smaller send rates sample the delays within the busy periods, Figure 4.2. According to Li and Mills [82], this sampling describes the multi-structure of delay, which consist of a short term non-stationary component, delays within busy periods, and a long-term stationary component, delays of different busy periods. The long-term stationary component can be modeled as a Gaussian noise process because there is no correlation between the occurrences of busy periods of a queue. Moreover, the delays sampled within the queue for each busy period have no correlation with delays sampled from other busy periods. The multi-structure characteristic of network delays will be used to develop a model for jitter.

W(t)

Interval between sampling of delays within busy periods

t_{Bp-1} t_{Bp} t_{Bp+1} t

(a)

W(t)

Interval between sampling of delays of busy periods

t_{Bp-1} t_{Bp} t_{Bp+1} t

(b)
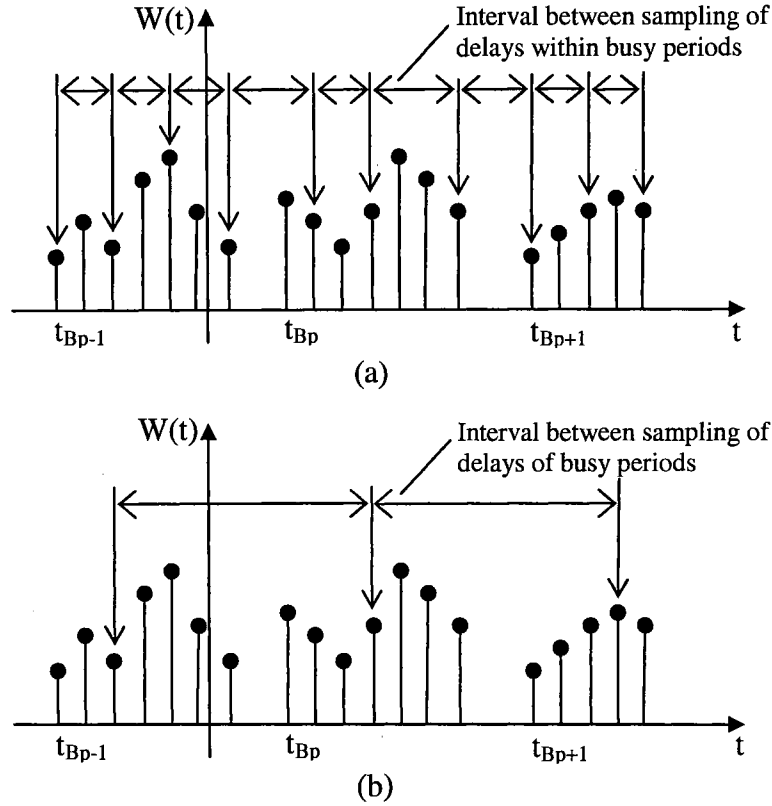
Figure 4.2 (a) Tagged stream sampling of queuing delay within busy periods and (b) Tagged stream sampling of busy periods of a queue. W(t) is an instance of a queuing delay process and t_{Bp} represents the start time of a busy period. [82]

Inter-arrival Delay for CDMA Cellular Links

For CDMA cellular data links, end-to-end and inter-arrival delays are dependent

on the RLP frame time. RLP frames transport data at 20ms intervals. The data length

within an RLP frame is limited (171 bits for IS-95, and 172 – 20712 bits for IS-2000), so

if the payload (IP headers and data) exceeds the RLP frame size it will take multiple RLP

frames to transport the data. In this implementation the number of RLP frames it takes to

transport a frame containing FNBDT data will depend on the number of MELP frames

included in a packet and IP header size. If multiple RLP frames are required to transport

the data, the end-to-end delay will increase by multiples of 20ms. For example, if it takes

2 RLP frames to transport an IP packet with FNBDT data, than the end-to-end delay will

reflect the additional 20ms to complete the data transport by RLP. Likewise, the inter-

arrival time will be 40ms between received (whole) packets. This will merely add to the

total end-to-end delay for each packet. Ideally, with data service operating over a circuit

switched network channel such as with CDMA cellular data links, the inter-arrival delay

should be constant except in lost packet cases, for instance with fades.

Average end-to-end delay for CDMA without RLP retransmissions and error

correction is characterized by equation (4.5) [113], which take into account the

probability of frame error in the air-link.

$$T_{ave} = T\left[\frac{2-(1-p)^k}{2(1-p)^k}\right] + D + (k-1)\tau, \qquad (4.5)$$

where T is the packet transmission interval (transport layer), p is the probability of frame

error at the air-link, k is the number of frames per (transport layer) packet transmitted

over the air-link, D is the end-to-end frame propagation delay over radio channel

(typically 100ms), and τ is the inter-frame time of RLP (typically 20ms). An example of

a 9600 bps CDMA data link used to transport MELP data at 2 frames per IP packet using

the UDP protocol is as follows. One packet containing 2 MELP frames would consist of

28 bytes of header (IP – 20bytes, and UDP – 8bytes) and 14 bytes of MELP data (7 bytes

each) for a total of 336 bits of payload for the CDMA data link channel to transport with

RLP. Therefore the packet transmission delay would be T= 336 bits / 9600 bps = 35 ms.

Assuming normal conditions the probability of frame error is 1% [58], with the number

of packets per frame $k = 2$ as it will take at least 2 RLP frames at (172 bits per frame) to

transport a packet containing 336 bits of data (header and payload). With an end-to-end

delay and an interframe time for CDMA links of 100ms and 20ms, respectively, the

average end-to-end delay would be T_{ave}=138.21 ms. If our packet size increased to

require $k = 4$ RLP frames to transport one packet, T_{ave}=178.94 ms.

From the discussion above it can be concluded that the inter-arrival delay for

packets based on RLP frame times on a CDMA link are based on the number of RLP

frames required to transport a packet (header and payload), shown in equation (4.6).

$$T_{\text{int}} = (k)\tau, \tag{4.6}$$

where k is the number of RLP frames required per packet, and τ is the inter-frame time

of RLP frames (typically 20ms). So, the inter-arrival time per packet in on a CDMA link

with RLP at 9600 bps with 172 information bits per frame and a MELP packet size of

336 bits (including IP and UDP header and 2 MELP frames, 7 bytes each) would be

40ms, as it will take 2 RLP frames at 20ms each to deliver one packet. However, as the

data rate increases for 3G IS-2000, so does the number of available information bits per

RLP frame. This decreases k, the number of RLP frames required to transport IP packets,

which results in decreased inter-arrival delay. Table 4.1 below shows IS-2000 cdma2000

data rates and corresponding RLP information bits.

Table 4.1 IS-2000 cdma200 data rates and corresponding information bits [78].

Data Rate (bps)	Information bits per RLP frame
1.0368M	20712
614.4k	12264
460.8k	9192
307.2k	6120
153.6k	3048
76.8k	1512
38.4k	744
19.2k	360
9600	172

Network Protocols

From a desktop PC-to-PC application (which is the basis of this research) the

Internet Protocol, or IP, is a natural protocol of choice, given IP's availability to desktop

platforms. IP handles routing of packets from a source system, potentially through several

intermediate networks, and finally to the destination end system.

There are many physical and data link protocols that can be used to transport IP

traffic. One of the goals of the designers of IP was to make IP portable, with the intention

of the possibility of making "IP over everything" [26]. Packet networks that use IP are

LANs, WANs, corporate intranets, the Internet, and cellular data networks.

Voice over packet (IP) systems utilize the Transport layer services to deliver end-

to-end messages. These systems can employ TCP or UDP. Both of these protocols use IP

at the network layer for the end-to-end delivery of messages. Delay constraints are so

tight in a real-time voice application that a connection-oriented protocol such as TCP

cannot be used. While TCP maintains reliable end-to-end transmission through its

timeout-and-retransmissions, a single packet loss causes TCP to decrease its transmission rate with either its congestion avoidance or slow start techniques ([60] [32]). Therefore, it is difficult for TCP to provide adequate throughput over a lossy path [32].

UDP, in contrast to TCP, provides connection-less transmission of stream data. UDP however does not provide reliable service; it is a "best effort" type of service. There are neither acknowledgments sent nor retransmission of data. It is up to the end user to provide this functionality. It is also up to the end user to provide a means of re-sequencing out-of-order packets. UDP is a good candidate for transmission of real-time voice data because it provides efficient, although unreliable, transport of data. Therefore, in this research UDP will be used exclusively in simulations and gathering of empirical data for voice transmission over the Internet and CDMA cellular data networks.

CDMA cellular data networks use RLP for end-to-end delivery (at the data link layer) of data frames across CDMA channels. RLP can be either transparent or non-transparent [56]. Transparent RLP provides maximum throughput for CDMA data channels with no error or out-of-order recovery of packets. It leaves the packet error recovery and packet reordering to the upper layers (network layers and above). Therefore the channel error rates experienced by the upper layers (including the application) are reflective of the raw CDMA channel. Non-transparent RLP provides error recovery (at the data link layer) to reduce the error rates exhibited by the CDMA channel, with the goal of providing error rates close to wireline links, as seen by transport layers and above. Non-transparent RLP reduces FER for CDMA cellular data through retransmission of lost packets in response to automatic repeat request (ARQ). However, the process of

retransmissions increases the delay, which is undesirable to real-time applications. Therefore, transparent RLP will be the protocol of choice for this research.

To verify network characteristics reported in this chapter and to gain further understanding of network behavior, empirical "real-life" simulation is performed in the next chapter. Simulation studies conducted in chapter 5 and documented network properties and characteristics will be used to identify and develop models for Internet and CDMA networks.

CHAPTER V

EMPIRICAL NETWORK PERFORMANCE STUDY

In addition to simulations using network models for the Internet and cellular data, experiments across the actual networks (Internet and CDMA cellular data) is done to look at "real-life" scenarios using an FNBDT demonstration system [5] and a network performance application specifically designed to collect network statistics. Empirical network statistics are collected between remote sites to gain real data traffic statistics. The measured network statistics will be used to develop better models by modifying parameters based on "real-life" traffic characteristics to focus our simulations, as opposed to ad-hoc variation of parameters. The results of the empirical studies as well as published network statistics are used to verify results of the model simulations.

Empirical Experiment with FNBDT Demonstration System

This experiment uses a previously developed voice-over-IP application that uses the MELP audio codec and includes a limited simulation of FNBDT. The MELP-over-IP (MoIP) application is a multi-threaded MS-Windows based application that uses the OSU implementation of the publicly available MELP algorithm from the Defense Digital Voice Processor Consortium (DDVPC) (http://www.plh.af.mil/ddvpc). The application has demonstrated full-duplex real-time voice communication over office LAN (802.3, Internet), wireless LAN (802.11b), CDMA cellular data (IS-95), and a combination of these networks.

A preliminary experiment was conducted to study the "real-life" characteristics of concatenated Internet and CDMA cellular data networks. The study of the combined

network behavior will be used in the evaluation of FNBDT and MELP operation in packet network environments. The experiments use the MoIP application as a client-server pair between two PC's. The server (transmitter) is on an office LAN connected to the Internet and the client (receiver) is connected to the Internet through a CDMA cellular connection, Figure 5.1. The UDP protocol is used for transport packets across the IP networks, however non-transparent RLP (with retransmissions) was the protocol employed by our CDMA link, disabling this option was not available. For each trace two MELP frames (72 bits each (unpacked)) are sent per packet every 45ms from the server to the client. A sequence number is appended to each packet before transmission. The client application monitors received packet sequence numbers to detect lost and out-of-order packets, and monitors inter-arrival delay between packets. Lost packets are detected through gaps in the sequence numbers of incoming packets. Out-of-order packets are detected as a result of sequence numbers of consecutively received packets being out-of-order. Packet inter-arrival delay is attained by recording the relative time between received packets.
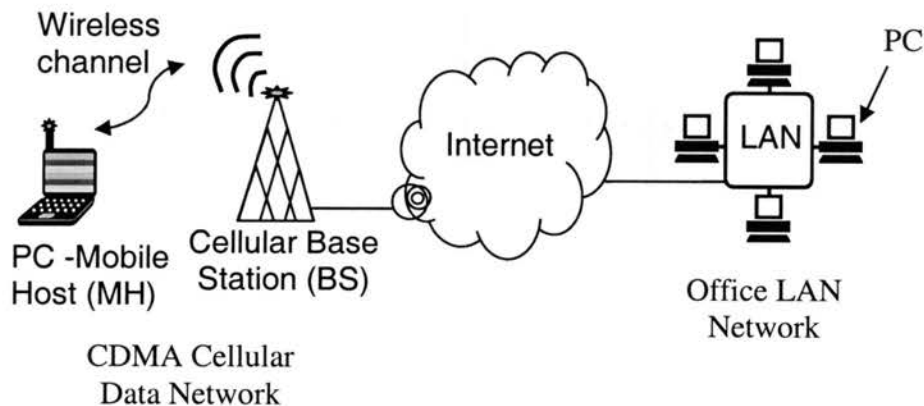
Figure 5.1 Experimental Setup: Office LAN to CDMA Cellular Data.

57

FNBDT Demonstration System Results

The experiment ran traces of packet data across the LAN and cellular data connections. Trace 1 was done on a short-term (small scale), in seconds, where 500 packets were sent over a period of 22.5 seconds. Trace 2 reflects the long-term (large scale) of approximately an hour, where approximately 66,666 packets were sent over a period of 50 minutes. The resulting inter-arrival delay and packet loss were recorded and are presented below.

Figure 5.2 shows the inter-arrival delay of trace 1. It is obvious that the delay is bursty with jitter reaching as high as 1.8 seconds. The mean jitter for this trace is 81.29 ms and the standard deviation was 208.36 ms. Figure 5.3 shows the inter-arrival delay of trace 2. The delay is also bursty with high inter-arrival delay bursts. The mean jitter delay for trace 2 is 48.91 ms with a standard deviation of 59.09 ms. In both traces packets are sometimes delayed in excess of one second, leading to large end-to-end delay and/or dropped packets. Delay is often asymmetric. This appears to be a characteristic of the CDMA cellular segment of the connection. Although overall average delay is reasonable for both traces, agreeing with the finding above, the large burst delays can certainly cause problems in real-time communications.
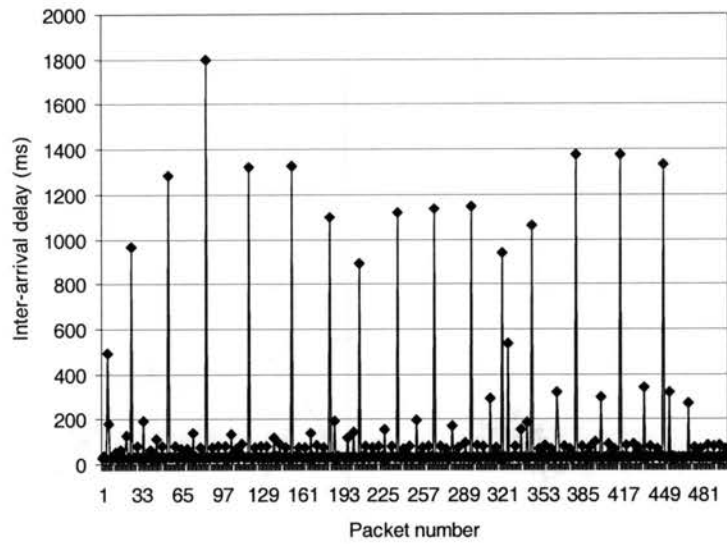
Figure 5.2 Packet Inter-arrival Delay vs. Packet Number for (Short Term) LAN to Cellular Data – Trace1.
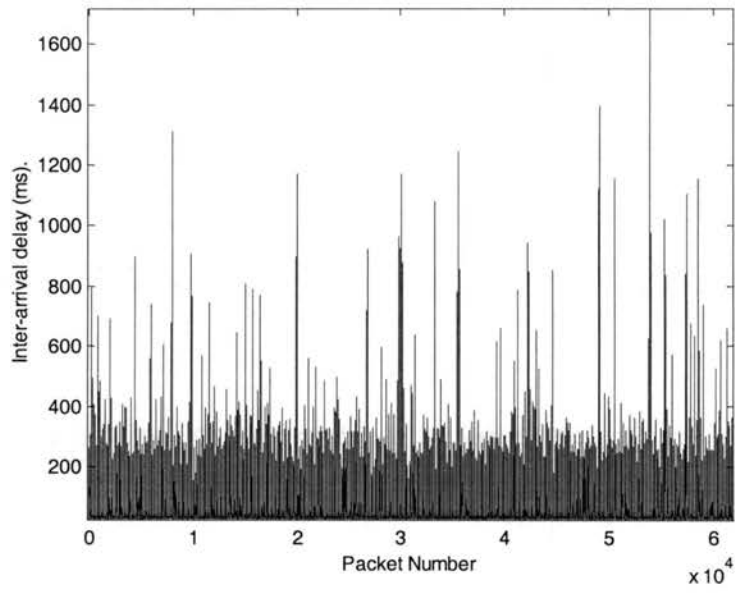
Figure 5.3 Packet Inter-arrival Delay (Long Term) LAN to Cellular Data – Trace 2.

59

Figures 5.2 and 5.3 show that delay is bursty and appears self-similar. This self-similar property not only observed between the two traces but within trace 2, the longer trace, where delay is bursty over all but also in smaller regions on the graph as well.

Figure 5.4 illustrates the frequency distribution of consecutive packet loss in trace 2. This figure agrees with results reported in [83], the slope close to the origin is approximately linear. The graph is plotted on a log scale because the probability distribution decreases geometrically away from the origin. The mean burst length for the trace is 1.076, also in agreement with findings that packet losses of one or two occur most often. Total packet loss for the trace is 5408 (out of approx. 66,666 packets sent), giving an average packet loss rate is 8.11%.
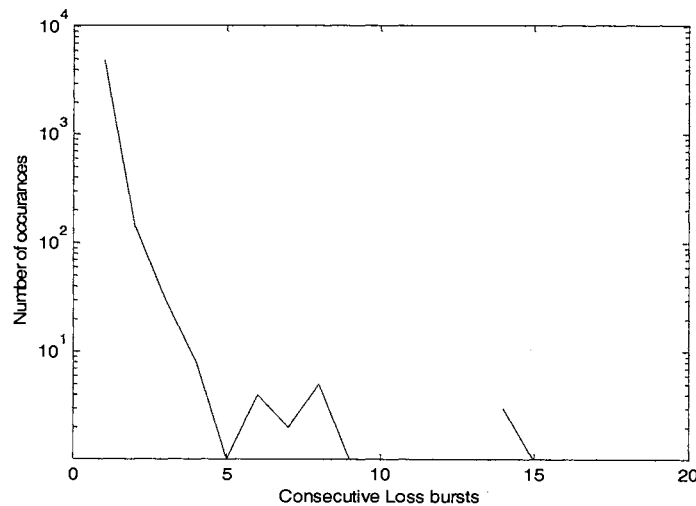
Figure 5.4 Frequency Distribution of Consecutive Packet Loss Bursts for Trace 2.

The overall delay burstiness could potentially be due to Internet properties, but we suspect that the large loss bursts are possibly due to the non-transparent RLP attempting

to recover lost packets through its retransmits or the delay imposed by reordering out-of-order packets. This could also be a reason the overall packet loss is low, and in the case of trace 1 there were no lost packets, with high burst delays. This makes the use of non-transparent RLP unfavorable for real-time voice applications.

There were three cases where packets were received out-of-order. Since RLP reorders out-of-order packet on the CDMA link, this leads us to believe that the packets arrived at the cellular link out-of-order, thus arriving out-of-order as a result of the Internet segment of the connection.

Empirical Experiment with Network Performance Application

Similar experiments were conducted with a Network Performance Application (NetPerf) that was specifically developed to collect packet network statistics. NetPerf is a program written in Visual C++ to operate under MS Windows operating systems. The program runs as a server/client pair over two computers connected to an IP network (possibly concatenated networks). The application was specifically designed to simulate sending FNBDT/MELP formatted data transmission over IP networks, but is flexible enough to accommodate other data sizes and send rates. The user specifies the IP packet parameters to simulate VoIP traffic such as frame sizes, packet size (or frames per packet), send rate, duration of experiment, etc. During an experiment NetPerf appends packet sequence numbers on all outgoing packets, records all packet send and arrival times (keeping track of received packet numbers). This information is used to calculate packet statistical information, such as packet inter-arrival time (jitter), end-to-end packet delay, packet loss rate, etc. Accurate timing is critical for this application; therefore the two computer's (client and server) clocks are synchronized using Network Time Protocol

(NTP) [84]. NetPerf is a multithreaded architecture that used multiple TCP connections and a UDP connection for data transfer.

The results of using NetPerf provided similar results as above. Similar network trends were observed, however, the time statistics were more accurate for NetPerf over our MoIP application because NetPerf uses synchronized system clocks. A trace was run under the same scenario as the MoIP experiment over a LAN (connected to the Internet through OSU's campus network) to CDMA cellular data (Sprint data) connection between 2 PCs. Both computers were synchronized using NTP. Experiment parameters reflected FNBDT/MELP data transfer with 2 MELP frames per packet at 7 bytes per frame at a send rate of approximately 45 frames per second. The experiment was conducted during a busy period of the campus network (12 noon), where network usage is high. A "trace route" was executed using the *tracert* utility to determine the router path and number of hops from the source PC, connected to the Internet via OSU campus LAN, to the destination PC, connected to the Internet via the Sprint CDMA data connection. The trace route command revealed that there were 15 hops from the source to destination, where packets were routed through the Oklahoma State University campus network to Onenet's Internet backbone going through Oklahoma City, Oklahoma to Sprint's U.S. IP backbone network at Fort Worth, Texas, Chicago, Illinois, Roachdale, Indiana and a series of nodes (and obviously cell towers) until reaching the PC connected with the Sprint CDMA data connection. The experiment was run for 1 hour with a total of 69,671 packet sent. The following statistics were recorded and reported: packet inter-arrival jitter, loss rate, end-to-end delay, jitter delay distribution, and loss distribution.

Network Performance Application Experiment Results

The packet inter-arrival jitter for the run is presented in figure 5.5. Again the inter-arrival delay is bursty with a delay spike reaching close to 3 seconds. The mean jitter was 52.35ms with a standard deviation of 77.95ms.
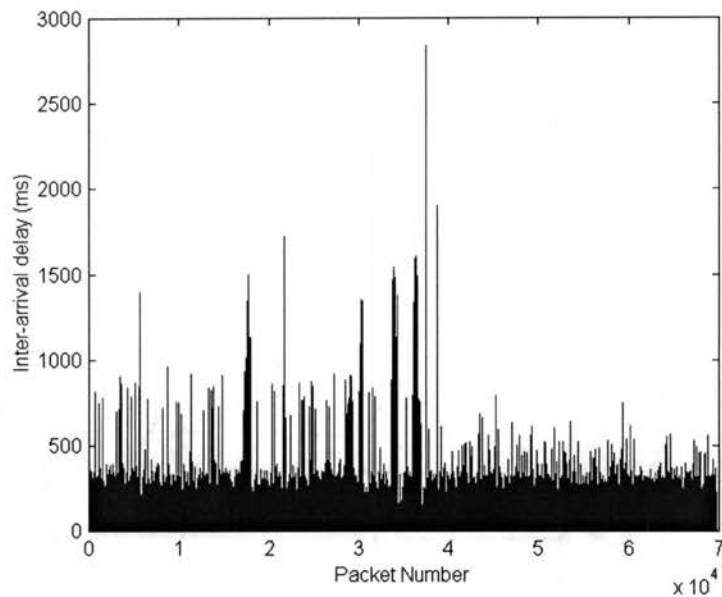
Figure 5.5 Packet Inter-Arrival Delay LAN to Cellular Data Entire Trace –Netperf.
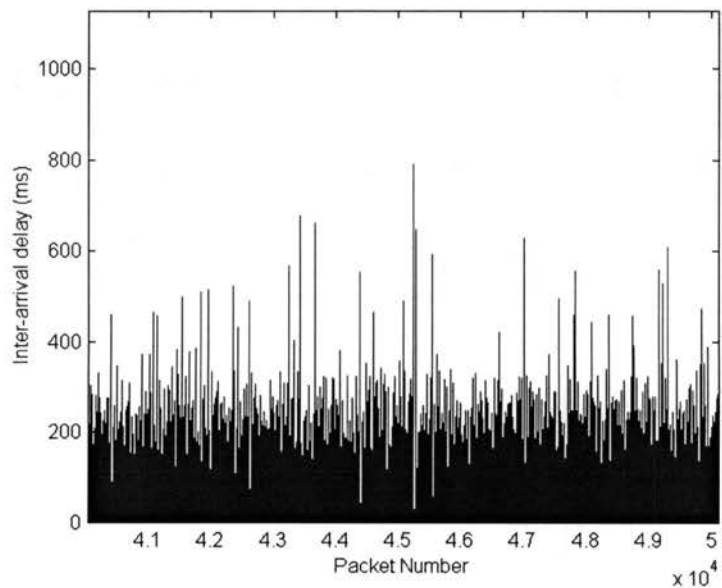
Figure 5.6 Packet Inter-Arrival Delay LAN to Cellular Small Time Scale –Netperf.

The inter-arrival delay also shows signs of the self-similarity property for burstiness.

Figure 5.6 shows a smaller time scale of the trace, graphically showing the bursty and

self-similarity qualities.
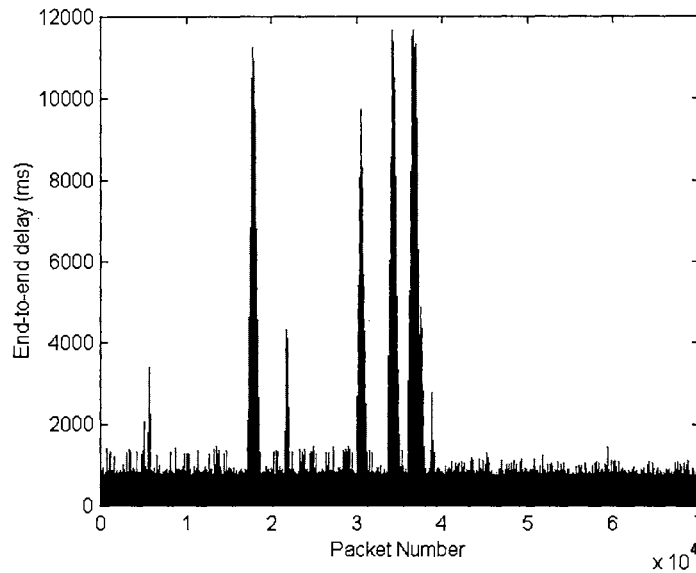
Figure 5.7 End-to-end Packet Delay LAN to Cellular Data – NetPerf.
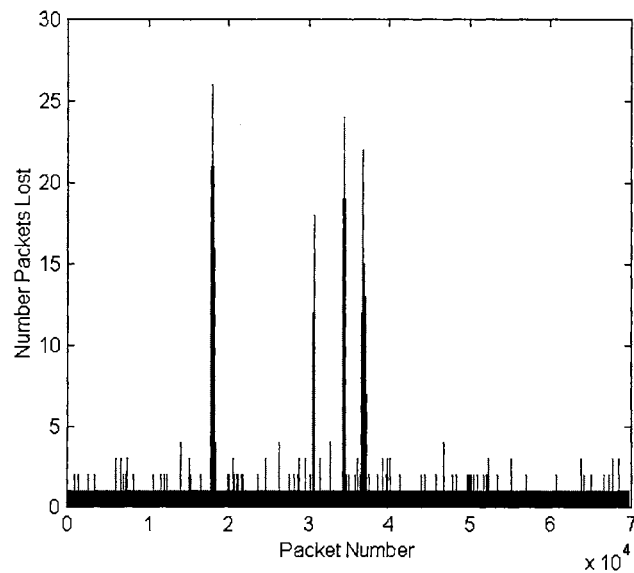
Figure 5.8 Packet Loss vs. Packet Number – NetPerf.

Figure 5.7 and figure 5.8 are included because they give clues to some

irregularities in network behavior during the trace run. The end-to-end delay for the trace

overall is bursty. There are significant delay spikes at packet numbers of approximately

$2x10^4$ and between the range of $3x10^4$ and $4x10^4$. These high delay spikes, on figure 5.7

(with corresponding high inter-arrival delays, figure 5.5), correspond with high loss areas

on figure 5.8, which could be areas of congestion on the Internet (as evident with high

inter-arrival delay spike around these areas on figure 5.6) or deep fades for the wireless

link that lead to loss. This could also be due to non-transparent RLP not being able to

keep up with data following large delay spikes (see discussion below).

Delay Bursts on CDMA Link

The inter-arrival delay jitter pdf, figure 5.9, has a sharp Laplacian like distribution

at approximately 28 ms with an additional peak at 25 ms. This trace is very bursty with

large delay bursts which pulls the mean delay up to 52 ms, even though the majority of

the delays are around 28 ms. Our send rate is at a constant 45 ms, however the majority

of the inter-arrival delays at the receiver is approximately 28 ms according to the jitter

pdf. This is probably due to RLP which transmits a packet every 20ms. Further, the delay

spike phenomenon is apparent in the trace, shown in figure 5.10, which shows a delay

spike followed by several packet jitter delays of around 28ms.
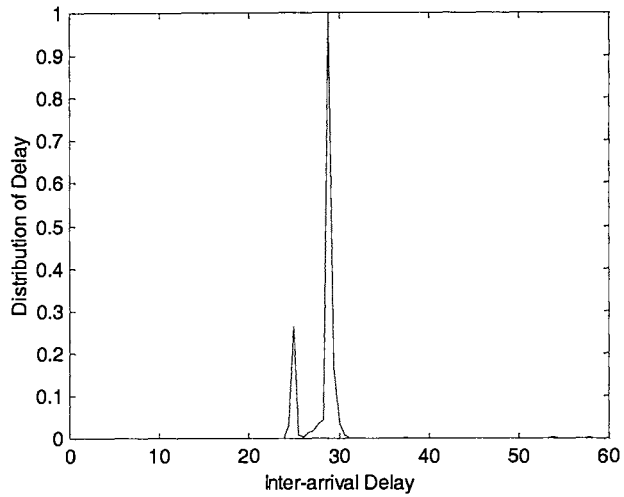
Figure 5.9 End-to-end Packet Delay LAN to Cellular Data – NetPerf.
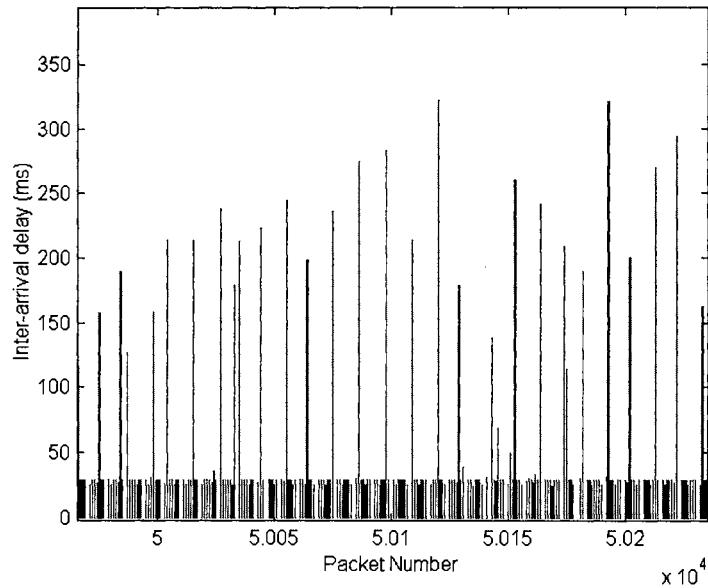
**Figure 5.10 Inter-Arrival Delay LAN to Cellular Data Showing Delay Spikes–
Netperf.**

There are 2 possible causes for the delay spikes followed by several lower inter-

arrival delays including 1) large Internet delays and 2) high packet loss rates at the

CDMA link. First, since RLP transmits frames in 20 ms intervals with limited data

payload size (171 information bits for IS-95), when packets from the Internet experiences large delays and data "builds up" (delay spike), it takes multiple RLP frames to deliver the data. As an illustration of this, using equation (4.6) it takes inter-arrival time $T_{int} = 2$ x 20 ms = 40ms to transport one packet containing 2 MELP frames (336 bits including headers). MELP frames are transmitted every 45 ms and thus reach the CDMA channel every 45 ms + Internet delay. So, if packets are delayed thus producing a delay spike where subsequent packets "catch up", there will be multiple packets at the CDMA (RLP) link. If say, n packets arrive (somewhat) simultaneously, it will take $T_{int}=2n$ x 40ms for RLP to deliver the packets. As an example, if 4 packets arrive at the CDMA link there will be $T_{int}=2(4)$ x 20 ms=160 ms of delay to deliver the packets. In the mean time, there will be an additional 3 to 4 packets arriving while the previous group of packets are being delivered. It will take an additional 120 to 160 ms for RLP to transport those waiting packets, and so on. While RLP is in the process of delivering the "built up" data, taking multiple frames at RLP frame times, incoming data is still building up which again results in RLP taking multiple frame times to deliver the "built up" data. This cycle is reflected in figure 5.10. Consequently, the jitter distribution shows the majority of the packet jitter is close to 20 ms. Also following this cycle of "built up" delays are lost packets. These are shown as gaps in the graphs between delay values. It appears that packets are dropped whenever RLP cannot catch up with delayed packets. This trend will potentially be avoided in future 3G cdma2000, because of the higher data rates supported. Higher data rate translates into more information bits included in an RLP frame, table 4.1. For example, at 1.036M bps 20ms RLP frames can hold 20712 information bits (2.589k bytes). Therefore, if a large delay results in data "build up", RLP may be able to

accommodate its timely delivery due to the larger data payload capacity, effectively

smoothing the delay jitter.

The second potential cause of delay spikes is high packet loss rates on the CDMA

link. When RLP with retransmissions is used the delay goes from T_{ave} to T'_{ave} [58]

[113], where

$$T'_{ave} = D + (k-1)\tau + \frac{k(P_f - (1-p))}{P_f^2} \times (\sum_{j=1}^{n} \sum_{i=1}^{j} P(C_{ij})(2jD + (\frac{j(j+1)}{2} + i)\tau)), \quad (5.1)$$

$n = 3$, the maximum number of RLP retransmissions (retrials) for recovery of lost frames,

k is the number of RLP frames required to transport a transport layer packet and C_{ij} is the

first frame received correctly at the destination, being the ith retransmitted frame at the

jth retransmission trial, so, $P(C_{ij})$ is

$$P(C_{ij}) = p(1-p)^2((2-p)p)^{\frac{j^2-j}{2+j-1}}, \quad (5.2)$$

and P_f is the probability of transmitting a frame successfully over the RLP link, given by

$$P_f = 1 - p + \sum_{j=1}^{n} \sum_{i=1}^{j} P(C_{ij}) = 1 - p(p(2-p))^{\frac{n(n+1)}{2}}, \quad (5.3)$$

where packet loss rate is given by $q = 1 - P_f^K$. From equation (5.1) it is clear as the

packet loss rate approaches zero, the equivalent delay is dependent on the end-to-end

delay, D, and the number for RLP frames required to transport a packet, k, as the 3rd and 4th terms contribute little or nothing to the delay calculation. However, as the packet loss rate increases, delay increases because of retransmissions, as the 3rd and 4th terms contribute more to $T^{\sim}_{ave}$. The retransmissions, and consequently the increase in delay, also lead to the "build-up" of delays as packets continue to arrive at the CDMA link from the Internet link while RLP retransmissions attempt to correct for lost packets. Thus a similar situation arises as illustrated previously. Again, this trend can be avoided with 3G cdma2000 as RLP with retransmissions can be disabled, alleviating the additional delay imposed by retransmissions. However, the overall frame loss rate will increase because of the lack of error correction.
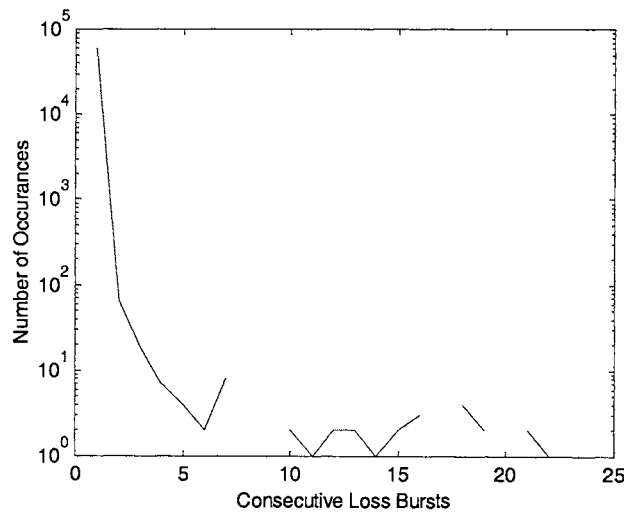
Figure 5.11 Frequency Distribution of Consecutive Packet Loss Bursts for Netperf.

The packet loss rate for the trace was 13.9% with a burst loss rate of 7.23%. The loss distribution in figure 5.11 indicates that packet loss of 1 to 2 packets occurs most often, with burst loss reaching as high as 25 consecutive packets.

The experimental results in many respects are consistent with those presented in the literature. Delay is bursty. The traces experienced large delay bursts approaching 1 second followed by data packets arriving in close proximity to each other (clustered together), which could be a product of RLP, retransmits, or handling of delay spikes in the cellular data segment of the connection. Overall packet loss was low, with loss bursts of one to two packets being most common. Low loss rates with high delay bursts further our suspicion of RLP, which would make it unsuitable for real-time communication. Contrary to what is reported in the literature, there were almost no occurrences of packets delivered out-of-order.

The large delays exhibited by these networks make real-time communication over the concatenation of these networks impractical, especially with non-transparent RLP enabled. Third generation (3G) wireless networks have the promise of providing true packet data service over CDMA networks, where transparent RLP (no retransmits) will be employed. Phase 1 of 3G is on the horizon which should improve performance of the CDMA cellular data networks.

The next chapter present network models and an FNBDT sync loss model based on the behavior of the Internet, CDMA, and FNBDT protocol (chapter 2). These models will be used to simulate and evaluate the performance of FNBDT over concatenated Internet and CDMA networks.

CHAPTER VI

NETWORK MODELS

Network Modeling Assumptions

The emphasis of this study is a PC-to-PC implementation that will transport real-time voice data over packet networks using the FNBDT data format. The particular focus of this research is on packet networks that employ IP. The Wide-Area Internet and CDMA cellular data networks are the networks of interest, as will the concatenation of these networks. The data performance of these networks will be evaluated from end-to-end on the packet level without regard to the physical underlying signal propagation. The underlying network protocols are UDP for the efficient real-time transport of packets, and for CDMA cellular data the transparent RLP protocol will be assumed for the data link layer wireless data transport. The network models presented in this chapter are used for the investigation of the performance of FNBDT over mixed networks.

Packet Network Models

Several packet network models are proposed in the literature, but it is important that the model capture the characteristics of the network being modeled. Two models found in the literature that model packet loss for the Internet and CDMA cellular data as well as a new multi-structure IP packet network delay jitter model are used in this study. The widely used Gilbert model [69] (burst loss model) for the Internet, the correlated fading channel model [66] for CDMA cellular data (IS-95 and IS-2000 cdma2000) radio links, and the multi-structured network delay jitter model, developed based on characteristics reported in the literature and observed from empirical network

71

simulations. The Gilbert model captures the burst loss error characteristics of the Internet using both unconditional and conditional loss probabilities. The correlated fading channel model uses a first-order Markov chain based on Rayleigh distributed fading to model burst loss due to multipath fading in CDMA data channels. The multi-structure network delay jitter model captures the multi-structure characteristic [82] of packet network delay. These models will be utilized to predict the behavior of FNBDT and MELP in a mixed network environment.

Internet Burst Loss Model

An adequate loss model is needed to capture the characteristics of the Internet. It is important to capture not only the rate at which packets are lost, but the burstiness of packet loss. [85] [86] [87] investigated models that capture the loss rate and burstiness of Internet packet loss and recommended a Markov model. They proposed several variations of Markov models, including an extended Gilbert model and an n-state Markov model, that have varying number of states and memory implementations. The consensus is that higher order models yield better accuracy in representing large burst losses, however, for implementation simplicity and low complexity the commonly used two-state Markov model, also known as the Gilbert model, is used for network simulations. Sanneck [88] found that the higher order models did not provide much additional information, particularly since overall packet loss gaps are usually close to one or two packets. Therefore, the two-state Gilbert model is considered sufficient in modeling the temporal loss dependency of the Internet.

We use the probabilistic Gilbert model to simulate packet deletion based on packet loss rates and burst loss. Figure 6.1 illustrates the Gilbert model.
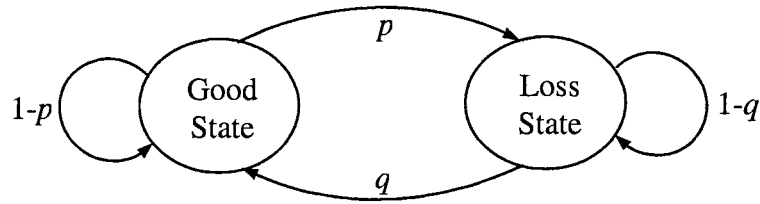
Figure 6.1 The Gilbert Model

The transition probability that a packet is lost, given the previous packet was not lost (*Good State → Loss State*) is denoted in Figure 6.1 as p, and vice-versa for q (probability packet received (not lost), if the previous packet was lost (Loss State → Good State). The conditional loss probability (*clp*), which gives burst loss, is shown in figure 6.1 as 1-q, the probability that the current packet is lost given the previous packet was lost. The probability that n consecutive packets are lost is $p_n = (1-q)^{n-1}q$, which has a geometric distribution. The steady state probabilities for each state are:

$$P_{Good} = \frac{q}{p+q}, \qquad P_{Loss} = \frac{p}{p+q} \qquad (6.1)$$

The steady state loss probability P_{Loss} is referred to as the unconditional loss probability (*ulp*). Note that in a Gilbert model, $p + q < 1$, however when $p + q = 1$ the model becomes the Bernoulli model.

CDMA Fading Channel Model

Wireless channels can be modeled as Markov processes. Markov processes have been widely used in the literature to characterize the bursty error characteristics of wireless channels. This research has extended this to include fading channels. A first-order (two-state) Markov process has been investigated to determine its accuracy on fading channels (at the packet-level) by [62] and [66]. They found that it provides a good model with reasonable accuracy for fading channels. In [66], Zorzi, Rao, and Milstein found that a Markov process is a very good model for a broad range of parameters (block lengths, block error rates, fading margins, etc.).

Wang ([67]) investigated the accuracy of the first-order Markov process over Rayleigh fading channels from the perspective of the data transmission envelope. Zorzi et al. ([62]) extended the work of Wang to include the accuracy of the first-order Markov process for data block transmissions (packet level), with a similar approach used in [67]. Zorzi et al. used a first-order binary (two-state) Markov chain to match the average ON (success) and OFF (failure) periods of packet transmission on a multipath fading channel. This channel model is also used by Chockalingam and Bao [68] to model DS-CDMA forward link from base station-to-mobile link for the evaluation of the performance of the TCP/RLP protocol for IS-95 CDMA. Bai *et. al.* ([89]) used this model to characterize TCP/IP performance over high data rate cdma2000 (IS-2000) ([78] [79]) with transparent and non-transparent RLP, making it a desirable model for this study.

Zorzi *et al.* assumes a flat fading channel and states that the channel can be modeled as a Gilbert channel ([69]), and the pattern of errors follows a first-order

74

Markov model. A first-order Markov model can be described by the transition matrix
$M(x) = M(1)^x$, with

$$M(x) = \begin{pmatrix} p(x) & q(x) \\ r(x) & s(x) \end{pmatrix}, \quad M(1) = \begin{pmatrix} p & q \\ r & s \end{pmatrix}, \tag{6.2}$$

where $p(x) = 1 - q(x)$ (or $r(x) = 1 - s(x)$) is the probability that the ith block

transmission is successful given that the i-x block transmission was successful (or

unsuccessful). Therefore, given the matrix $M(1)$ the channel properties can be completely

characterized. In particular, it is possible to find the steady-state distribution of the chain.

The steady state probability, P_e, that a frame error occurs is

$$P_e = \frac{q}{q + r} \tag{6.3}$$

The transition probability $r = P[\text{success|failure}]$, with $1/r$ representing the average length

of a burst of frame errors, which is described by a geometric random variable. For the

Rayleigh fading channel, the average FER (frame error rate) can be calculated as

$$P_e = 1 - e^{-\frac{1}{F}} \tag{6.4}$$

where F is the fading margin,

$$r = \frac{Q(\theta, \rho\theta) - Q(\rho\theta, \theta)}{e^{1/F} - 1} \tag{6.5}$$

and

$$\theta = \sqrt{\frac{2/F}{1 - \rho^2}} \tag{6.6}$$

with $\rho = J_0(2\pi f_d T)$ as the Gaussian correlation coefficient of two successive samples of the complex amplitude of a fading channel with Doppler frequency f_d, taken T seconds apart. The parameter fdT is the normalized Doppler bandwidth that describes the correlation in the fading process. J_0 is the Bessel function of the first kind and of zero order. $Q(\cdot,\cdot)$ is the Marcum Q function ([70]), given by

$$Q(x, y) = \int_y^\infty e^{-\frac{(x^2+y^2)}{2}} I_0(xw)wdw \tag{6.7}$$

where I_0 is the modified Bessel function of the first kind and of zero order. P_e (6.4) and $1/r$ (6.5) are independent parameters from which the Markov parameter p can be calculated using (6.4). For this proposed CDMA implementation the Markov model will be applied at the RLP frame level where the interframe time is T=20ms. Chockalingam and Bao ([68]) state that by choosing different P_e and fdT values, fading channel models can be established with different degrees of correlation in the fading process.

A New Multi-Structure Network Delay Jitter Model

A network delay jitter model must capture the multi-structure, small and large time scales, nature of delay [78]. Li and Mills' ([78]) study and analysis of network delay states that smaller sampling intervals can be modeled as a non-stationary process, while the longer time intervals can be modeled as a stationary process. The point at which the delay changes from its small time scale characteristic to its large time scale characteristic is referred to as the crossover [90].

Several researchers have developed complex queuing models and performed empirical simulation studies of network jitter including [80] [91] [81]. These studies show that network jitter follows a Laplacian distribution. Yletyinen and Kantola ([91])

showed that the probability distribution function widens and narrows based on increases in traffic load, utilization, burstiness, and burst length. Widening of the Laplacian pdf signifies higher probabilities of long jitter delays, while narrowing of the pdf implies higher probability for small jitter delays.

While investigating an adaptive jitter buffering algorithm, Ramjee *et al.* [92] reported the delay spike phenomena, a condition where a delay spike in a packet stream is followed by a series of packets arriving very close together. When a packet experiences a large delay, the packets following get "held up" and as a result they all arrive with very small inter-arrival times, sometimes almost simultaneously. Ramjee *et al.* found that the delay spike phenomena decayed exponentially, and incorporated an exponential decaying function into the spike detection jitter buffering algorithm. When their jitter buffering algorithm detected a delay spike it was allowed to decay exponentially to follow the slope of the spike until it leveled off at approximately zero slope, signaling the end of the delay spike. This obviously indicates a correlation attribute of inter-arrival jitter delays.

The network delay jitter characteristics described above are combined in the design of a new multi-structured delay jitter model, developed and presented in this dissertation for the performance evaluation of FNBDT in packet networks. The multi-structured network delay jitter model uses a Laplacian distribution to model the small time scale, non-stationary, process, while the large time scale stationary process is modeled with additive Gaussian white noise. An exponential decaying function is added for modeling the delay spike phenomena providing correlation between jitter delay values. When the Laplacian or Gaussian distribution produces a delay spike, subsequent jitter delays are allowed to decay exponentially. The combination of the Laplacian

distribution, Gaussian white noise, and delay spike adjustment provides a means to model the multi-structure characteristics of packet delay. The new multi-structure network delay jitter model is shown in figure 6.2.

```
┌─────────────┐      ┌─────────────┐      ┌─────────────┐
│  Laplacian  │      │  Additive   │      │ Exponential │
│ Distributed │─────▶│  Gaussian   │─────▶│  Decaying   │
│   Delays    │      │ White Noise │      │ Delay Spike │
│             │      │             │      │ Adjustment  │
└─────────────┘      └─────────────┘      └─────────────┘
 Small time          Large time          Delays following
 scale delays        scale delays          delay spikes
```
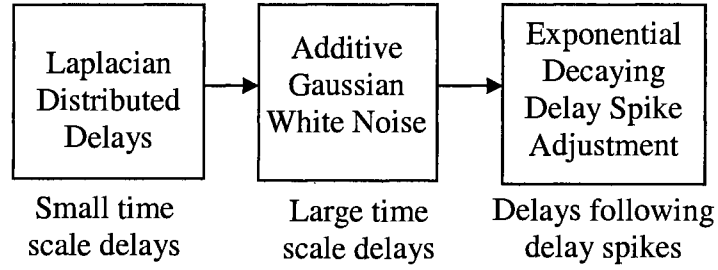
Figure 6.2 The New Multi-Structured Network Delay Jitter Model

The Laplacian distribution is used to generate packet jitter delay values based on a specified mean and variance. The characteristic equations (6.8) and (6.9) for the Laplacian distribution function are given as follows:

$$F(X) = \frac{1}{2} e^{-\frac{|X-\alpha|}{\beta}} \qquad , \quad x \leq \alpha \qquad (6.8)$$

$$F(X) = 1 - \frac{1}{2} e^{-\frac{|X-\alpha|}{\beta}} \qquad , \quad x > \alpha \qquad (6.9)$$

where,

$$\alpha = mean \text{ and } \beta = \sqrt{\frac{var}{2}} \ .$$
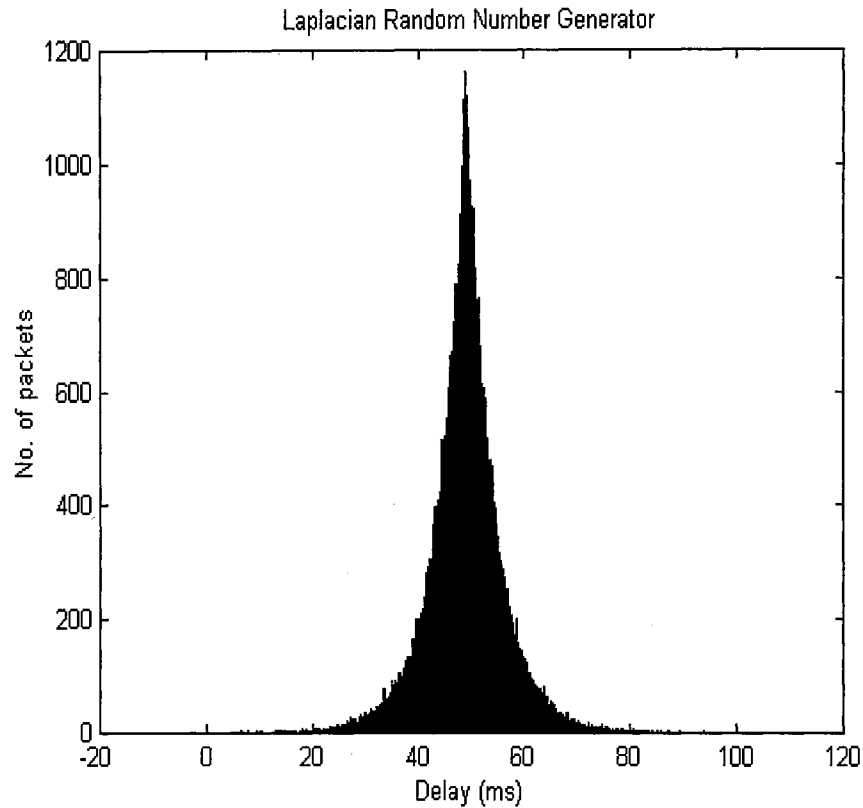
Figure 6.3 Generated Laplacian Distribution using Laplacian Random Number Generator.

Figure 6.3 shows the Laplacian distribution function implementation simulating 10,000 packets with $\alpha = 49$ and $\beta = 5$. Gaussian white noise is added to the delay values generated by the Laplacian distribution function to introduce the large time scale properties. This is done based on a sampling that follows the crossover value. For example, delay values are altered by the Gaussian white noise process at intervals based on the crossover from small time scale to large time scale. This process models sampling of the busy periods of the queue.

When a delay spike is detected, based on a threshold, delay values following the delay spike are reduced exponentially, allowing them to "catch up" with the delayed

(spike) packet. Equation (6.10) gives the exponential decay function used to model this characteristic. The number of succeeding packets that are affected by the delay spike is denoted as the *spike interval*. In order for the packets following the spike to "catch up" with the delay spike the average jitter delays within a spike interval should be less than or equal to the mean.

$$\tilde{d}(t) = d(t)e^{-xt}, \tag{6.10}$$

where, *d(t)* is the inter-arrival jitter at time *t*. Figure 6.4 provides a pictorial description of this delay jitter adjustment process.
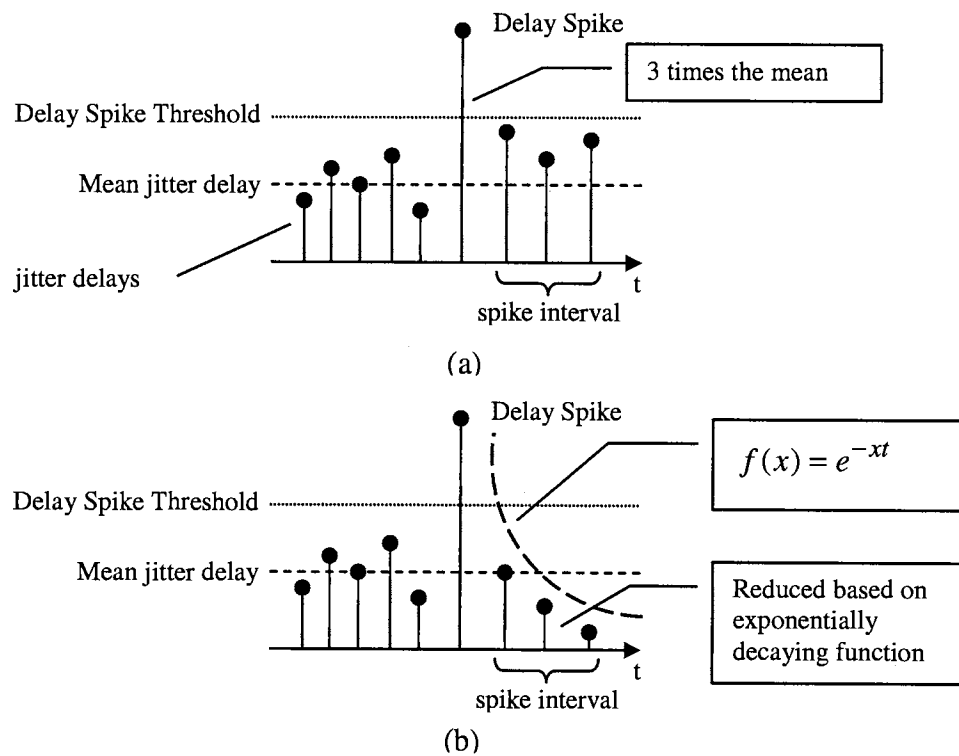
Figure 6.4 Jitter Delay Spike Adjustment Procedure. (a) Delay Spike Produced from Model and (b) Reduction of Delays Following Delay Spike.

Jitter Model Comparisons

To verify the multi-structured network delay jitter model, experiments were conducted with the network performance application, *NetPerf*, to collect network inter-arrival delays between a PC connected to the Wide Area Internet via an office LAN (Local Area Network) and a remote PC connected to the Internet. The packet trace (stream of voice data) sent simulated voice data every 45ms, at 14 bytes of payload per packet, from server to client. The resulting inter-arrival delay jitter is compared with the data simulated from the multi-structured network delay jitter model. The model was used to generate jitter delay values based on the statistical data reported by the *NetPerf* simulation. The jitter data collected empirically using *NetPerf* had a mean of 45 ms and a standard deviation of 5.5. From the delay burst characteristics of the collected data, an estimate for the model crossover value and spikescale threshold is 3 seconds and 100 ms, respectfully, for Gaussian noise sampling and spike detection adjustment. These values were used as the jitter model inputs. Figure 6.5 compares the collected jitter delays from *NetPerf* and the jitter delays produced by the new multi-structure network delay jitter model. The model returned a mean jitter of 45 ms with a standard deviation of 5, values equaling the *NetPerf* data.
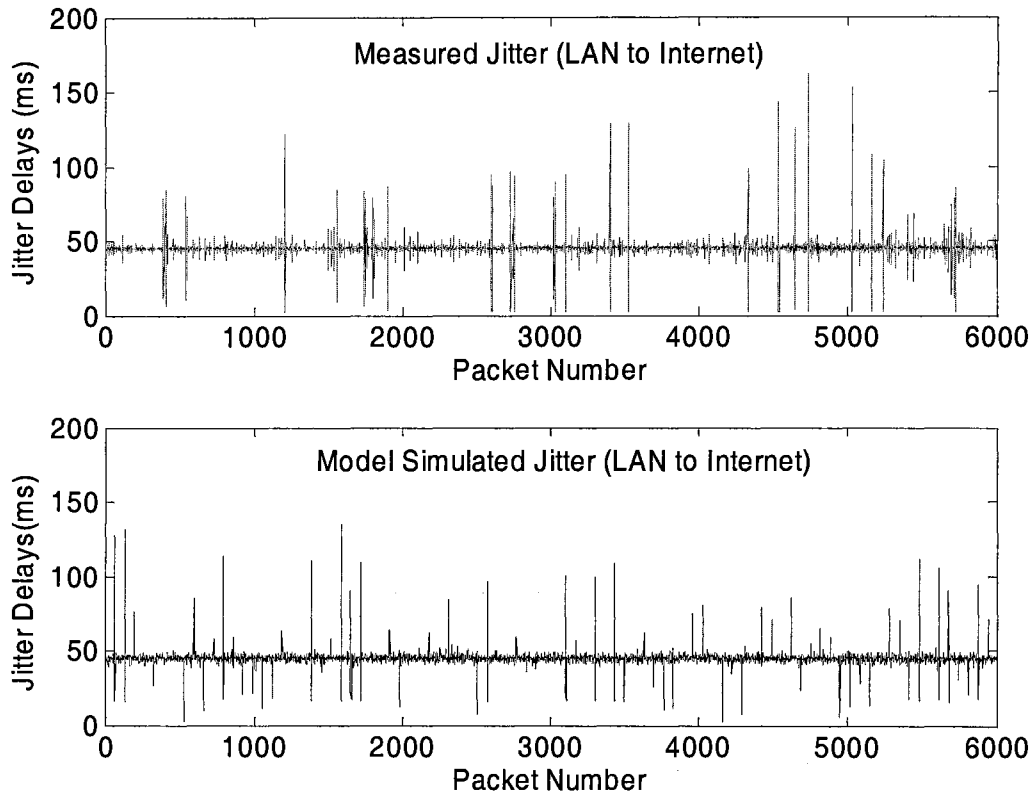
Figure 6.5 Measured vs. Model Jitter Delays

Figure 6.5 shows that both the measured and modeled jitter are bursty, with the majority of the jitter delays appearing near the mean as expected. This is also evident in the inter-arrival delay jitter distribution shown in figure 6.6. The delay distribution for both the measured and modeled delay follows a Laplacian distribution about the mean of 45ms.
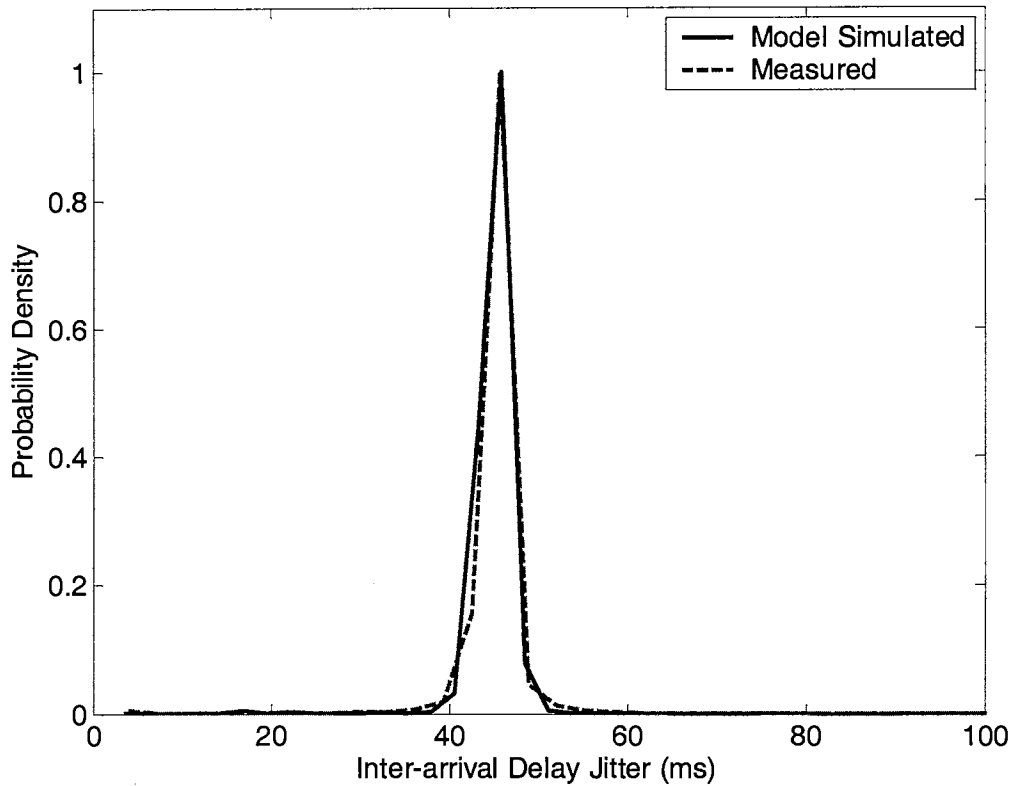
Figure 6.6 Jitter Delay Distributions for Measured and Modeled Jitter Delays.

The data generated from the above simulation and multi-structured network delay

jitter model was compared with trace data runs over LAN (connected to the Wide Area

Internet) to a CDMA data network connection. Trace runs for several times of day were

compared for the LAN to CDMA data trace to get a representation of different levels of

busy periods. Normalized variance characteristics are compared by averaging variance

values of m-aggregated time series. This technique is typically used for self-similarity

determination, as illustrated in [114]. The m-aggregated time series $x^{(m)} = \{x_k^m, k = 0,1,2,...\}$,

of a stationary time series x, can be defined by summing the original time series over

non-overlapping, adjacent blocks of size m. The aggregated variance calculations

essentially compress the time scales, with $m=1$ representing the highest magnification

and $m=10$ a factor of 10 reduction in magnification. This can be expressed as,

$$x_k^{(m)} = \frac{1}{m} \sum_{i=km-(m-1)}^{km} x_i \quad .$$ (6.11)

In this comparison, an m-aggregate time series of the variance of the jitter delays

is used, illustrated in equation (6.12)

$$Var(x_k^{(m)}) = \frac{Var(x_{mk-(m-1)}) + Var(x_{mk-(m-2)}) + \cdots + Var(x_{mk-(m-m)})}{m}, \quad (6.12)$$
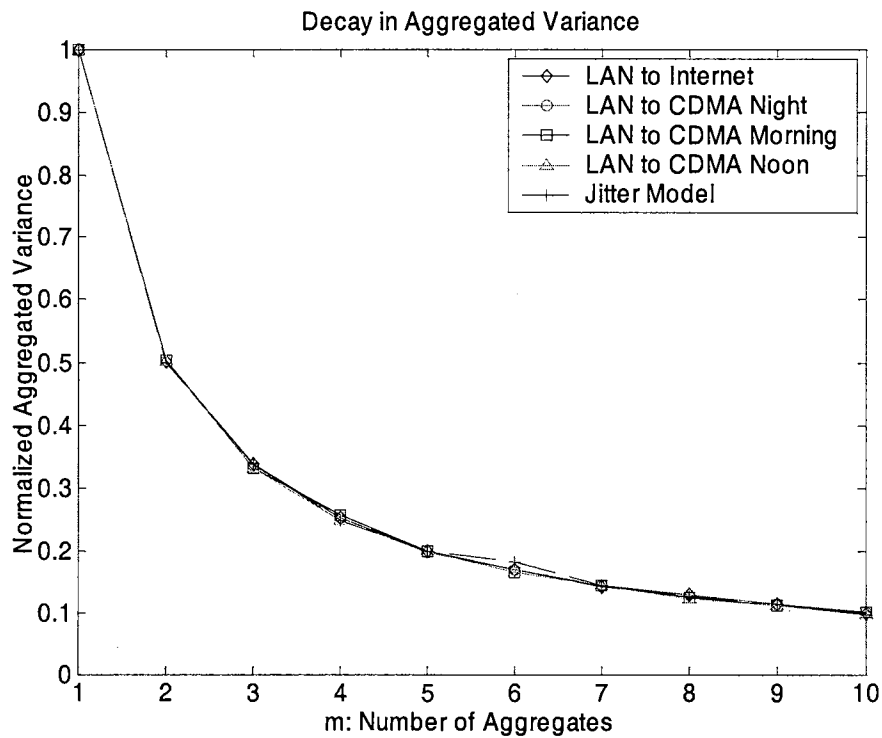
Figure 6.7 Aggregated Variance Comparisons.

84

Figure 6.7 shows the results of the aggregate variance comparisons. For self-similar traffic the aggregate variance would decay slowly over the different time scales, maintaining a somewhat constant normalized aggregated variance value [49]. This would indicate that the variance statistics for the data series would be preserved over the various time scales and thus would describe a self-similar process. However, the simulated model and empirically measured data each decay rapidly. In comparison, the overall variance statistics of the modeled data and the collected data appear to have "exponential like" decay consistent with one another. It can be concluded from the above analysis that the jitter model data follows the short and long term bursty trends of the jitter delay collected from actual packet data networks.

The results of the new multi-structured network delay jitter model will be used to simulate (model) network jitter characteristics. This model will be used the study the effects of network jitter delay on the operation of FNBDT in packet network environments, and its affects on speech quality. An adaptive jitter buffering scheme will be studied and evaluated to improve loss performance in the face of network jitter delay, while minimizing overall system delay.

FNBDT Synchronization Loss Model

An application using FNBDT signaling is not the most typical VoIP application. In a VoIP application, a lost packet results in an equivalent amount of lost speech. FNBDT adds another "layer" to the application due to its framing requirements and security. Each FNBDT superframe incorporates 23 or 24 MELP frames, with a sync management frame for superframe cryptographic synchronization. When sent across a

packet network, an FNBDT superframe is probably fragmented by the underlying VoIP application into several packets to minimize framing delay.

The fragmented FNBDT superframe packets can tolerate some packet loss because of the FNBDT bit-wise encryption scheme. If the amount of data loss (number of packets lost) is not known or is excessive, FNBDT's performance will degrade significantly, because once lost at least one (contained in one superframe) and perhaps three sync-management frames (contained in three superframes) must be received before cryptographic synchronization can be regained. However, for small sequential packet loss of one to two packets, as long as the number of packets lost is known, the remainder of the packets can be decrypted given the index of the encryption key (based on the location of the packet lost in the superframe). Nevertheless, any packet loss results in lost MELP voice frames; therefore the loss is propagated up to the MELP codec algorithm.

The FNBDT Sync Loss Model is a probabilistic model that provides a means to study, evaluate, improve, as well as make suggestions concerning FNBDT's behavior in the presence of packet loss, particularly as it relates to loss of synchronization. The loss of a sync management frame (containing encryption counter information) prevents the decryption of subsequent data, resulting in data loss equivalent to one superframe (approximately 0.5 seconds of speech) or up to 3 superframes (approximately 1.5 seconds of speech). This loss of synchronization will have a dramatic impact on the quality of the communication system.
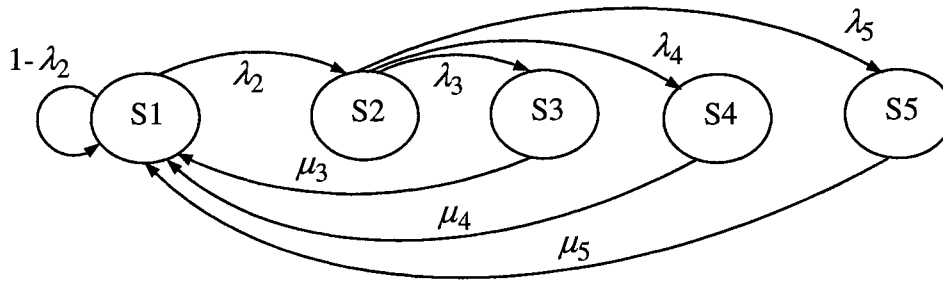
Figure 6.8 FNBDT Sync Loss Model

The FNBDT sync loss model is illustrated in figure 6.8. In a network model

simulation packets that are lost by the proposed network models are propagated to the

FNBDT sync loss model which determines if the loss results in a sync management frame

loss (state S2) or a MELP data loss (state S1). If the loss translates to a sync management

frame loss (from state S1 to S2) then the model determines if it is sync management

frame 1, 2, or 3 (states S3, S4, or S5 respectively). A sync management frame loss results

in at least one superframe worth of MELP data loss, which makes this condition

important to study.

Sync management frames contain the encryption counter information. Due to

bandwidth constraints defined in the FNBDT signaling plan, the counter value is spread

across 3 sync management frames with the start of the counter (partially) being placed in

the first sync management frame (SMF1). The second sync management frame (SMF2)

contains partial counter values that allow the counter to pick up where the last counter

left off to continue to decrypt the next superframe data. Finally the third sync

management frame (SMF3) contains values of the counter to continue from the last superframe (SMF2) and decrypt the final superframe in the three superframe set. Then a new counter value is provided in the next SMF1, and the process continues. If the first sync management frame, SMF1, is lost FNBDT must wait for three superframes until the next SMF1 is received with a new counter value, thus achieving encryption synchronization. If SMF1 is received and SMF2 is lost then encryption synchronization is achieved when the next SMF1 is received, two superframes later. Likewise if SMF3 is lost then encryption synchronization is regained at the next superframe, SMF1. The loss of a sync management frame means that the corresponding superframe data can not be decrypted and is considered lost (unusable by the application).

The transitional probability matrix for the FNBDT synchronization loss model is provided below.

$$
P_{sync} = \begin{bmatrix} 1-\lambda_2 & \lambda_2 & 0 & 0 & 0 \\ 0 & 0 & \lambda_3 & \lambda_4 & \lambda_5 \\ \mu_3 & 0 & 0 & 0 & 0 \\ \mu_4 & 0 & 0 & 0 & 0 \\ \mu_5 & 0 & 0 & 0 & 0 \end{bmatrix}, \tag{6.13}
$$

The rate of departure for state 1 or probability that a lost frame is a sync management frame, given the sample space is a superframe, is $\lambda_2 = p_{SMF}$. If a frame is lost by the network there is a 1 in 24 (or 25) probability the lost frame will be a sync management frame. Therefore, $p_{SMF} = 1/N$, where N is the number of frames in a superframe. Since synchronization information, counter values, are spread across 3 consecutive sync management frames in this model, if it is determined that a sync management frame is

lost, there is a 1 in 3 chance that SMF1($\lambda_3 = p_{SMF1}$), SMF2 ($\lambda_4 = p_{SMF2}$), or SMF3 ($\lambda_5 = p_{SMF3}$) will be the lost frame ($p_{SMF1} = p_{SMF2} = p_{SM3} = 1/3$). The probability of a MELP frame loss is 1-λ_2 = 1-p_{SMF}, and $\mu_3 = \mu_4 = \mu_5 = 1$.

Model Concatenation

The goal of this research is to get a better understanding of the type of errors and network characteristics that the concatenation of Internet and CDMA networks will yield, specifically how these errors affect the QoS of the FNBDT voice communication system. Therefore, these network models will be concatenated to model the mixed network environment for the Internet and CDMA cellular data. The relevant information for the FNBDT analysis can be gained by investigating the parameter space of the concatenation of these network models. By manipulating the relevant parameters for the network models, variations in network conditions, behavior, etc. can be introduced to study the effects of different scenarios on FNBDT and MELP.

The unconditional and conditional loss rates of the Gilbert model are used to simulate the loss and burst loss rates of the Internet. The average burst length, given by $(1-q)^{-1}$, may be used to vary the burst loss length, based on the properties of the Internet. For the CDMA channel model, Chockalingam and Bao ([68]) provide results of average length of frame error bursts, 1/r, as a function of the normalized Doppler bandwidth fdT and the link-fading margin F. They show that when fdT is small, the fading process is very correlated (long bursts of frame errors), however for larger values of fdT successive samples of the channel are almost independent (short bursts of frame errors). The mutli-structured network delay jitter model has the mean, variance, and crossover value to adjust the small scale burst characteristics and the long-range dependent properties of the

network delay jitter. Increasing the variance introduces larger delay spikes, broadening

the delay distribution. Decreasing the variance introduces smaller delay spikes, narrowing

the distribution. The crossover value may be adjusted to vary the long-range dependent

burstiness.

The FNBDT sync loss model will be used in conjunction with the models for the

Internet loss, CDMA fading channel loss, and multi-structured network delay jitter model

to evaluate FNBDT's performance and to provide insight or suggestions for development

of a robust FNBDT voice over packet network application. Figure 6.9 shows the overall

concatenation of network models used in conjunction with the FNBDT sync loss model.

The network models will be viewed as a black box with various parameters being

used to vary the response of the black box (models). The input of the black box will be

the input data traffic that follows the FNBDT / MELP format. The corresponding output

leaving the black box will be the input data traffic as a function of the responses of the

models – possibly being excessively delayed or even lost. For voice quality simulations

the input speech file will be encoded by the MELP codec and the resulting frames fed

into the network and FNBDT model the output of the concatenated models will be

decoded by the MELP codec and the resulting speech quality will be assessed. The

quality assessment will be performed with the quality measures described in the next

chapter.
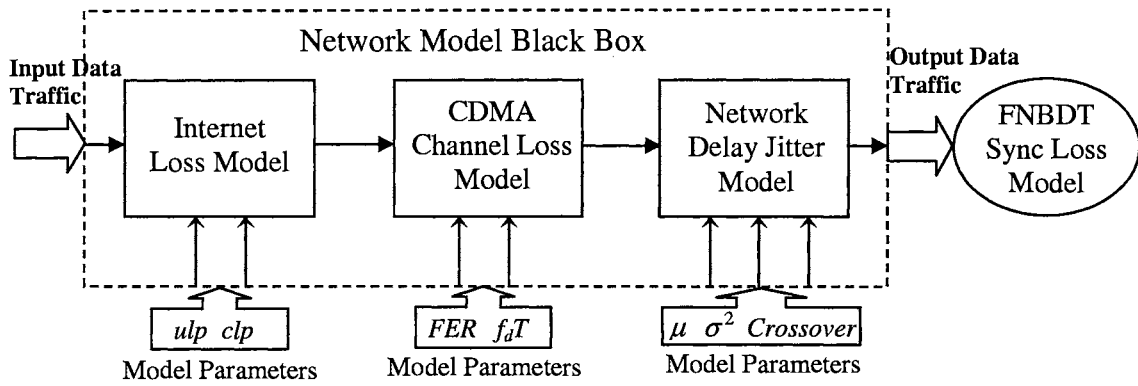
Figure 6.9 Concatenated Network Models including FNBDT Sync Loss Model.

CHAPTER VII

QUALITY MEASURES

System Quality

The quality of a voice communication system lies within the audio quality

perceived by the end user. However, Quality of Service (QoS), which encompasses voice

quality, of a voice-over-packet network system is dependent on the characteristics of the

communications network (or networks) and how the end-to-end system responds to the

network characteristics, and the resulting output audio quality of the system.

Specifically, a system's QoS is typically determined by its delay characteristics, the

ability to maintain uninterrupted speech output, and the level of audio performance in the

presence of packet loss. Thus, a high quality system is a system that maintains continuous

speech output, with low delay, and produces high quality audio output. An FNBDT (and

VoIP) system's quality is a direct result of the performance of the MELP voice codec,

particularly, the naturalness and intelligibility of the synthesized speech in the presence

of packet loss, as well as its ability to compensate for high network delays, including end-

to-end and inter-arrival jitter delays.

To measure the output audio quality of an FNBDT communications system

implementation over packet networks the following quality measures are employed; the

frequency weighted spectral distortion measure, the perceptual evaluation of speech

quality (PESQ) measure, informal listening tests, and a self-organizing map (SOM). The

frequency weighted spectral distortion measure was specifically developed for the

evaluation of the MELP codec and produces a spectral distortion value in dB. The PESQ

measure was developed under the International Telecommunications Union (ITU-T) for the end-to-end evaluation, including delay, of voice system quality, it produces a quality score similar to the mean opinion score (MOS) for speech quality. Informal listening tests using untrained listeners provide subjective audio quality assessment to validate the objective measures. Finally, a QoS map using a self-organizing neural network or Self-Organizing Map (SOM) enable the presentation, categorization, and evaluation of the multi-dimensional QoS related factors, in this case network model parameters and quality score. The use of the frequency weighted spectral distortion measure, PESQ, informal listening tests, and QoS mapping quality measures provide a means of evaluating the overall FNBDT system QoS.

Frequency Weighted Spectral Distortion Measure

Vocoders such as MELP estimate a set of parameters based on a vocal tract model. Unlike waveform coders, vocoders do not attempt to preserve the speech waveform, but attempt to preserve the perceptual qualities of the speech signal through modeling and parameter estimation techniques. Once the parameters are estimated, the original speech waveform (signal) is discarded and a synthetic speech signal is generated using the estimated parameters. The fidelity of a vocoder is based on observed naturalness and intelligibility – both subjective criteria that are in stark contrast to Signal-to-Noise Ratio (SNR). To predict the subjective performance of MELP the frequency-weighted spectral distortion measure was developed [9]. This performance measure has been accepted as a quantitative measure of MELP quality. This measure was originally developed to compare the effects of quantization coding of MELP frames. In this application the measure is adapted to evaluate the quality of MELP with simulated frame

errors from an IP network. Although this measure is reported to yield adequate results for perceptual quality, listening tests are also recommended for the overall evaluation of MELP synthesized speech.

The frequency weighted spectral distortion (SD$_{fw}$) measure is evaluated on a Bark scale that is based on equal critical bandwidths. The SD$_{fw}$ is calculated using a linear frequency scale with an amplitude weighting given by the Bark scale, in order to reduce complexity and to avoid resampling of the frequency axis. To attain sufficient perceptual weighting the amplitude is squared, computed from a 256-point FFT. The SD$_{fw}$ is given in the following equations.

$$SD_{fw}(A_q(z), A(z)) = \sqrt{\frac{1}{W_0} \sum_{f=0}^{4000} |W_B(f)|^2 \left| 10 \log_{10} \frac{|A_q(f)|^2}{|A(f)|^2} \right|^2}, \tag{7.1}$$

where $A_q(z)$ and $A(z)$ represent the quantized and unquantized LPC spectrum. In our case they will represent the LPC spectrum of the lost frame (based on the frame replacement technique used) and non lost frame. W_0 is the summation of the Bark weighting factors. The Bark weighting is defined by

$$W_B(f) = \frac{1}{25 + 75\left(1 + 1.4\left(\frac{f}{1000}\right)^2\right)^{0.69}} \tag{7.2}$$

The spectral distortion is computed over all frames and an average value is calculated. Paliwal and Atal, [94], describe transparent distortion as having an average SD of about 1dB, with less than 2% of outlier frames having SD in the range of 2-4 dB. Having too many outlier frames with high spectral distortion (>4 dB) can cause audible distortion in a speech segment even if the average SD_{fw} is approximately 1 dB.

Perceptual Evaluation of System Quality (PESQ) Measure

To provide a better evaluation of FNBDT and MELP performance over packet networks the Perceptual Evaluation of Speech Quality (PSEQ) [111] adapted by the ITU-T for narrow-band speech quality is employed. PESQ is an objective measure applicable to speech codecs as well as end-to-end quality. It considers analog input filtering, variable delays, coding distortions, and channel errors. It produces an objective PESQ score as well as a prediction of the subjective Mean Opinion Score (MOS), which is traditionally obtained by subjective listening tests.

PESQ uses a perceptual model to compare the original signal and the degraded output signal of the system under test to measure one-way quality. PESQ can be implemented as an intrusive test that measures one-way system quality by sending a test signal through the system, however our implementation is non-intrusive which compares a reference signal with a system output degraded signal. The model structure for PESQ is presented in figure 7.1.
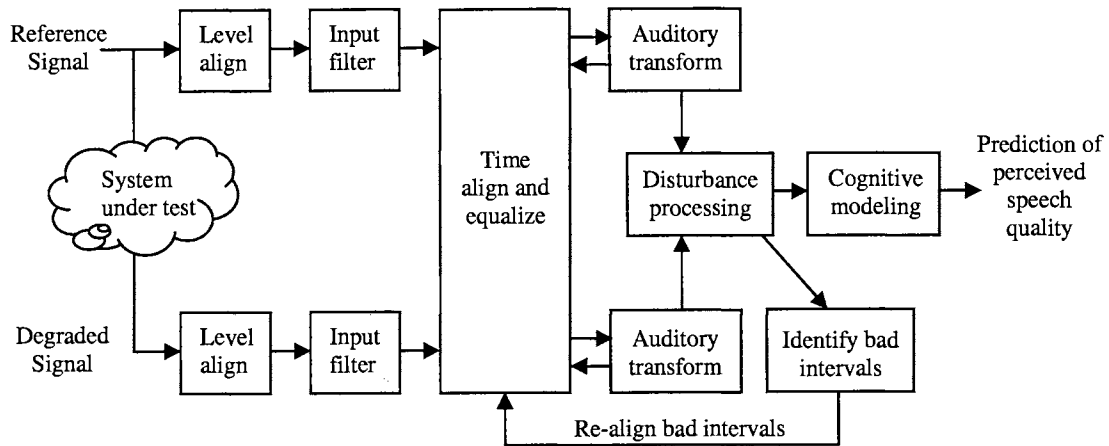
Figure 7.1 PESQ Model Structure [112].

The original (reference) signal and the degraded signal are first level aligned to a normal

listening level used in subjective tests. Then the two signals are filtered with an input

filter that models telephone headsets. To compensate for variations in delay (jitter)

imposed on the signal by the network, the signals are aligned with each other in time

using the time alignment model. The delay results from the time alignment are used in the

next stage, the auditory transform, which can compensate for delay changes in both

silence and active speech. The audio transform algorithm transforms the reference and

degraded signals based on the human auditory system. Then a perceptual model is used to

compare the reference and time aligned distorted signals. The results of the comparison,

resulting in the disturbance measure, give an indication of the absolute audible error and

audible errors significantly louder than the reference. These disturbance parameters are

then converted to a PESQ score and mapped to a predicted MOS score. The PESQ score

ranges from -1 to 4.5 and the MOS-like scores are between 1 and 4.5. Very good quality

corresponds to PESQ scores close to 4.5; on the other extreme very bad quality is around

-0.5. At scores of 2 and below the degradation level make speech difficult to understand.

A reference to MOS scores vs. quality level reported by ITU-T P.800 is provided in

Table 7.1 [115].

Table 7.1 MOS Score vs. Speech Quality

Speech Quality	MOS
Excellent	5
Good	4
Fair	3
Poor	2
Bad	1

Informal Listening Tests

Informal listening tests are warranted as a subjective speech quality measure to

complement the objective audio quality tests. Listeners provide a perceived audio quality

measure that is sometimes immeasurable by the objective measures. Several listeners will

listen to selected test files including original and degraded speech signals. Listeners are

given two types of tests, an acceptability test and a comparison test. For acceptability

tests, listeners are asked if the quality of the sound file is acceptable, unacceptable, or no

opinion. The comparison tests involve listeners comparing two files and selecting the file

they perceived as having "better" quality, or selecting "same" if sound files were close to

one another in quality.

QoS Mapping with Self-Organizing Neural Network

The aforementioned quality measures provide a means to measure codec speech

quality in the presence of packet loss, end-to-end system voice quality with the provision

for inclusion of delay and packet loss, and subjective human voice quality measurement. These quality measures alone cannot show relative relationships across and amongst quality measurements. Likewise, it is difficult to effectively show the multi-lateral aspects of QoS evaluations, given a number of QoS related parameters (i.e. delay, loss, buffering, PESQ score, etc.). However, a self-organizing map neural network can evaluate the overall FNBDT system QoS that takes into account the multiple input model parameters and their corresponding measured quality output.

The QoS mapping of VoIP communications is introduced by Masugi [116], that makes use of the SOM based mapping model developed by Kohonen [117], to provide objective evaluations of VoIP systems' QoS in real environments, taking into account several QoS-related factors. An SOM can show relative correlation between output QoS measures that are not achievable by individual quality measures. It provides a means of displaying relationships of multi-dimensional QoS parameters in a two-dimensional space, converting the non-linear statistical relationships in multi-dimensional data without altering the metric and topological relationships between data [116]. An SOM can also graphically show the multi-lateral variations in parameters (network characteristics) and their effects on system quality. Finally, it can show the total system QoS by tying together all the QoS related parameters including network characteristics, system response to network characteristics, and the resulting audio quality.

QoS mapping using a SOM is used in this research to illustrate overall QoS for raw and enhanced FNBDT/VoIP system performance in error prone packet network environments. As demonstrated in [116], QoS mapping is applicable for use with VoIP (and thus FNBDT) systems in real environments to provide system performance in terms

of QoS level. One particular application of a QoS mapping implementation is to measure and provide real-time QoS information (or current QoS level) to a receiver system. This information can be then used by the receiver system to dynamically adapt to network characteristics. For example, if the measured packet loss rate and delay are high, and consequently results in low QoS, the receiver system can initiate packet loss recovery as well as increase its playout buffers to accommodate the high delays to increase the overall QoS level.

Self-Organizing Map (SOM)

SOM, also known as SOFM (Self-Organizing Feature Map), is used to evaluate and categorize the high-dimensional QoS parameters in various 2-D map fields. The map (or feature map) is a sheet-like artificial neural network that behaves as a competitive network where neurons compete for the best matching weight vector to the input signals. Neurons that best match the input signals patterns cluster together or "organize", topologically preserving the relative relationships of the input signal. The cell locations, and resulting coordinate system, reflect the relative domain of the input signal [117]. Figure 7.2 shows a SOM neural network and the two-dimensional topology of its neurons [118].
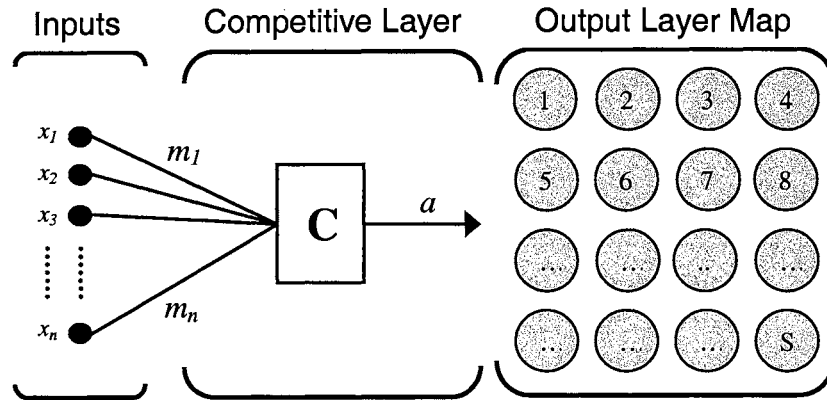
Figure 7.2 Self-Organizing Map

Training Process of the Self-Organizing Map

In the training process of the SOM neural network, input signals are presented to the network where neurons compete for the best matching weight vector to the input signal. At each step the winning neuron and its neighbors, within a certain neighborhood, are updated using the Kohonen rule [118],

$$m_i(t+1) = m_i(t) + h_{c(x),i}(x(t) - m_i(t)), \qquad (7.3)$$

where m is the weight vector (initially set to random values), x is the input vector, $h_{c(x),i}$ is the neighborhood function, and t is the index of the regression step. The input vector x is compared with every weight vector m_i using the script $c(x)$ based on the Euclidean distance metric,

$$\|x - m_c\| = \min_i \{\|x - m_i\|\} ,$$ (7.4)

where m_c is the winning neuron that best matches input x. The neighborhood function $h_{c(x),i}$ is a Gaussian function that contains the indices of all the neurons within radius R of the winning neuron,

$$h_{c(x),i} = \alpha(t) \exp\left\{ -\frac{\|r_i - r_c\|}{2\sigma^2(t)} \right\} ,$$ (7.5)

where $0 < \alpha(t) < 1$ is the learning rate, $r_i \in R^2$ and $r_c \in R^2$ are the vertical locations on the grid, and $\sigma(t)$ is the width of the neighborhood function. So, the weight vectors of the winning neuron and its neighbors are updated by the learning rate parameter as a function of the neighborhood function. The neighborhood function exponentially decreases the learning rate parameter away from the winning neuron because the exponential function decreases as the distance between neurons i and c increases. For linear training $\alpha(t)$ and $\sigma(t)$ are defined as,

$$\alpha(t) = \alpha(0)(1 - \frac{t}{T}) ,$$ (7.6)

and

$$\sigma(t) = \sigma(T) + \{\sigma(0) - \sigma(T)\}(1 - \frac{t}{T}) ,$$ (7.7)

assuming that T is the total training number. For training efficiency the training process is divided into two steps. The first training step uses a large initial neighborhood radius, followed by a small neighborhood radius for fine tuning the map.

During the training process, when the input x is presented, the weights of the winning neurons and its neighbors will move toward x, eventually, after several presentations, resulting in neighboring neurons having learned weight vectors similar to each other [118].

Euclidian Distance of Map locations

The Euclidian distance metric will provide a means of measuring distance between QoS points on the QoS map. The Euclidian distance metric, as used in image processing applications, provides a straight line distance between two pixels, equation (7.8).

$$d = \sqrt{\sum (X - Y)^2}$$ (7.8)

The results will correlate distance with QoS level. Each map (simulation) will contain a reference quality score, an error free case, which is only a function of the MELP codec encode/decode process on the input speech signal (Original signal compared with encoded and decoded MELP synthesized signal). The reference point will be used to compare all other quality scores. The corresponding distance between points and the overall distance from the reference point will be the relative indication of QoS level.

CHAPTER VIII

FNBDT SYSTEM QUALITY ENHANCEMENTS

The negative effects of potential packet network abnormalities on the output

audio quality of an FNBDT communication system warrant quality enhancements.

Presented below are FNBDT / VoIP system quality enhancements. The quality

improvements include voice packet loss recovery techniques, FNBDT Forward Error

Correction (FEC) for sync management frames, and an adaptive jitter (playback)

buffering scheme.

Voice Packet Loss Recovery Techniques

There are several techniques for recovery of lost packets in voice applications

[98] [99], many of which are summarized in [6]. There are two basic categories of

recovery schemes, transmitter and receiver. Transmitter based recovery schemes include

sending block codes, sending multiple copies of packets, interleaving packets,

retransmission of packets, and adaptive packetization and concealment. Receiver based

recovery schemes fit into three categories: insertion, interpolation, and regeneration [98].

Of all the packet loss recovery techniques mentioned above, receiver based

insertion and interpolation are good candidates for real-time interactive VoIP applications

[98]. Insertion techniques such as frame replacement are simple to implement and have

little complexity and low delay. If a packet loss is detected, silence, noise or adjacent

frames are inserted in place of the lost packet. Interpolation techniques are more complex

than frame insertion, but provide an incremental increase in quality [98]. Two

interpolation techniques are waveform and pitch waveform substitution. Waveform substitution uses the adjacent waveform to generate the replacement frame's waveform, while pitch waveform substitution uses pitch detection to provide a waveform (for the missing frame) with the appropriate pitch length.

A number of techniques are not relevant to real-time interactive audio applications because of the delay and complexity they impose on such a system. Others are more applicable to real-time applications but may lack in quality of reproduced audio compared to non-interactive audio applications. In this performance evaluation silence substitution and frame replication will be used. Silence substitution represents the lower limit for loss recovery, and frame replacement will be used to show an incremental improvement to show the potential of more complex techniques such as interpolation.

The proposed recovery schemes are better suited for low-level packet loss and are not expected to perform well under continuous high packet loss situations. Consequently, for FNBDT sync loss where potentially several consecutive voice frames may be lost when an sync management frame is lost, these loss recovery techniques will not provide adequate loss recovery quality. Therefore, in the next section a FNBDT FEC scheme is presented that will aid in the recovery of sync loss in error prone high bandwidth channels.

FNBDT Forward Error Correction for High Bandwidth Channels

FNBDT uses sync management frames to send encryption synchronization information to the receiver system for decryption of FNBDT formatted data. These sync management frames hold critical information for the operation of the system. Unlike a single voice frame of data, if a sync management frame is lost the system quality can be

drastically degraded as several frames of data, corresponding to seconds of speech, will potentially be useless to the receiver. Therefore, it seems important that these frames of data be protected or recoverable at the receiver. Transmitter based loss recovery schemes for audio applications summarized in [6], although not applicable to real-time voice, may have the greatest potential for recovery of sync management frames in an FNBDT implementation where bandwidth is available.

Media-independent transmitter based FEC schemes use block coding to produce additional packets to aid in the correction of packet errors or loss [98]. These additional packets may contain a parity coding or Reed-Solomon coding of whole packets. The bits can be used to code the bits of the entire packet containing the sync management frame for correcting (or recovering) the loss of the whole packet. The additional delay should be minimal, and the increase in bandwidth should be small since sync management frames are sent 1 in every 23 or 24 MELP frames.

Media-specific FEC is another implementation that may be applicable to FNBDT sync management frame recovery. It involves sending multiple copies of each frame (not whole packets) in successive packets, figure 8.1 [98]. With a sync management frame this can be accomplished by replicating, at the transmitter, the sync management frame with no additional delay and no significant bandwidth increase. The redundant sync management frames are used to replace lost sync management frames to recover sync loss at the receiver. Depending on the number of frames included in a packet, the sync management frame can be included in every packet, every other packet, every 5 packets, every 10 packets, etc. within a superframe.

1 2 3 4 Original Stream

1 1 2 2 3 3 4 Media-specific FEC (redundancy)

1 1 2 ███ 3 4 Packet Loss

1 2 3 4 Reconstructed Stream
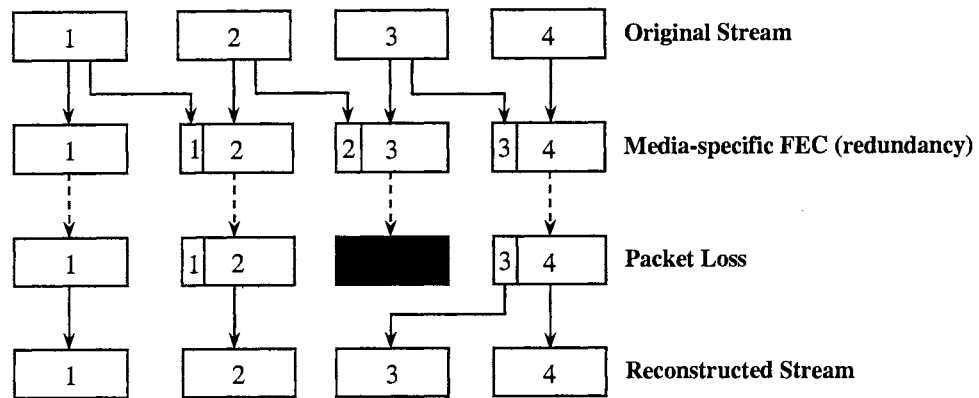
Figure 8.1 Repair using Media-Specific FEC [98]

Adaptive FEC–Based Error/Rate Control Scheme for Sync Management Frames

It is intuitive that the more packet redundancy the better the packet loss recovery at the receiver system. However, there are two things to consider when employing redundancy: 1) the more redundancy the higher the bandwidth and 2) if loss rates are low (or even zero) there is no need to include redundant information. Therefore, there must be a balance in the amount of redundancy to achieve an objective loss rate after recovery and bandwidth conservation.

Bolot *et. al.* [119] introduced an adaptive FEC and rate control algorithm for real-time audio data over the Internet. This scheme adapts FEC to varying loss conditions in the network. Varying amounts of redundancy are determined to minimize loss rate and conserve bandwidth. Mean loss rate feedback from the receiver system is used to determine the amount of redundancy (and combination) needed at the transmitter system to achieve a loss rate closest to a predetermined target loss rate at the receiver.

The network loss process was modeled with redundancy to determine effects of redundancy on loss rate. The network loss process is modeled as a 2-state Gilbert model where the mean (steady state) loss rate is $\pi_1 = \dfrac{p}{p+q}$ without redundancy. Bolot *et. al.*, show that by adding a single redundant packet, the loss rate after recovery becomes

$\pi_2 = \dfrac{p(1-q)}{p+q}$. A single redundant packet assumes packet *n* contains redundant

information about packet *n-1* only, and the current packet is lost if the current packet and the next packet (containing the redundant information) is also lost. This showed that by adding a single redundant frame the loss rate is reduced proportional to *(1-q)*, i.e. if *q =* 0.70 that means a 70% loss reduction. Table 8.1 below shows loss rates after recovery for zero through three redundant packets along with the corresponding interleaving configuration in the packet stream. Table 8.1 also includes an example with parameters *p*=0.05 and *q*=0.07, which shows that the loss rate decreases with the addition of redundancy.

Table 8.1 Redundancy and Loss Rates After Reconstruction [119]

Redundancy			Loss rate after recovery	Example $p=0.05, q=0.7$
none			$p/(p+q)$	6.7%
n-1			$p(1-q)/(p+q)$	2%
n-2			$(p^2 q + p(1-q)^2)/(p+q)$	0.83%
		n-3	$(p(3pq - p^2 q - 2q^2 p + 1 - 3q + 3q^2 - q^3))/(p+q)$	0.54%
n-1	*n-2*		$p(1-q)^2/(p+q)$	0.6%
n-1		*n-3*	$(p(1-q)(pq + 1 - 2q + q^2))/(p+q)$	0.25%
	n-2	*n-3*	$(p(1-q)(pq + 1 - 2q + q^2))/(p+q)$	0.25%
n-1	*n-2*	*n-3*	$(p(1-q)^3)/(p+q)$	0.18%

Although this technique is used for FEC of audio data, it has potential for use with FNBDT sync management frames. This adaptive FEC technique will send redundant copies of sync management frames to improve sync loss rate while minimizing bandwidth. However, a slight modification of the above loss rates after reconstruction equations must be made. The network loss process above is assumed Gilbert, but the sync loss process based on the sync loss model is Bernoulli, where $p = \lambda_2 /(\lambda_1 + \lambda_2)$, and $\lambda_1 + \lambda_2 = 1$. So, loss rate without redundancy is therefore equal to p and the loss rate with one redundant packet is $p(1-q)$, and so on.

Redundancy of up to 3 redundant sync management frames per superframe will be considered and tested in this implementation. The potential for burst losses in the Internet and CDMA networks warrants spacing the FEC sync management frames, as consecutively sent FEC packets (suggested above for audio data), have a greater potential for loss due to burst losses. Therefore, to maximize the usefulness of FEC and minimizing the susceptibility of loss of FEC due to consecutive burst losses, FEC frames will be space evenly throughout the FNBDT superframe. For, example for a superframe size of 24, 1 redundant sync frame will be sent in the middle of the superframe between frame 12 and 13. For the case of 2 redundant frames, FEC frames will be sent to effectively divide the superframe into $1/3^{rds}$, after frame 8 and again after frame 16. Likewise, 3 redundant frames will be sent to divide the effectively divide the superframe in to 1/4ths, after frames 6, 12, and 18.

The mean sync loss rate from the receiver system will be fed back every 5 seconds; this assumes the use of RTCP (Real-Time Control Protocol)[6]. The number of redundant packets (and combination) will be chosen based on the minimum redundant packets (containing sync management frames) required to achieve the lowest possible average sync loss rate. The simulator will simulate sending the corresponding redundant packets from the source system. Finally, the loss rates before and after recovery of sync management frames will be shown and evaluated for this algorithm.

Adaptive Playback (Jitter) Buffering Schemes

As packets traverse a packet network from source to destination they may arrive at their destination with different inter-arrival times (jitter). For real-time audio applications such as an FNBDT/VoIP implementation, audio data must be received in a timely fashion to provide a continuous stream of data to the playback device. Packets received after their scheduled playback time will be discarded by the application, figure 8.2 (a). Packets containing audio data that experience large delays can result in "starvation" of the audio codec and will result in discontinuous audio playback which is annoying to the listener. Therefore, the playback routine at the receiver must compensate for jitter.

Receiver buffering is the primary means of smoothing jitter, Figure 8.2 (b). However, if a packet is delayed by more than the amount of buffering provided in the receiver, the packet will be discarded. Ideally, there should be enough buffering at the receiver to compensate for the variation of packet inter-arrival delays. Consequently,

---

[6] RTCP is capable of sending feedback of mean loss rate p every 5 seconds from the destination [ref 32 of Adaptive FEC paper]

there is a compromise between the jitter buffering delay and overall end-to-end system

delay. As the amount of jitter buffering increases the overall system delay also increases.
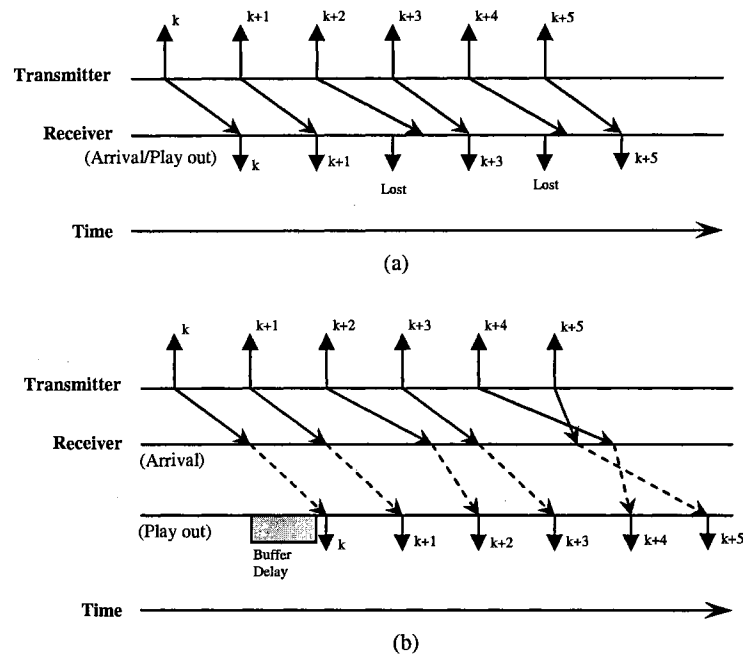
Figure 8.2 Packet Delay Jitter Problems (a) late packets are considered lost at play out (b) buffering corrects late and out-of-order packets [100].

The primary objective of a jitter compensation scheme is to maintain a continuous stream

of data to the audio playback device regardless of the variations in received packet inter-

arrival delays. Figure 8.3 ([101]) shows a timeline for packet transmission and reception

from the network.
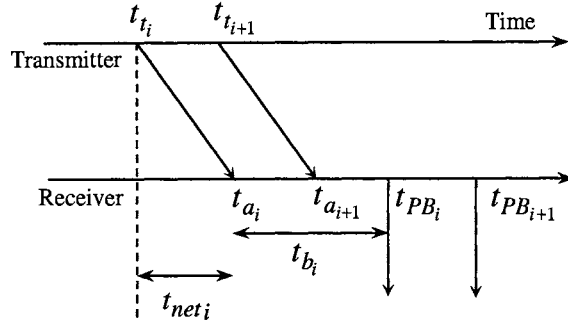
Figure 8.3 Audio Playback Timeline.

The ith packet has a transmit time of t_{t_i}, with an arrival time at the receiver of t_{a_i}.

The uniform interval at which packets are sent is established by the packet time, Δ_{Pkt_i}, which is the time it takes to accumulate a packet, (8.0).

$$\Delta_{pkt} = k * t_f \qquad (8.0)$$

k is the number of voice frames included in a packet and t_f is the voice packet frame length[7]. As the packet traverses the network it will experience a delay of t_{net}, (8.1).

$$t_{net} = t_{a_i} - t_{t_i} \qquad (8.1)$$

The network delay is the difference between the packet arrival time, t_{a_i}, and the packet transmit time, t_{t_i}. The ith audio packet has a scheduled playback time at the receiver denoted by t_{PB_i}, (8.3b). Packets that arrive later than the playback time will be considered lost, (8.3a). Therefore, with the variability of packet arrival times from the network there must be idle time between the packet arrival time and the corresponding

---

[7] MELP frame length $t_f = N / F_s = 22.5ms$ with number of samples N=180, and sampling rate F_s=8,000 samples/second.

audio playback time in order to insure packets arrive in time for their scheduled playback

time. Idle time, t_{b_i} , is included to allow slight variation in packet arrival times, (8.2).

$$t_{b_i} = t_{PB_i} - t_{a_i} \qquad (8.2)$$

The delay, t_{b_i} , is the amount of time the packet spends in the receive (jitter) buffer before

it is played by the audio playback routine. Therefore any packet with an arrival time later

than the playback time is considered late, (8.3a) and (8.3b).

$$t_{a_{i+1}} > t_{PB_{i+1}} \;, \qquad (8.3a)$$

where

$$t_{PB_i} = t_{a_i} + t_{b_i} \;. \qquad (8.3b)$$

In order to play output speech at a constant rate, the receiver must keep a positive idle

time between the arriving packets and the audio playback. This is effectively done by

adjusting the playback time based on network delays. This can be accomplished

adaptively by monitoring network delays or by using a fixed delay (fixed idle time).

A fixed delay implies that all incoming packets will have the same idle time. The

delay (playback time) is initialized at the beginning of the received audio stream, so

before the first audio samples are sent to the audio playback device they are delayed for a

hold time, t_{b_i} (8.4),

$$t_{b_i} = n * t_f \;, \qquad (8.4)$$

where t_f is the frame accumulation time (22.5ms for MELP) and n is the number of

frame lengths of delay. The initial hold time can be any delay value, not just multiples of

frame times as equation (8.4) suggests. A static jitter buffer is the simplest to implement,

nevertheless, it has some drawbacks including the inability to adapt to network

112

conditions. For example, if packets are experiencing delays much less than the predetermined buffer delay, there will be excessive delay imposed on the system by the static jitter buffer. Also, if network delays are larger, even slightly, than the static buffer the packets will be considered late and thus lost. In light of these drawbacks, many have investigated adaptive playback buffering (jitter buffering) schemes that minimized delay while still preventing lost packets due to jitter, including [100] [102] [101] [92] [103].

There are both predictive and reactive approaches to adaptively adjust jitter buffers (playback time) based on network delay. These approaches adaptively respond to changing network delays and dynamically change the playback (idle time) delay. This is accomplished by predicting or calculating a mean delay, $\hat{d}_i$, and standard deviation delay, $\hat{v}_i$. These calculation are typically used to adjust playback time, t_{PB_i}, of the first talkspurt [92], (8.5).

$$t_{PB_i} = t_{t_i} + \hat{d}_i + 4 * \hat{v}_i$$
(8.5)

Subsequent packets belonging to the same talkspurt have a playback time of, t_{PB_j}, (8.6).

$$t_{PB_j} = t_{PB_i} + t_{t_j} - t_{t_i}$$
(8.6)

Various adaptive schemes differ only in the way in which they calculate $\hat{d}_i$. The variance value of $4 * \hat{v}_i$ is estimated to allow the playback time to deviate from the delay enough to minimize the number of late packets [92].

Ramjee *et al.* [92] proposes several algorithms that estimate network delay for playback delay adjustment. One particular adaptive playback buffering algorithm, referred as Algorithm 4, adaptively adjusts the playback buffer based on incoming packet delay measurements that includes adjustment for delay spikes, typically observed in

113

Internet packet delay measurements. Ramjee *et al.* indicates that this algorithm with spike detection performs better for wide-area Internet traces for smoothing network jitter than the other algorithms proposed in their study. Others have conducted research that improve the delay estimates presented by Ramjee *et al.*, including [102] and [100], however the delay estimation improvements come at the expense of increased complexity. The performance and low complexity of the adaptive playback buffering algorithm makes it a good candidate for this implementation of FNBDT over IP networks.

Adaptive Playback Buffering Algorithm with Delay Spike Detection

The adaptive playback buffering algorithm estimates the delay of the ith packet based on the RFC793 algorithm [104]. The delay variance is calculated based on the round-trip-time estimates for the TCP retransmit timer, per a suggestion by Jacobson [105]. The delay of the ith packet is estimated by

$$\hat{d}_i = \alpha * \hat{d}_{i-1} + (1-\alpha) * t_{net\,i} \tag{8.7}$$

and the variance is estimated by

$$\hat{v}_i = \alpha * \hat{v}_{i-1} + (1-\alpha) * \left| \hat{d}_i - t_{net_1} \right| \tag{8.8}$$

The weighting factor α characterizes this delay estimation process, which is a linear recursive filter [92].

Ramjee *et al.* [92] defines a *spike* as a sudden, large increase in the end-to-end network delay, followed by a series of packets arriving almost simultaneously, leading to the completion of the spike. The adaptive playback buffering algorithm uses a threshold to detect a spike, after which it goes into an *impulse* mode. While in *impulse* mode, the

delay estimates are calculated with most recent received delay values to follow the spike

trend. The end of the spike is detected when an exponential function that tracks the slope

of the spike indicates there is no longer a significant slope (no more spike). Once the

spike interval has ended, the algorithm reverts back to *normal* mode, equations 8.7 and

8.8. The pseudo code that describes the adaptive playback buffering algorithm is shown

in figure 8.4.

```
1.   n_i = Re ceiver _ timerstamp − Sender _ timestamp;
2.   if (mode = = NORMAL)
     {
         if (abs(n_i − n_{i−1}) > abs(v̂) * 2 + spkescale)
         {
             var = 0; /* Detected beginning of spike. */
             mode = IMPULSE;
         }
         else
         {
             var = var/ 2 + abs((2n_i − n_{i−1} − n_{i−2})/ 8);
             if (var <= 63)
             {
                 mode = NORMAL; /* End of spike. */
                 n_{i−2} = n_{i−1};
                 n_{i−1} = n_i;
                 return;
             }
         }
     }
3.   if (mode = = NORMAL)
         d̂_i = α * n_i + (1 − α) * d̂_{i−1};
     else
         d̂_i = d̂_{i−1} + n_i − n_{i−1};
         v̂_i = α * abs(n_i − d̂_i) + (1 − α) * v̂_{i−1};
4.   n_{i−2} = n_{i−1};
     n_{i−1} = n_i;
     return;
```

Figure 8.4 Adaptive Playback Buffering Algorithm with Spike Detection.

The above jitter buffer analysis uses end-to-end packet delays to adaptively adjust playback jitter buffers. However, the multi-structured network delay jitter model used in this study produces packet inter-arrival jitter delay values. Therefore, to implement the above adaptive jitter buffering scheme a slight modification is required. To reflect inter-arrival delay, the playback time of the first talkspurt, equation (8.5), becomes

$$t_{PB_i} = \hat{j}_i + 4 * \hat{v}_i ,$$ (8.9)

where, $\hat{j}_i$ is the mean inter-arrival delay between packet i-1 and i, and $\hat{v}_i$ is the variance of the inter-arrival delays. This equation varies slightly from equation (8.5), because the packet transmit time or packet inter-frame time, t_{t_i} , is included in the jitter delays in which $\hat{j}_i$ is calculated. As a result, the inter-frame time is omitted from the calculation. The mean jitter delay is calculated as follows,

$$\hat{j}_i = \alpha * \hat{j}_{i-1} + (1-\alpha) * u_i ,$$ (8.10)

and the variance calculation is

$$\hat{v}_i = \alpha * \hat{v}_{i-1} + (1-\alpha) * \left| \hat{j}_i - u_i \right| .$$ (8.11)

The inter-arrival delay of the ith packet measured from the network (or network model) is shown as u_i, which includes the network jitter delay and packet generation time. Note in equation (8.1), t_{net}, the packet transmit time is subtracted out of the arrival time, so when the playback delay t_{PB}, equation (8.5), of the first talkspurt is calculated the packet generation time has be added.

Now that simulation models, performance measure, and potential performance enhancements have been presented, a performance enhancement will be conducted. The next chapter evaluates the performance of the raw FNBDT system as well as the

enhanced FNBDT system under varying network conditions using models and quality

measures presented above.

CHAPTER IX

FNBDT PERFORMANCE EVALUATION

Network Simulator

A network simulator developed in MATLAB® is used to simulate FNBDT

performance in mixed packet networks. The aforementioned models are implemented

along with FNBDT quality enhancements and quality measures. A graphical user

interface (GUI) allows users to input simulation and model parameters for an efficient

way to analyze different network conditions and simulation scenarios, including various

FNBDT operational modes. Figure 9.1 shows a block diagram of the FNBDT Network

Simulator implemented in MATLAB®. Figure 9.2 shows the graphical user interface of

the simulator application.

```
┌──────┐   ┌─────────┐   ┌─────────┐   ┌──────────┐   ┌──────────┐
│ GUI  │──▶│  MELP   │──▶│ Network │──▶│  Jitter  │──▶│   Loss   │
│      │   │ Encode  │   │ Models  │   │Buffering │   │ Recovery │
└──────┘   └─────────┘   └─────────┘   └──────────┘   └──────────┘
                                                             │
  ┌─────────┐   ┌─────────┐   ┌─────────┐   ┌─────────┐      │
  │QoS Map  │◀──│ Quality │◀──│  MELP   │◀──│ *FNBDT  │◀─────┘
  │   NN    │   │Measures │   │ Decode  │   │  Model  │
  └─────────┘   └─────────┘   └─────────┘   └─────────┘
```
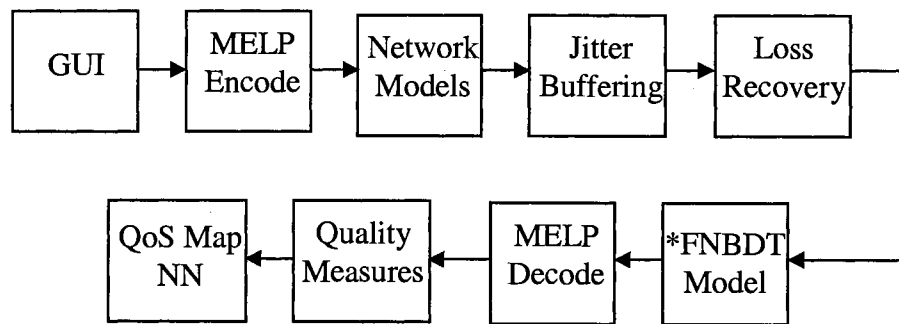
* Includes forward error correction of FNBDT sync management frames

Figure 9.1 MATLAB® FNBDT Network Simulator.
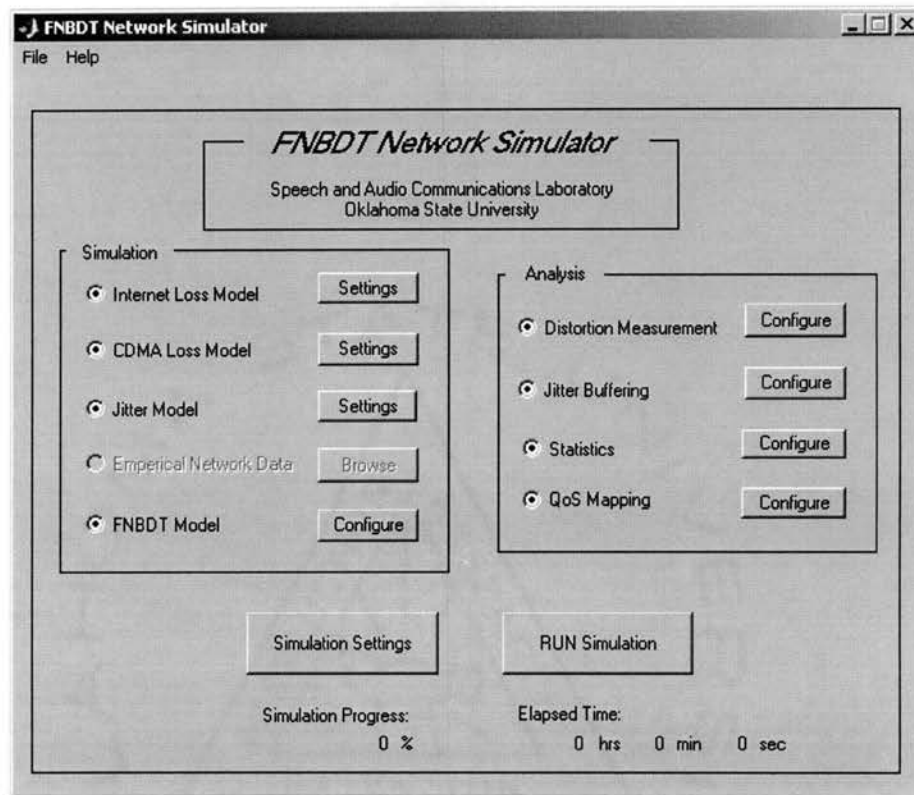
Figure 9.2 GUI FNBDT Network Simulator.

Simulations are initiated by selecting network models or specifying an input empirical data file consisting of collected statistics from a packet trace run on an IP packet network, for example using *NetPerf*. If network models are selected, the user can input individual network model parameters. Next, the FNBDT model parameters are initialized with options of operational mode, FNBDT enhancements, and probabilistic or system quality analysis.

Probabilistic analysis is a probabilistic model simulation that generates simulation statistics such as network loss rates, distribution, jitter buffer length versus time, etc. On the other hand, the system quality analysis involves using a digitized speech input file to

simulate a voice data stream. During the simulation a packet stream is simulated by encoding the speech file with the MELP codec and grouping the resulting MELP frames into FNBDT superframes (if secure FNBDT mode is selected) through insertion of sync management frame headers. The FNBDT frames are collected into packets, where simulation of network effects by the network models is applied to the packetized data stream. This is done by assigning each packet a jitter delay value from the multi-structured network delay jitter model and/or a loss code corresponding to the appropriate loss model responsible for "dropping" the packet. On the received end, the simulated streams of packets are deleted based on the loss code or the jitter value if it is greater than the simulated jitter buffer calculation. The FNBDT sync loss model performs an additional analysis on packets lost by the network model to determine if the packet loss resulted in the loss of a sync management frame. If a sync management frame loss is detected, a superframe, or up to 3 superframes, of MELP data is considered lost. The lost packets are then reconstructed using the appropriate loss recovery scheme. If no recovery scheme is selected, lost packets are removed from the data stream and zeros (for the time interval equivalent to the encoded speech frame within the lost packet) are assigned at the output synthesized speech. Reconstructed packets (MELP frames) and non lost frames are then decoded by the MELP decoder and the synthesized speech output is forwarded to the appropriate quality measure, PESQ, SD_{fw}, or written out to a file for offline listening.

If multiple input speech files or multiple simulation scenarios are chosen (i.e. a range of network model parameters) the resulting quality measurements as well as the corresponding model parameters can be used to train the SOM neural network (QoS map) for QoS level mapping.

Statistics are also available for offline analysis of network model and data

simulations, including packet loss rate, packet loss distribution, jitter delay distribution,

FNBDT synchronization loss (resulting from loss of sync management frames), and delay

(jitter buffer length vs. time). Other simulation settings include frames per packet,

simulation duration, simulation session name, number of files to average per simulation

instance, MELP codec frame size, MELP frame interval, and voice activity detection

simulation. Statistics can be calculated and displayed both during and after simulations.

Performance Evaluation

The goal of this analysis is to determine the performance of an FNBDT

application over a mixed packet network. FNBDT performance in this research will be

measured in terms of packet loss (network loss and late loss), sync loss (packet loss as a

result of sync management frame loss), delay (system delay buffering to accommodate

packet inter-arrival delay), and output speech quality. It is pertinent in this study to

determine performance of FNBDT given network limitations. Raw FNBDT will be

simulated in an attempt to obtain insight on limitations that non-ideal packet networks

will have on FNBDT system operation. After which the proposed FNBDT enhancements

will be analyzed individually to determine performance improvements to the raw FNBDT

system.

In the following sections, a MELP and FNBDT quality analysis is performed in

the presence of packet loss. The forward error correction of FNBDT sync management

frame is employed to study the potential improvements in sync loss due to loss of sync

management frames. Finally, a jitter buffer evaluation is performed to show the effects of

late packet loss due to inadequate receiver buffering on FNBDT quality as well the benefits of jitter buffering.

Clear MELP Quality Analysis

Clear MELP Distortion with Packet Loss: Experiment Setup

To evaluate the quality of MELP in the presence of packet loss, the Gilbert model is used to simulate packet loss on an IP network. This experiment uses a file containing 2 minutes and 8 seconds of digitized speech, 3 male and 3 female talkers in a quiet environment. The file is encoded using the MELP encoder (producing approximately 5,714 frames). The output (parameterized) MELP frames are grouped to form packets; in this case 2 MELP frames are included per packet. Voice Activity Detection (VAD) is not used in this simulation. These packets containing MELP frames are deleted (dropped) according to the probabilistic Gilbert model. The p and q values used for each simulation are calculated based on the steady state probabilities in equation 6.1, shown in equations 9.1.

$$p = \frac{ulp(1 - clp)}{1 - ulp}, \qquad q = 1 - clp \qquad (9.1)$$

For example, in figure 9.3 for ulp=0.10 and clp=0.10, p=0.10 and q=0.9, and for ulp=0.10 and clp=0.40, p=0.066 and q=0.6, etc. Two frame insertion techniques are used when frames are deleted: silence substitution and frame replication. Silence substitution simply replaces the lost frame with silence (zeros). Frame replication replaces the missing frame with an adjacent frame. These two replacement schemes are the most common and the simplest to implement, with silence substitution representing the lower bound for quality of frame replacement strategies. The silence substituted or replicated MELP frame is then

122

fed to the MELP decoder for synthesis. The spectral distortion is calculated for the synthesized speech using the frequency weighted spectral distortion measure on a frame by frame basis. The resulting speech file that has errors introduced by the Gilbert model is compared to the original file synthesized by MELP without error. The average spectral distortion, as well as the percentage of outlier frames, is then calculated for the entire file.

Clear MELP Distortion Evaluation with Loss Recovery Techniques

Figure 9.3 shows the average frequency-weighted distortion measurement for the MELP vocoder with frame deletions based on the unconditional loss probabilities (*ulp*) and conditional loss probabilities (*clp*) as inputs to the proposed Gilbert model. Several conditional loss rates were used to test MELP's reaction to burst loss. Two frame replacement strategies for lost frames, silence substitution and frame replication, are plotted for comparison. Figure 9.4 shows the corresponding percentage of outlier frames with average spectral distortion greater than 2 dB. Both figures (9.3 and 9.4) show that overall average spectral distortion of MELP frames is low for high loss rates, less than 2 dB. However, the percent outlier frames are well above the tolerable 2% at high loss rates. Moreover, at higher loss rates the percentage of outlier frames for silence substitution increases rapidly. This results in degradation in quality even with lower loss rates. Spectral distortion climbs above 1 dB starting at *ulp* of 20%. On the other hand frame replication seems to fare well for all loss rates; spectral distortion remains consistently below 1 dB to *ulp* rates of 45%, while percent outliers stay below 2% to *ulp*=10% and below 4% to *ulp*=20%. Spectral distortion for frame replication never reaches above 8%, compared to 30% for silence substitution.

123
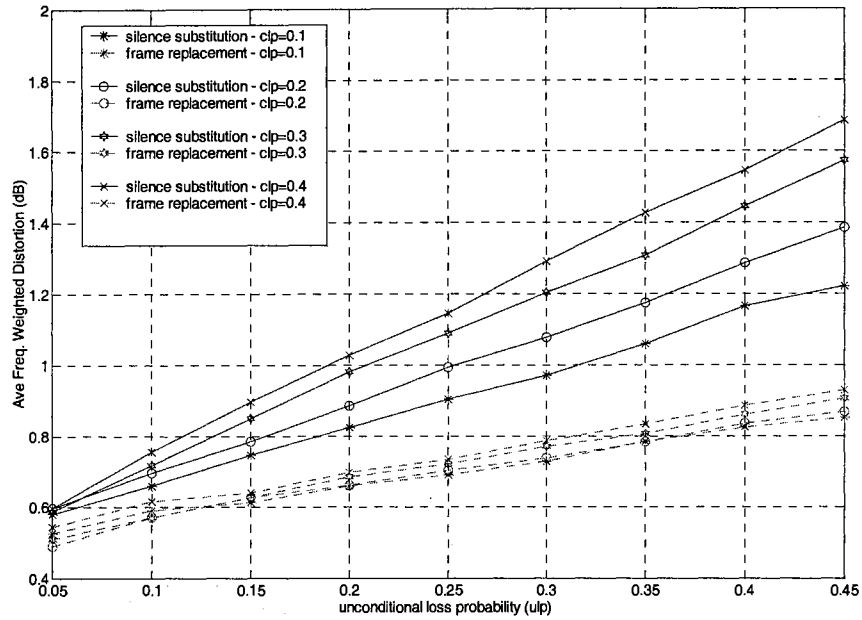
Figure 9.3 Average Frequency-Weighted Spectral Distortion of MELP with Packets Deleted Based on Gilbert Burst Loss Model.
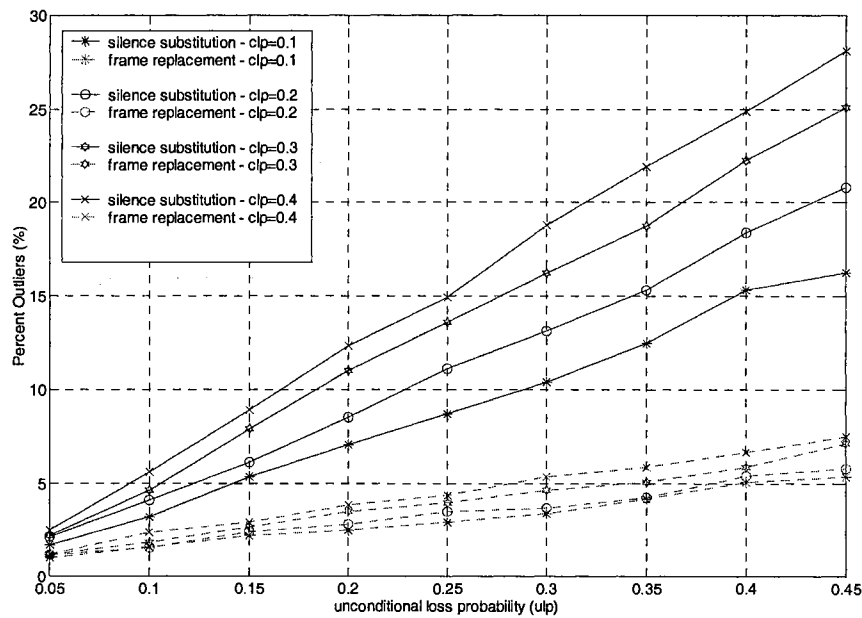
Figure 9.4 Percentage of Outlier Frames with Spectral Distortion Greater Than 2.0 dB, with Gilbert Burst Loss Model.

124

The results for spectral distortion for frame replication are deceptive when viewed alone. When coupled with percent outliers a more complete picture is presented. Listening tests confirm that at *ulp* loss rates above 20%, perceptual quality is significantly degraded (unacceptable) for frame replication as well as silence substitution. This is apparent in the percent outlier results. A loss rate of 20% represents an effective upper limit on MELP quality vs. network loss rate (*ulp*) for frame replication. Although the percentage of outlier frames does not achieve transparent distortion of <2% outlier frames at *ulp*=20%, the majority of listeners that participated in an informal listening test (see listening test section below) for both male and female speech files indicated that the quality of the synthesized speech is acceptable. This shows that acceptable quality is achieved for 4% outlier frames, with an average spectral distortion below 1 dB. However, as loss rates increase above 20% synchronization will be difficult to maintain with FNBDT and the system will have to take action to adjust to the extreme network conditions.
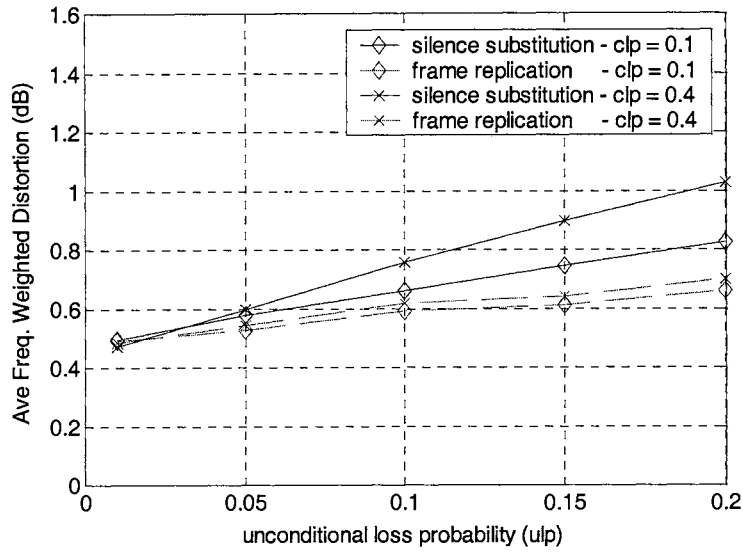
**Figure 9.5 Average Frequency-Weighted Spectral Distortion of MELP for *Ulp* Rates
< 20%.**
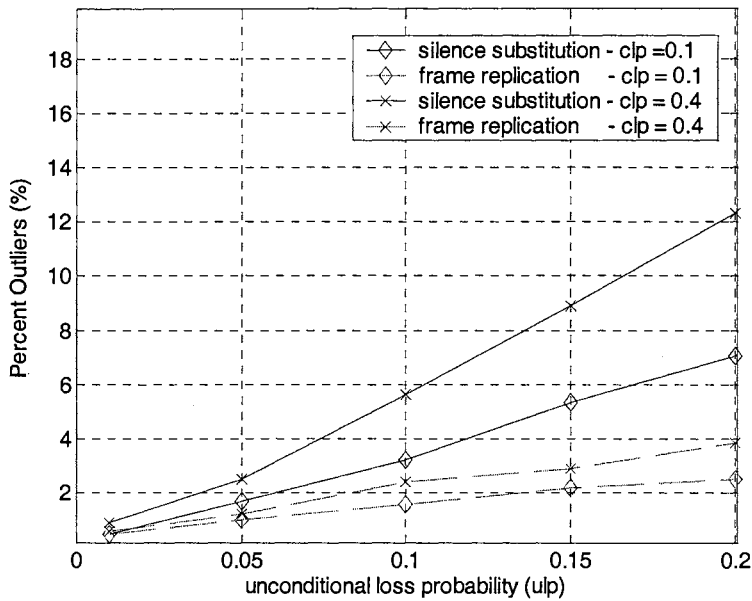
**Figure 9.6 Percent Outlier Frames with Spectral Distortion Greater Than 2.0 dB,
Ulp < 20%.**

Figures 9.5 and 9.6 show that MELP is robust, achieving transparent distortion, to

high burst loss rates (*clp*) at low loss rates, *ulp* less than 10%. There is also an obvious

126

improvement in spectral distortion and percent outlier with loss concealment, as evident in frame replication, which yields almost indistinguishable results for *clp* rates.

There are factors in human speech, as well as in the MELP vocoder algorithm, that may contribute to MELP's apparent robustness to low error rates when a frame replacement strategy such as frame replication is used. First of all, human speech does not change very much over short time periods (i.e. MELP frame accumulation time). Second, the MELP algorithm interpolates all parameters across frames (MELP implements frame look-a-head and look-back) to insure smooth transitions between frames of the synthesized speech output [97]. Evidence of the promise of frame replacement strategies also emanates from the algorithm by the way it deals with bit errors detected in a MELP frame (parameterized MELP vocoder frame). The MELP algorithm uses Hamming codes to correct and detect single and double bit errors respectively. If an error is uncorrectable it uses the previous frame's parameters to represent the current frame that is in error. This implies that if MELP uses frame replication, an acceptable amount of quality can be expected from this procedure. Finally, there is a high amount of redundancy in the output parameters of MELP, specifically in the most significant bits of the pitch and gain parameters. Rahikka *et. al.* ([21]) reports that the pitch and gain parameters, which represent the "important" perceptual parameters, have redundancy that exceeds 50%, with the MSVQ1[8] parameter approaching 20% redundancy. Replicating frames, on the short term, effectively takes advantage of the redundant character of these output parameters, intuitively achieving perceptual quality.

---

[8] MSVQ (Multi Stage Vector Quantizer) stages which characterize the LPC coefficient line spectral frequencies (LSFs).

The results from the spectral distortion measure at high loss rates are also misleading in that mathematically the spectral distortion is high for silence as opposed to frame replacement. When comparing the spectral frame of a replicated packet (for N consecutive packets), the redundancy of packet parameters will give mathematical results that are lower on average than comparisons to silence substituted (zeroed) frames. However, perceptually, frame replication at high loss rates will audibly produce sustained sounds, potentially on the order of seconds, which is annoying to listeners. On the other hand, silence or even comfort noise has the potential to be more pleasing to listeners. These arguments further show the importance of listening tests to validate mathematical results. The following section shows the results of informal listening tests taken from this experiment.

Validation of SD$_{fw}$ Measurement using Informal Listening Test

Informal listening tests were conducted with 13 listeners, using selected test files from the simulation above. Individual listening tests for both male and female talkers were performed. Untrained listeners were given two types of tests, an acceptability test and a comparison test. For the acceptability tests, listeners were asked if the quality (naturalness and intelligibility) of the speech was acceptable, unacceptable, or no opinion. The comparison tests involve listeners comparing two files and selecting the file they perceived as having better quality, or selecting "same" if speech files were close to one another in quality. The results of the male and female informal listening tests are reported in tables 9.1 and 9.2 below.

Table 9.1 Male Results of Informal Listening Tests.

Quality	Acceptability Test			Robust to Burst Loss Test	
	Silence Sub. *ulp*=0.4, *clp*=0.3	Frame Repl. *ulp*=0.4, *clp*=0.3	Frame Repl. *ulp*=0.2 *clp*-0.1	Conditional loss rate	Frame Repl. *ulp*=0.05
Acceptable	7.7 %	19.2 %	53.9 %	*clp*=0.1	15.4 %
Unacceptable	92.3 %	80.8 %	46.1 %	*clp*=0.4	15.4 %
No opinion	0 %	0 %	0 %	same	69.2 %

Table 9.2. Female Results of Informal Listening Tests.

Quality	Acceptability Test			Robust to Burst Loss Test	
	Silence Sub. *ulp*=0.4, *clp*=0.3	Frame Repl. *ulp*=0.4, *clp*=0.3	Frame Repl. *ulp*=0.2 *clp*-0.1	Conditional loss rate	Frame Repl. *ulp*=0.05
Acceptable	7.7 %	7.7 %	76.9 %	*clp*=0.1	7.7 %
Unacceptable	92.3 %	92.3 %	23.1 %	*clp*=0.4	23.1 %
No opinion	0 %	0 %	0 %	same	69.2 %

The results of the acceptability tests at extremely high loss rates, *ulp*=0.4 and *clp*=0.3, for silence substitution for both male and female talkers were 92.3% unacceptable. Frame replication for male and female talkers were 80.8% and 92.3% unacceptable, respectively. Even though mathematically the spectral distortion is low, with a high percentage of outlier frames the quality is unacceptable. The biggest complaints from listeners were that silence substitution has several "pops" before abrupt changes to silence; frame replacement was smoother but had several areas of sustained sounds. Both were classified as annoying. These loss rates represent extreme network conditions and consequently in an FNBDT application, if loss rates are this high, synchronization will be hard to maintain, resulting in the system attempting to resynchronize or even reconnect.

The acceptability tests for lower loss rates were conducted for frame replication at *ulp*=0.2 and *clp*=0.1, because of its favorable results in the distortion measurement. Listening test results were unanimous at 76.9% acceptable for female talkers and slightly

129

less acceptable at 53.9% for male talkers. At 20% packet loss rate the average spectral distortion is below 1 dB, however, the percentage of outlier frames is 4%. This establishes 1 dB of spectral distortion with 4% of outlier frames as an upper limit on acceptable quality. The final listening test considered here is the evaluation of MELP's robustness to burst loss at low loss rates. Listening tests show that for low rates of *ulp*=0.05 and *clp*=0.1 and 0.4, 69.2% of the listeners (for both male and female talkers) indicated that MELP had the same quality for both loss rates. Again, this is possibly due to MELP's internal smoothing mentioned above.

The performance improvement of frame replication over baseline silence substitution gives promise to other proven frame replacement strategies such as frame interpolation. Other frame replacement schemes, including insertion, interpolation and regeneration, as well as forward error correction schemes, can be used to further improve the quality of MELP in error prone packet environments.

FNBDT Quality Analysis

In the following analysis FNBDT system quality is assessed. Two FNBDT modes are used for determination of FNBDT performance in packet network environments, Clear MELP and Secure Blank and Burst modes. Frame Replication is used for recovery of lost packets for each mode. Both male and female simulations were run, however for brevity, this analysis will focus on the male case as both male and female simulation yield similar results. The female simulation results can be found in the Appendix.

Experiment Setup

The evaluation of FNBDT performance (including jitter buffering schemes) is performed with speech data that follows the P.862 ITU-T recommendations for PESQ speech, which states that speech data, used with the PESQ algorithm, represent the temporal structure including silent intervals and phonetic structure of real speech signals [111]. The PESQ recommendation suggests 8 second speech files; each contains a minimum of 2 talkers with a minimum of 200 ms of silence between active speech segments and at least 40% active speech. For talker dependency 8 male and 8 female speech files are averaged respectively, to obtain an overall PESQ score per gender. Speech files were digitized in a quiet background environment.

A wide range of model parameters is used to simulate several different network conditions. Model parameters were selected based on either "real-life" collected network statistics using the NetPerf application, statistics reported in research literature, or parameters typical for communications operating conditions or environments. Internet loss rates reported in the literature include unconditional loss probabilities, *ulp*, up to 40%. This work will consider *ulp* up to 20%. Loss probabilities above 20% yield results that prohibit understandable (or tolerable) voice communication. Conditional loss probabilities, *clp* (burst loss rate), for the Internet model of up to 70% are considered. For CMDA data networks, error rates are dependent on speed of the mobile device as well as fading conditions (fading margins) (equations 6.3 – 6.6). Therefore, this analysis will consider reasonable fading conditions that yield frame error rates (FER) of up to 20%. The normalized Doppler frequency bandwidth, *fdT*, characterizes the CDMA channel burst loss rate and is dependant on speed of the mobile unit. Consequently, the speed of

the mobile unit results in strong or weak correlation of the fading process. Mobile unit speed specified for the different data rates of 3G cdma2000 are considered, see table 9.3 below. The relationship between mobile speed and the corresponding normalized Doppler bandwidth is given by,

$$f_dT = (v/\lambda)T,$$ (9.1)

where T=20 ms is the RLP frame inter-arrival time, and $\lambda = c/f_c$, with f_c=2G Hz and c is the speed of light (3×10^8 m/s).

Table 9.3 3G Mobile Data Rates, Mobile Speeds, and Normalized Doppler Bandwidth.

3G Data Rates	Speed of Mobile Unit	f_dT
144kbps	60 mph (fast moving driving speeds)	3.5
144kbps	35 mph (driving speed)	2.0
384kbps	3 mph (outside or walking speeds)	0.2
384k – 2M bps	0.372mph (Outside and inside fixed)	0.02

The training process of the SOM Neural Network for QoS level mapping of the FNBDT system performance in mixed packet networks involves initialization and selection of several parameters. The maps are size 20 x 20 with a rectangular sheet map topology. The maps are trained sequentially with a total training number of 3000, divided into 2 phases, 1000, and 2000 respectively, for training efficiency. The initial weight vectors are randomly initialized, and the first phase learning factor $\alpha(0) = 0.5$, with the second phase learning factor equal to 0.05. A Gaussian neighborhood function is used with first phase initial and final radius, $\sigma(0)$ and $\sigma(T)$, of 4 and 1, and second phase values of 2 and 1, respectively.

132

Clear MELP Mode vs. Secure Blank and Burst Mode

The potential for sync loss increases as network loss rates increase. This section presents the results of Clear MELP and Secure Blank and Burst mode with male simulations using concatenated Internet and CDMA network models. The models were simulated over multiple network conditions (parameter values) and resulting loss rate was recorded for each simulation run along with the quality score. There were 103 total simulation scenarios run for Clear MELP and Secure Blank and Burst modes. The overall simulated model(s) output packet loss rate and corresponding quality measurement, PESQ and SD_{fw} is shown in Figures 9.7 and 9.9, respectively.
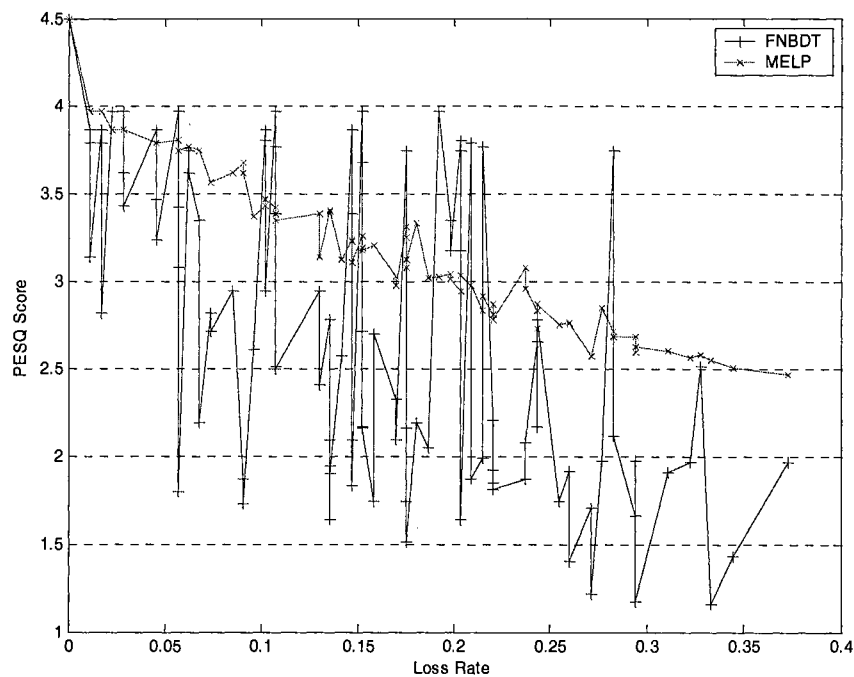
Figure 9.7 PESQ Score vs. Loss Rate for Clear MELP and Secure Blank and Burst FNBDT Modes – Male

Figure 9.7 shows a decrease of PESQ score of Clear MELP mode as somewhat linearly decreasing down to 2.5 at an overall loss rate of 0.35. However, the PESQ score of Secure Blank and Burst falls as low as 1.2 at a loss rate of 0.33. There are several areas that have PESQ quality values that swing from 1.5 to 3.7 for a given loss rate. This is a result of probabilistic models used for determining packet loss. If a sync management frame is lost a large number of consecutive voice frames is lost. The loss of a sync management frame at any given time in the simulation causes the quality to drop very quickly. If you follow the minimum quality values, Secure Blank and Burst mode show difficulty maintaining synchronization with quality decreasing rapidly from good to poor. Figure 9.8 shows the minimum values for Secure Blank and Burst for each loss rate, which represent the minimum possible performance. The overall quality of Secure Blank and Burst mode is significantly lower than Clear MELP mode, which makes communication difficult with secure FNBDT.
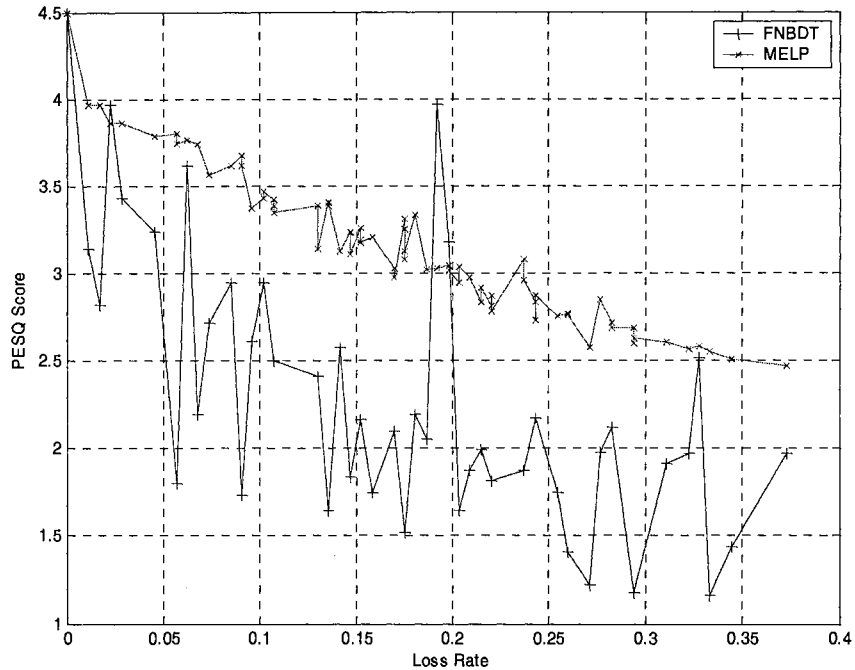
Figure 9.8 PESQ Score vs. Loss Rate for Clear MELP and Minimum Secure Blank and Burst FNBDT Modes – Male

Figure 9.9 shows the results of the same simulation scenario using the SD_{fw} and PO to measure output speech quality. The same trends can be observed for Secure Blank and Burst mode, FNBDT maintains transparent distortion until loss rates reach 0.07, and then synchronization is difficult to maintain resulting in unacceptable quality, $SD_{fw}{>}1$ dB with PO>4%.
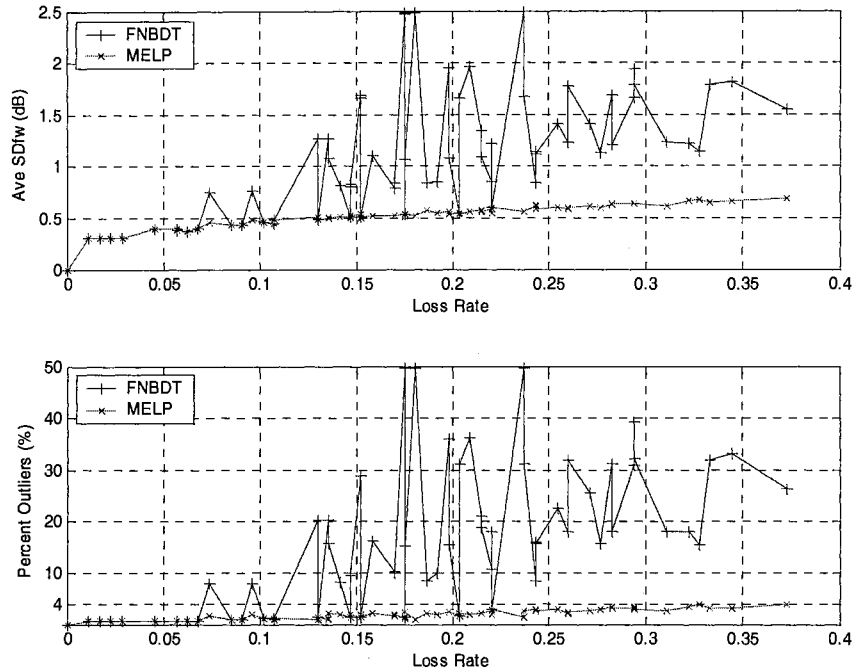
Figure 9.9 Ave SD$_{fw}$ and Percentage Outlier Frames vs. Loss Rate for Clear MELP and Secure Blank and Burst FNBDT Modes – Male

The above data shows that secure FNBDT yields unacceptable quality for mixed network environments when overall loss rates exceed 7%. However, it is unclear how the network parameters contribute to the output system quality. Therefore, to gain insight on what network conditions contribute to the degradation in FNBDT system performance, self-organizing QoS maps are used in the following section.

PESQ QoS Maps: Clear MELP vs. Secure Blank and Burst - Male

To assess the quality of the Clear MELP and Secure Blank and Burst FNBDT modes in relation to network parameters, the following QoS mapping analysis is given. The QoS map for the Clear MELP and Secure Blank and Burst simulation with male talkers is given in Figure 9.10. A map data point represents a simulation run for Clear MELP (filled triangles) or Secure Blank and Burst Mode (non-filled squares). There were

136

103 different simulation runs for both FNBDT modes used for the map. The minimum

quality scores for each output loss rate shown in figure 9.8 were used as representative of

Secure Blank and Burst mode. This enables an assessment of the minimum FNBDT

system quality. Each simulation run was conducted with different network model

parameter values, thus representing a wide range of network conditions. Simulation runs

used speech files following the PESQ recommendation described above consisting of 8

second speech files for male and female talkers. For each simulation scenario, an average

quality score is calculated for 8 speech files. The parameters used in each simulation run

and the corresponding quality score were used to train the self-organizing QoS map

neural network. Simulation runs are displayed on the map with marker symbols

representing Clear MELP and Secure Blank and Burst. In addition to marker symbols,

simulation index numbers correspond to a table entry that lists the simulation parameters

and quality value. A complete table of simulation indexes, parameter values, and quality

scores for each simulation run is located in the Appendix for reference. An example of

table entries for Secure Blank and Burst – male simulation with simulation index and

parameter values used to train the QoS map is shown in table 9.4.

**Table 9.4 Table of Male Blank and Burst Mode Simulation Run Parameter Values
Corresponding to Map Data Points.**

Simulation Index	Parameter values						
	ulp	*clp*	FER	*fdT*	mean	stdev	PESQ Score
:	:	:	:	:	:	:	:
63fr	0.2	0.7	0.01	3.5	x	x	1.5205
64fr	0.05	0.4	0.15	0.2	x	x	2.1924
65fr	0.05	0.7	0.15	2	x	x	2.0554
66fr	0.2	0.4	0.005	0.2	x	x	3.9674
:	:	:	:	:	:	:	:
x – correspond to parameter values that are not modeled in this simulation							

Spatial locations on the map are a result of the training process of the neural network. This is a self-organizing map, so spatial locations are a result of clustering of similar parameter values and quality scores (input vectors). Therefore, the important information lies in the locations of the clusters and the corresponding parameter values within the cluster. Clusters and their relative locations to one another provide over all quality trends of a particular data set, for example Clear MELP simulations. The individual map locations, shown as markers on the map, give an indication of how . parameter values within simulation runs affect quality. To appropriately interpret the QoS map, information about the parameter values at the location of the data points is needed, particularly the quality scores. The following analysis will relate the output quality scores to the network model parameters through the use of self-organizing maps. The goal is to show the overall performance in terms of speech quality output of FNBDT and what role individual parameters have on the output quality.

The simulation parameters for the Clear MELP and Secure Blank and Burst mode simulations (male and female) were used to train the QoS map neural network. The training process of the QoS map neural network results in map model vectors (weight vectors) that best represent the input data set. Once the map model vectors are trained the input data set is re-introduced to the map. The map vector with the minimum distance (best matching) to the input vector is determined for each data vector. The corresponding map index is used to map the location of the best matching input vector. The data set with input vectors that occur most frequently at each map location is chosen to represent the map location, shown with various marker types (square, triangle, circle, etc.). A reference point for quality is included on the QoS map for a no error case that gives the upper limit

on quality. In figure 9.10, the reference point, shown as a star, is located in the lower left corner of the map. As the data set move diagonally toward the upper right corner away from the reference point the quality decreases.
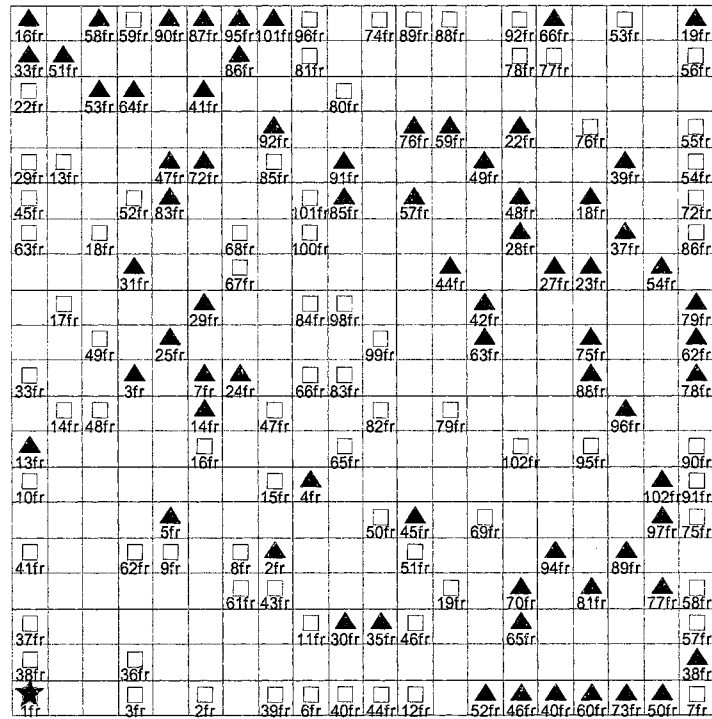
▲ fr – Blank and Burst   ☐ fr – Clear MELP

Figure 9.10 QoS Map: Clear MELP vs. Secure Blank and Burst – Male

The Euclidean distance between neighboring map units is calculated and presented on the unit distance map, figure 9.11, using all input parameters. The color bar in figure 9.11 shows the colors and corresponding distance values represented on the map. Clusters, map locations with similar values, are areas of low distance values on the unit distance map, shown as lighter colors. High values on the unit distance map mean

139

larger distances between neighboring map units, indicating cluster borders. In terms of quality these distances show the separation in quality.

The unit distance map reveals a large majority of the Clear MELP data set is clustered in the lower left corner near the reference point, which is expected from the above findings, figure 9.7 and 9.9. Likewise, there are clusters of the Secure Blank and Burst data set in the upper left and lower right corners as well as slightly lower than the upper right corner of the map. Distance values separating the upper right corner Secure Blank and Burst cluster from the Clear MELP cluster in the lower left corner are as high as 1.02 along the diagonal.
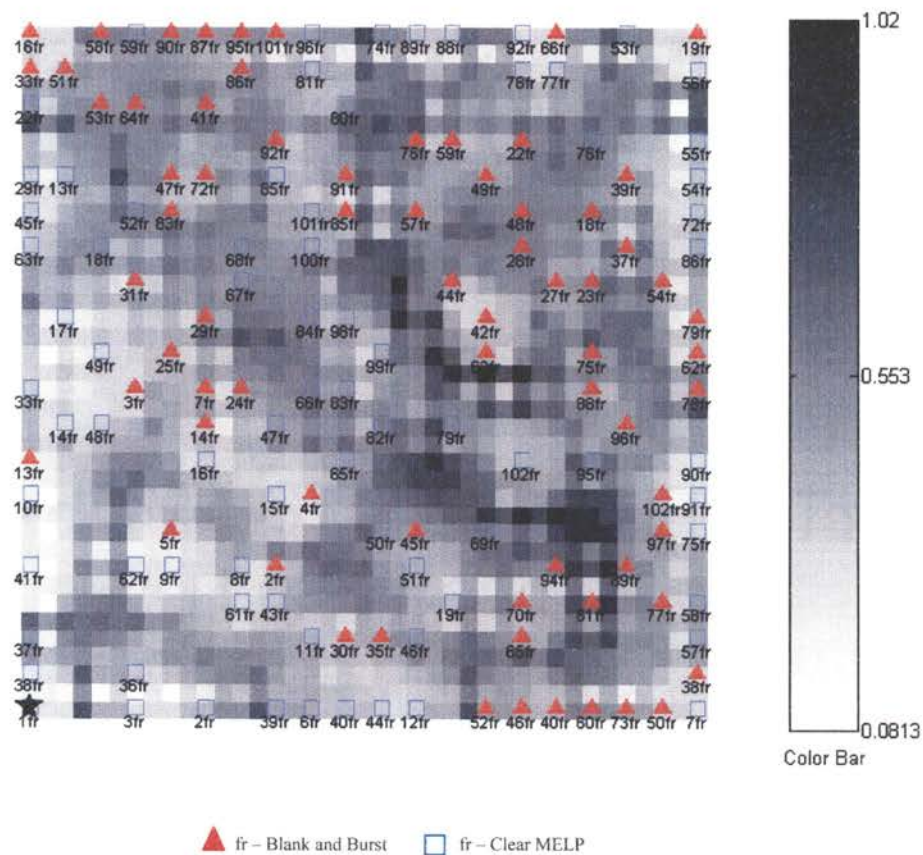
Figure 9.11 U-Matrix QoS Map: Clear MELP vs. Secure Blank and Burst – Male

140

Figure 9.12 is a color map of the PESQ component vector. A color bar to the left of the color map indicates the PESQ score for each region of the map. The direction of increasing quality is from the lower left corner to the upper right corner of the map. The PESQ color map shows that the distances that separate the clustering of data sets are an indication of separation in quality. The Blank and Burst clusters of the top left corner, upper right diagonal, and lower right corner are areas of poor quality (PESQ score less than 2). On the other hand, the Clear MELP data clusters yield good PESQ scores (greater than 3).
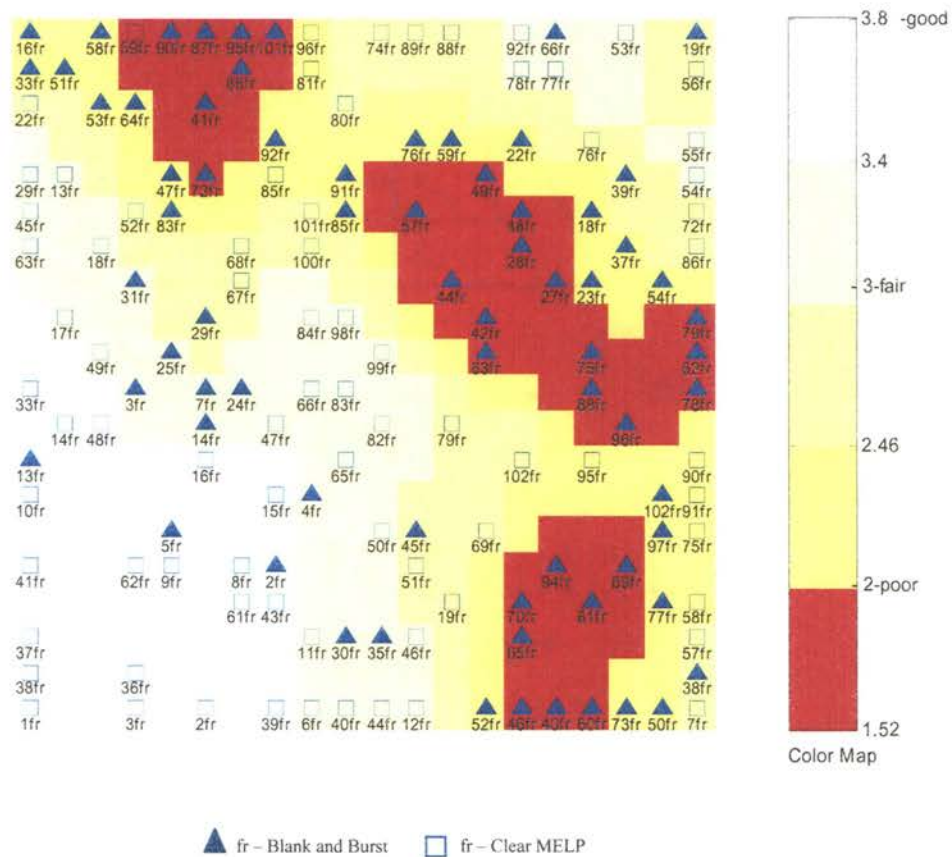
Figure 9.12 PESQ Color QoS Map for Clear MELP vs. Secure Blank and Burst – Male

To show which parameters have the largest impact on quality level change a pie chart that is made up of the distance measurement relative to each parameter is used. A distance measurement is made for each parameter and individual pie charts for each map location are calculated. Each pie chart shows the relative proportion of each parameter, scaled relative to all parameters. An indication of parameter importance is at cluster borders. At cluster borders (large distances between map locations) values change rapidly, and result in large changes in quality. Therefore, these areas will have larger values on the pie chart for the particular area.

Figure 9.13 is the distance pie chart map for the Internet parameters *ulp* and the burst loss rate parameter, *clp*. The size of the portion of the individual pie chart represents distance and shows the rate of change of the parameter value. Large distances represent a rapid change in the parameter and indicate that the parameter is the primary component for the change in difference of quality for that region of the graph. Figure 9.20 shows a rapid change along the center diagonal (upper left to lower right) for the *ulp* parameter, and the *clp* parameter is most significant for the upper right and lower left corners.
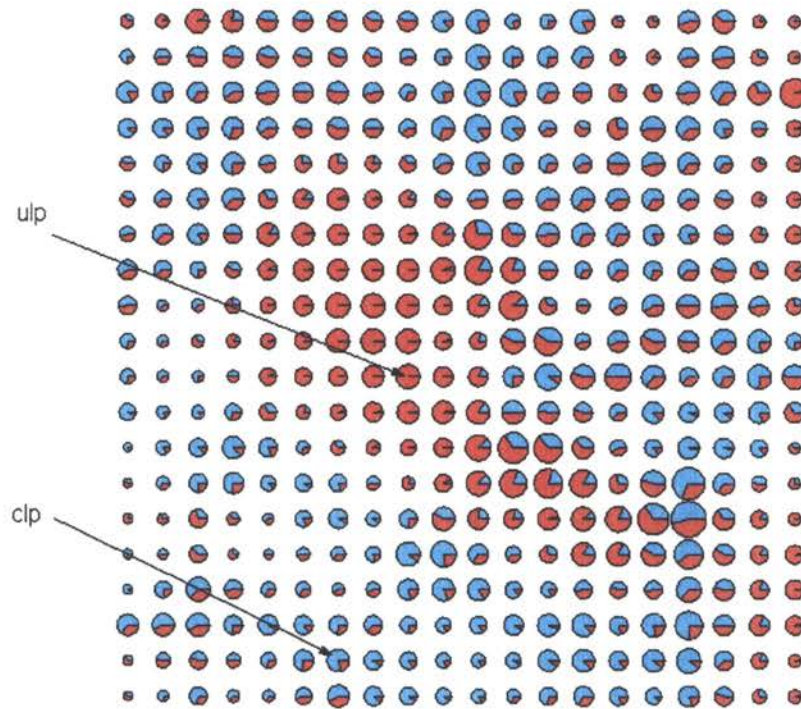
Figure 9.13 Relative Importance of Internet Model Parameters *ulp* and *clp* - Male

Figures 9.14 and 9.15 are the color maps for the *ulp* and *clp* parameters. They

indicate what the parameter values are in the clusters. These figures coupled with the

PESQ color map and the distance pie chart map, figure 9.13, show how they affect

quality. By examining figure 9.14 (*ulp* color map), one can see that along the upper left to

lower right diagonal there is a rapid change in *ulp* value, from low (0.1) to high (0.19) in

a short distance. Figure 9.15 also shows a rapid change from low to high values of *clp* in

the upper right and lower left corner toward the center of the map of 0.069 to 0.7. In

relation to the impact these parameters have on quality, the corresponding parameter

color maps must be viewed with the PESQ color map, figure 9.12. In the lower left

corner, the *ulp* color map, figure 9.14, has low values and from the PESQ color map, figure 9.12, the majority of data mapped within that area has good quality according to PESQ score, i.e. Clear MELP simulation data points *38fr, 36fr, 62fr*, and Secure Blank and Burst data points *2fr, 13fr*, and *30fr*. For high values of *ulp* just above the diagonal, there are Secure Blank and Burst data with poor quality, i.e. *75fr, 62fr,* and *86fr*. By cross referencing the *ulp* color map with the PESQ color map, the quality scores quickly change either side of the rapidly changing diagonal in inverse proportion to the *ulp* value change. As loss rate increases, quality score decreases. For example, Secure Blank and Burst simulation results change from good (with low *ulp*) to poor (with high *ulp*) on either side of the diagonal, as indicated by data simulation runs labeled *4fr* and *57fr*, with *ulp* rates of 0.01 and 0.15, respectively.

fr – Blank and Burst   fr – Clear MELP
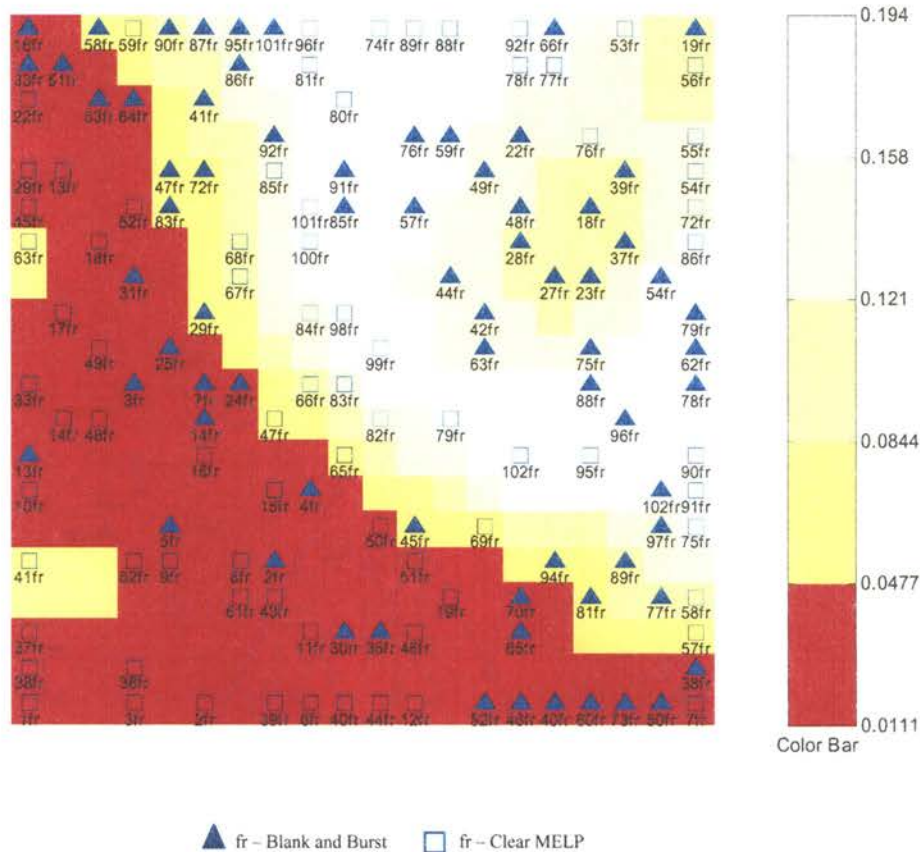
Figure 9.14 *Ulp* **Color Map: Clear MELP vs. Secure Blank and Burst - Male**

The *clp* color map, figure 9.15, viewed with the PESQ color map (figure 9.12)

shows the lower left corner and upper right corner of the map correspond to low *clp*

values with good quality scores for the data within the region. However, as *clp* values

rapidly change moving into the center diagonal, the corresponding quality scores

decrease. In the lower left corner of the map as *clp* values increase toward the center

Clear MELP scores move from good to fair quality, for example simulation runs labeled

2fr (good) to *19fr* (fair). In areas of high *clp* and low *ulp* Secure Blank and Burst quality

scores move from good to fair range (*13fr* to *2fr*), while high *clp* values in lower right

center diagonal quality scores remain in the poor range (ie. *70fr, 94fr*). In the upper left

145

corner of the map, high values of *clp* results in a significant cluster of Secure Blank and

Burst simulation data points in the poor quality region of figure 9.19, ie. *48fr, 44fr, 42fr,*

63fr, 57fr, 88fr, 75fr, 27fr, 28fr and *96fr*. This shows the burst loss has a major effect on

the reduction of quality of Secure Blank and Burst mode.
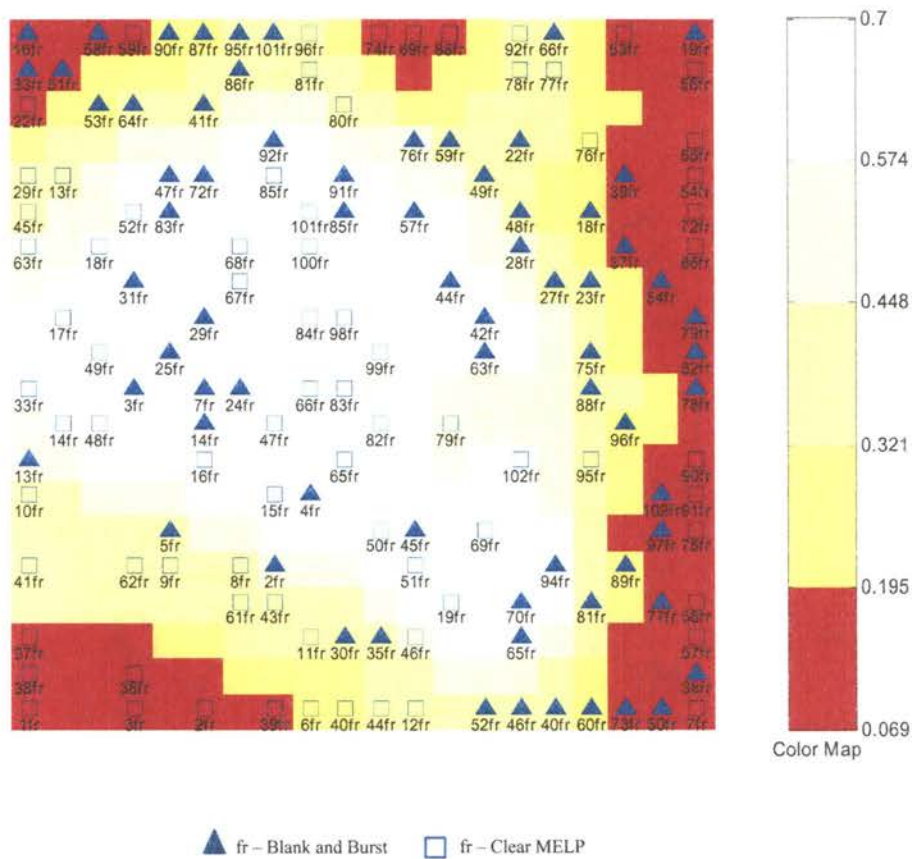
Figure 9.15 *Clp* Color Map: Clear MELP vs. Secure Blank and Burst – Male

The distance pie chart map of CDMA model parameters, figure 9.16, indicates

significant contributions of fading loss, FER, in the upper left and lower right corners of

the map, while normalized Doppler bandwidth, *fdT*, contributes along the center diagonal

from the lower left to the upper right. The FER parameter, figure 9.17, undergoes a rapid

change from low to high values from the center diagonal to the upper left and lower right

146

of the map. High values of FER in the corners result in clusters of poor PESQ quality
scores for Secure Blank and Burst runs found on figure 9.12, *60fr, 40fr, 70fr, 81fr, 77fr,
97fr, 102fr* and *94fr*. Likewise, low FER values in the lower left corner have good quality
scores, more so with Clear MELP than Secure Blank and Burst. But, in the upper right
corner Secure Blank and Burst quality values are poor even with low FER values. This is
due to the large burst loss rate, *clp*, in that region. In figure 9.15 along the diagonals of
rapid change in FER, the corresponding data PESQ scores go from good to poor in the
direction of the corners, ie. Secure Blank and Burst *14fr* (good – FER=0.05) to *90fr* (poor
– FER=0.2) upper left, and *4fr* (good – FER=0.005) to *81fr* (poor – FER=0.2) lower
right.
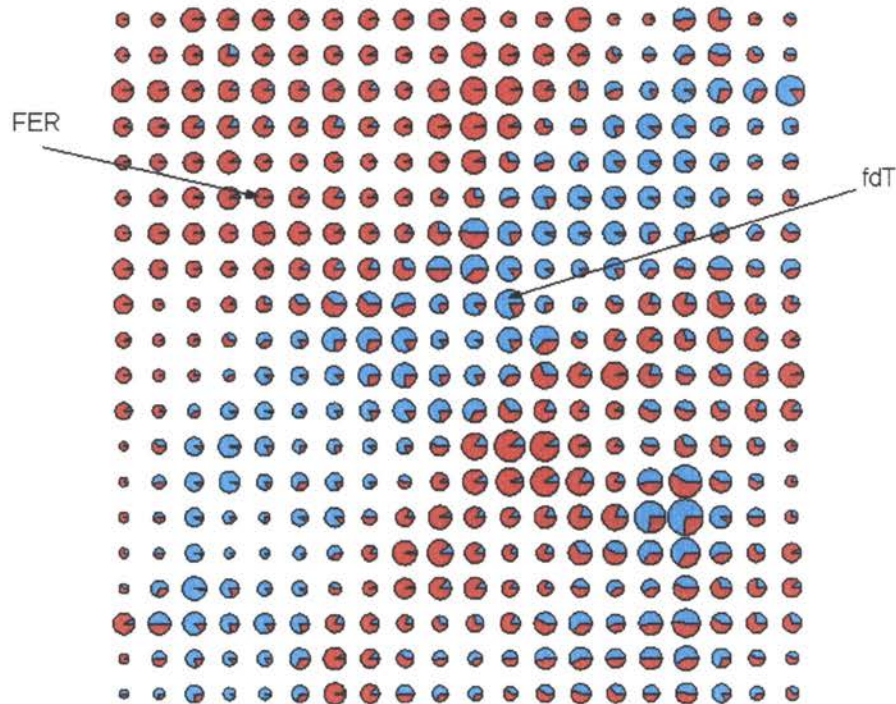
Figure 9.16 Relative Importance of CDMA parameters FER and *fdT* - Male
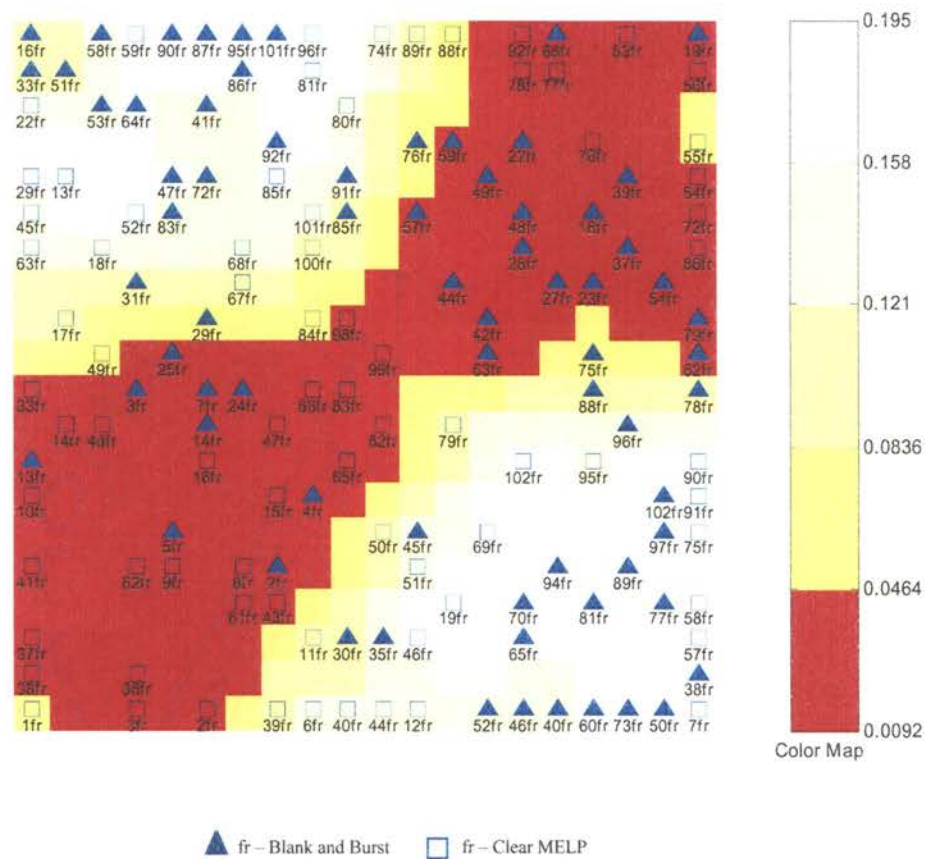
Figure 9.17 FER Color Map: Clear MELP vs. Secure Blank and Burst – Male

The diagonal across the center from the upper right to the lower left in *fdT* color

map, figure 9.18, represents the direction of rapid change of the *fdT* parameter. However,

cross referencing with the PESQ color map, figure 9.12, the rapid change on the *fdT*

parameter map does not yield any notable affect on quality scores. There are values on

either side of the diagonal of rapid change of *fdT* that remain the same. For example,

Secure Blank and Burst simulated data points quality scores remain poor, i.e. *41fr* to *75fr*,

and *31fr* to *45fr* remain fair from either side of diagonal. This signifies that the *fdT*

parameter does not have a big affect on quality scores as the other parameters. The

correlation in the fading process, which this parameter represents, is not strong enough to have a significant contribution. Evidently, the correlation in the fading process of CDMA networks is much less than the correlation in burst loss of the Internet, represented by *clp*.
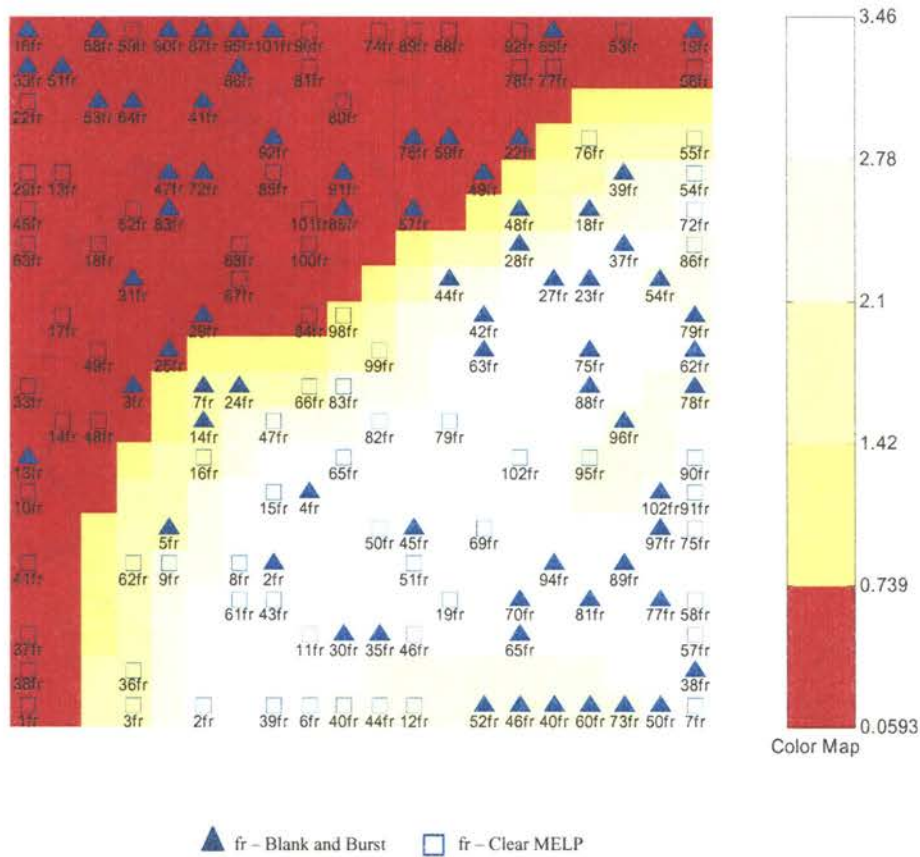
Figure 9.18 *fdT* **Color Map: Clear MELP vs Secure Blank and Burst – Male**

Ave SD_fw QoS Maps: Clear MELP vs Secure Blank and Burst – Male

The figures below are included to show the results of the average spectral distortion analysis of Clear MELP and Secure Blank and Burst modes. There is a very distinctive clustering structure that separates the Clear MELP and Secure Blank and Burst data sets, figure 9.19.
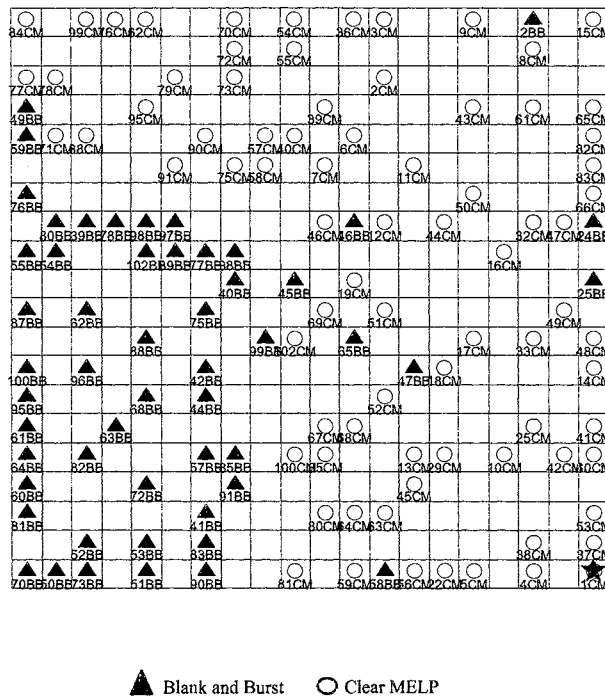
Figure 9.19 SD$_{fw}$ QoS Map: Clear MELP vs. Secure Blank and Burst – Male

The unit distance map shows the distinctive cluster separation as the Secure Blank

and Burst data set is clustered in the lower left corner of the map. The cluster is separated

from the Clear MELP data as indicated by the large distance values as high as 1.14,

shown in figure 9.20. Although the Clear MELP data appear to cover the majority of the

upper right side of the map, the low distance values within the cluster (figure 9.20)

compared to the Secure Blank and Burst data cluster distances allude to the closeness of

the data with respect to quality score.
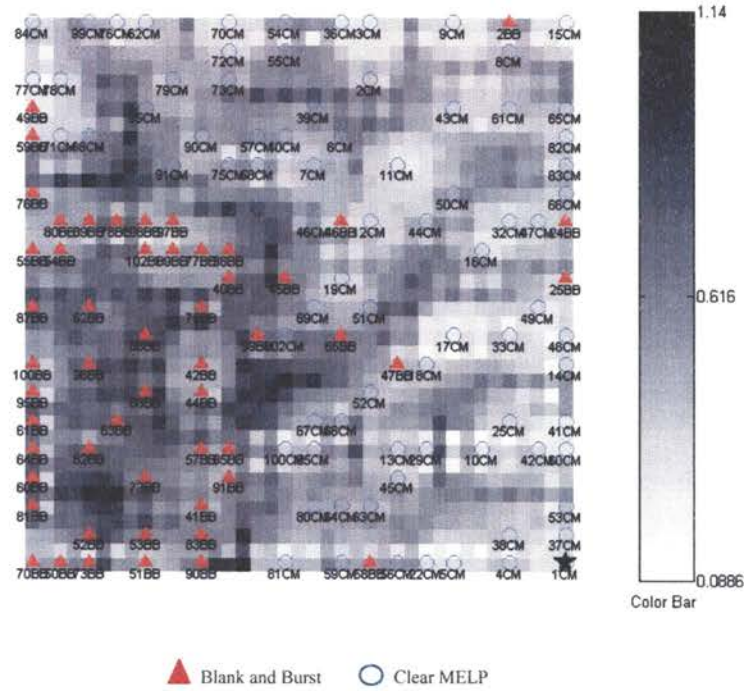
Figure 9.20 SD$_{fw}$ U-Matrix QoS Map: Clear MELP vs. Secure Blank and Burst – Male

The average spectral distortion and percent outlier color maps show the quality values for the data sets, figures 9.21 and 9.22. The Clear MELP values are well within the transparent region for the SD$_{fw}$ as indicated by figure 9.9 above, although the map does not have the color resolution to show it. On the other hand the majority of the Secure Blank and Burst mode simulation data cluster falls outside the transparent (SD$_{fw}$ < 1dB) and acceptable region (percent outliers < 4%).
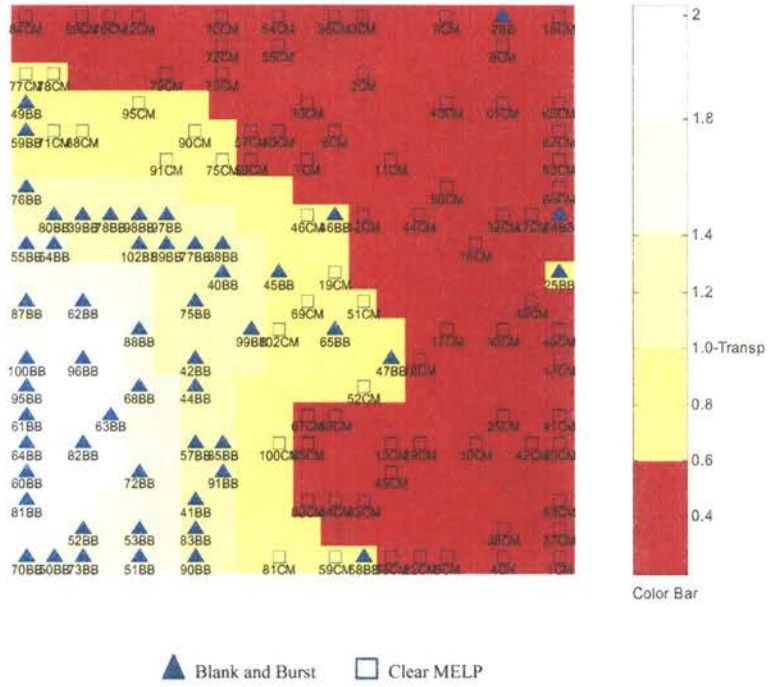
Figure 9.21 SD$_{fw}$ Color Map: Clear MELP vs. Secure Blank and Burst – Male
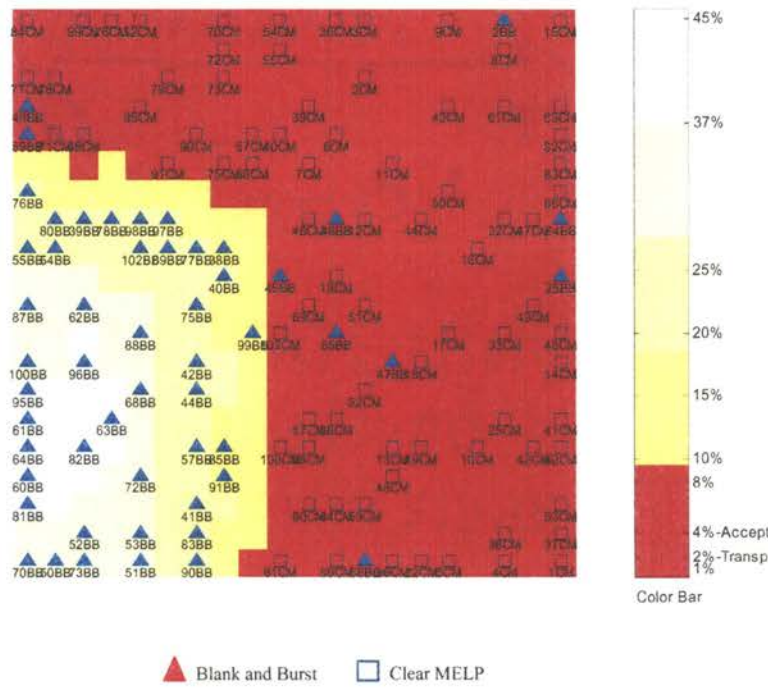
Figure 9.22 Percent Outlier Color Map: Clear MELP vs. Secure Blank and Burst – Male

The above analysis of FNBDT system quality shows that secure FNBDT communication quality is significantly degraded due to sync loss caused by the loss of sync management frames in error prone Internet and CDMA data networks. FNBDT is particularly susceptible to burst loss, resulting in poor speech quality. In the next section adaptive forward error correction is employed to improve secure FNBDT performance due to sync loss.

FNBDT SMF Forward Error Correction Analysis

This analysis of forward error correction of FNBDT sync management frames using adaptive feedback loss only considers synchronization loss[9] as a result of the loss of sync management frames. This is done to isolate the effects of sync loss due to FNBDT sync management frames. It also allows the illustration of system improvement relative to the recovery of FNBDT/MELP frames within recovered superframes with FEC. An average sync loss rate is calculated and feedback is sent to the transmitting system every 5 seconds (RTCP feedback time). Based on the sync loss rate a minimum number of redundant packets containing sync management frames are calculated to yield the lowest possible sync loss rate. The transmitter system then simulates sending the calculated number of redundant packets within the packet stream. The receiver system attempts to recover lost sync management frames from redundant copies when packets containing sync management frames are deemed lost by the network models.

---

[9] Sync loss is due to the number of MELP frames in a superframe the can not be decoded (de-crypted) without the encryption information contained in sync management frames.

Experiment Setup

The experiments involving the forward error correction of FNBDT sync management frames requires files of longer duration than those recommended by PESQ. The FEC implementation required feedback of sync loss at 5 second intervals in accordance with the RTCP specification for packet loss rate feedback in IP applications. Therefore, male and female speech files of 60 second duration are used and the corresponding output speech quality is measured with the frequency weighted spectral distortion measure. The 60 second speech files have three talkers (male or female) with 5 active speech segments per talker. The duration of the speech segments and silence between active speech segments meet the specifications described above with PESQ. Speech files were digitized in a quiet background environment.

Sync Loss Improvement with FEC

A statistical simulation is performed to show the improvements to FNDT sync loss under varying networks packet loss rate conditions. The figures below show the results of simulations 450 seconds in length. Loss rates with and without FEC (labeled before and after FEC reconstruction on graphs) are calculated and displayed.

154
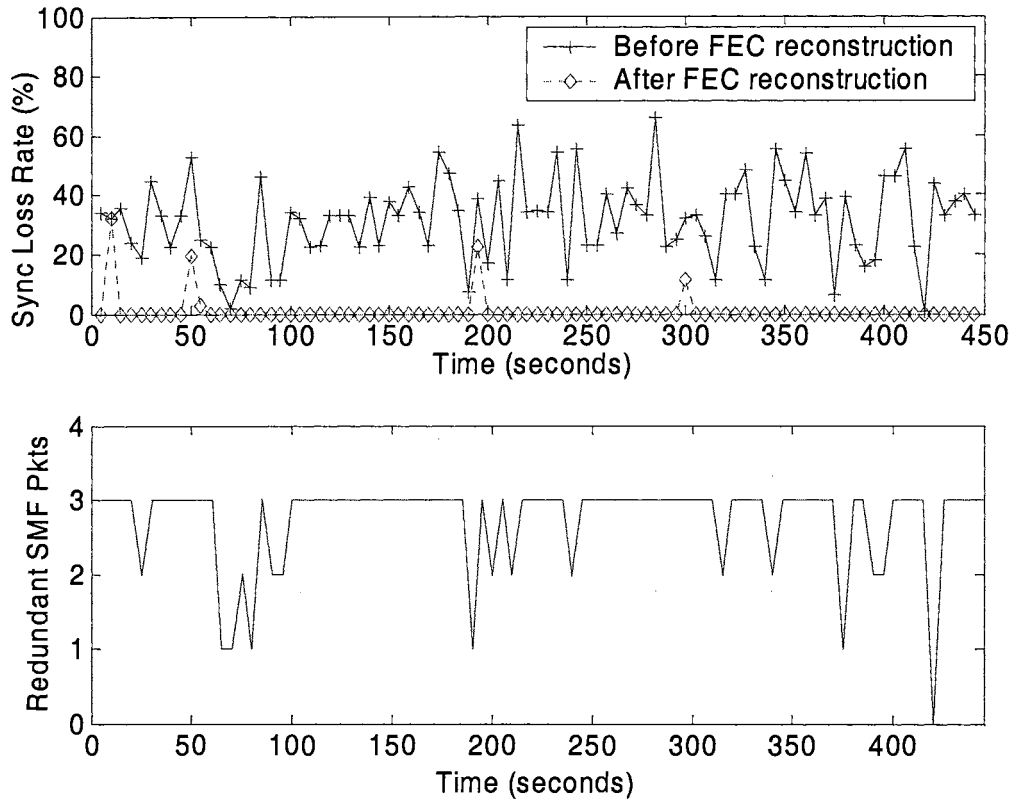
Figure 9.23 Adaptive FEC of SMF for Bandwidth Conservation.

Figure 9.23 reveals an overall average sync loss rate before and after FEC (with

and without FEC) can significantly reduce sync loss rate using adaptive FEC while

conserving bandwidth. The graph shows for practically every instance of feedback the

loss rate was reduced to zero. At time interval 285 seconds the sync loss rate was reduced

from approximately 65% to 0%, using 3 redundant packets. At feedback intervals 70, 75

and 80 seconds the sync loss rates were 10%, 8%, and 48%, and the adaptive FEC

scheme successfully reduced the sync loss rates to 0%, by adaptively selecting 2, 1, and 3

redundant packets. The maximum bandwidth increase with 2,400 bps Secure Blank and

Burst mode is 12% or 2,728 bps for 3 redundant sync frames. Table 9.5 shows the

corresponding bandwidth increase per number of redundant sync frames.

Table 9.5 Bandwidth Increase for Adaptive FEC of Sync Management Frames.

Number of Redundant Sync Management Frames	BW	BW Increase
0	2,400 bps	0 %
1	2,508 bps	4.3%
2	2,616 bps	8.25%
3	2,728 bps	12%

At 12% bandwidth increase the total bandwidth or data rate of 2,728 bps is well below

modern communication standards, particularly with IP networks. Even with the addition

of overhead (IP packet headers) this adaptive FEC scheme will satisfy bandwidth

requirements over lower bandwidth IP compatible networks such as IS-95 CDMA

cellular data, which has channel bandwidths as low as 9600 bps.

There is an added delay tradeoff using burst spacing for adaptive FEC. Depending

on the amount of redundancy used and thus the spacing of redundant copies, an

equivalent of ½ to 1/4th of a superframe of delay is required to recover from sync loss.

For example, if one redundant copy is needed, according to the burst spacing, the

redundant copy will be placed in the middle of the superframe after MELP frame 12 (of

24 MELP frames per superframe). If 2 frames are included per packet, 6 packet times of

delay (45ms + network delay) will expire before sync can be regained using the

redundant copy of the sync management frame. At any rate, the sync loss saving out

weigh the additional delay in cases were sync management frame 1 or 2 are recovered. If

sync management frame 1 is lost, sync is lost for 3 superframes, because sync

management frames 2 and 3 depend on sync management frame 1 to decrypt the

successive superframes of data. Likewise, if sync management frame 2 is lost, sync is lost

or 2 superframes. Therefore, the recovery of sync in these cases results in recovery of 1

second (2 superframes) to 1.5 seconds (3 superframes) of voice data.
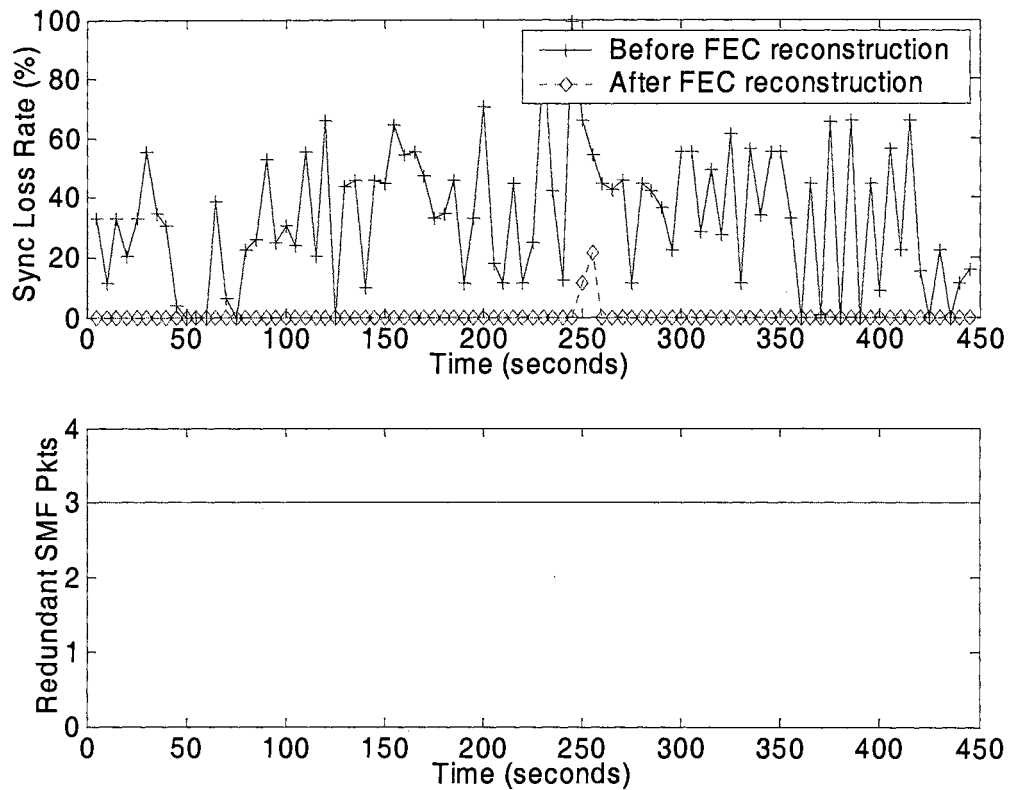
Figure 9.24 FEC with Fixed 3 Redundant SMFs, Spaced to Avoid Burst Loss.

Figure 9.24 shows how the FEC algorithm performs with high sync loss rates

using 3 fixed redundant sync management frame packets. The average sync loss rate

without FEC is approximately 40% with values as high as 98%. The fixed 3 redundant

frame FEC implementation is successful in reducing the sync loss rate to 0% in every

case except for intervals 250 and 255 seconds. However, the sync loss rates were reduced

by 88% and 40%, respectively.

The adaptive FEC algorithm with burst spacing of redundant sync management

frames make it robust to burst loss. In figures 9.25 and 9.26 the results of consecutive

redundant packets and burst spacing are shown for the same simulation. Each simulation

uses the same number of redundant packets at precisely the same time intervals. The only

difference is the spacing of redundant frames. The results however are drastically

different, in that the burst spacing of redundant packets reduces the loss rates 5 times

more than with consecutive spacing. Nevertheless, figure 9.25 reiterates how burst loss

can degrade performance even with FEC. However, figure 9.26 shows how robust the

adaptive FEC algorithm with burst spacing is to burst loss.
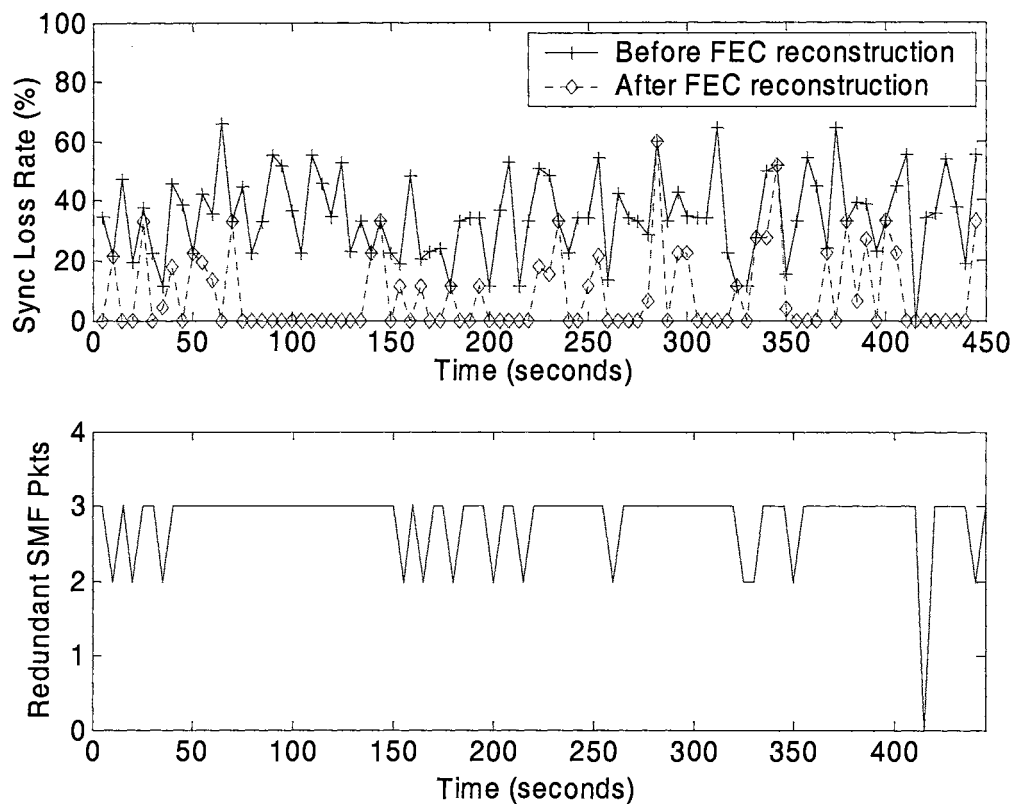
Figure 9.25 Adaptive FEC of SMF with Consecutive FEC Frames – Subject to Burst Losses.
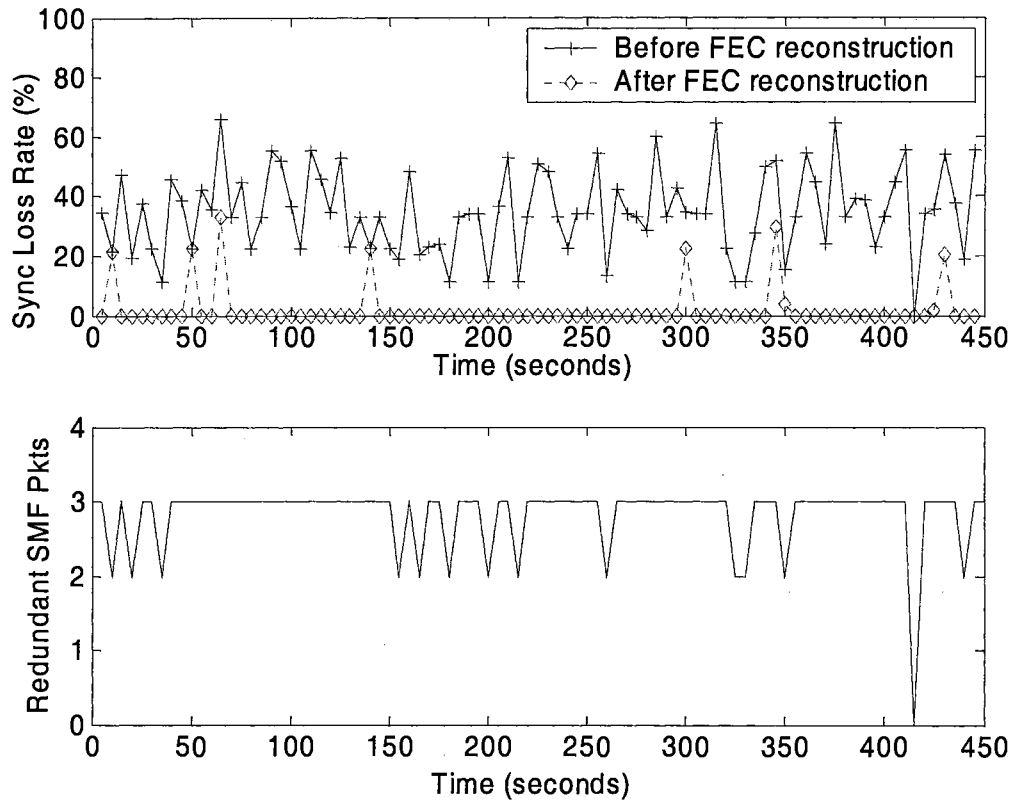
Figure 9.26 Adaptive FEC of SMF with Spacing of FEC Frames to Avoid Burst Losses.

FNBDT Secure Blank and Burst Mode with vs. without FEC

Adaptive FEC has the greatest potential for quality improvement of secure

FNBDT modes. Adaptive FEC with burst spacing is tested for quality improvement for

raw FNBDT. Simulations are conducted with Secure Blank and Burst FNBDT mode over

varying concatenated network models parameters, for 60 second male and female speech

files. The overall simulated network model output loss rate and the corresponding quality

measurement, SD_{fw} for female talkers is shown in Figure 9.27 below. The corresponding

male simulation results can be found in the Appendix.
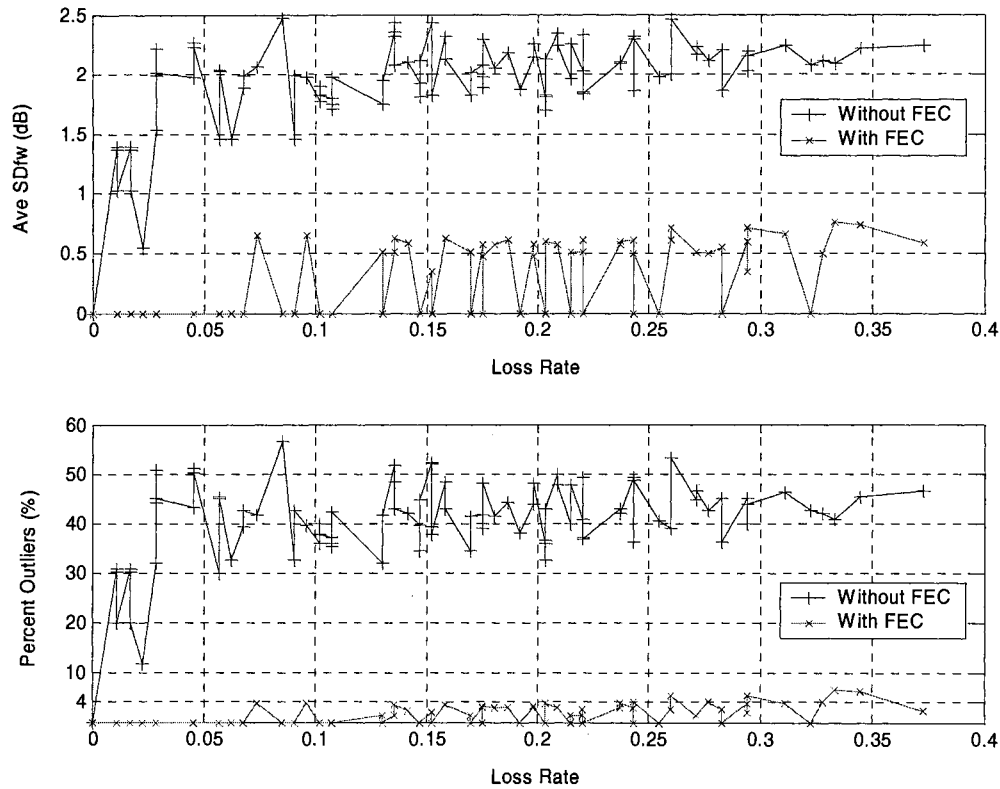
Figure 9.27 Ave SD$_{fw}$ and Percentage Outlier Frames vs. Loss Rate for FNBDT with FEC and FNBDT without FEC – Female.

Figure 9.27 shows that without FEC quality immediately becomes unacceptable with average spectral distortion above 1 dB at a loss rate of 0.02 and reaching 2 dB beyond 0.04. The percentage of outlier frames go well above 4% reaching 40% beyond a loss rate of 0.04. However, acceptable quality is maintained for adaptive FEC, SD$_{fw}$ < 1 dB and PO < 4% to a loss rate of approximately 0.32. FEC reduces the spectral distortion by an average of 1.5 dB and the percentage of outlier frames by an average of 40%, to achieve acceptable quality.

Ave SD$_{fw}$ QoS Maps: FNBDT with FEC vs FNBDT without FEC – Female

The QoS maps for Secure Blank and Burst mode with and without FEC are provided in figures 9.28 – 9.31 for simulations with female speech files. The figures show the relative clustering of the data set with FEC and without FEC. The FEC data set represented with hollow squares, cluster on the left side of the map while the raw Blank and Burst data set clusters on the right side of the map. The SD$_{fw}$ and percent outlier quality measure's colors map, figures 9.29 and 9.30 show the areas on the left side of the maps have quality values within the acceptable range, encompassing the FEC data set. The right side of the figures has quality values well outside the acceptable quality range where the non FEC data set reside. Finally the unit distance map, figure 9.38, has large distance values between the FEC and the non FEC data set, concurring the separation in quality. In conclusion, the QoS map and related maps confirm that FEC improves the quality of the Secure Blank and Burst modes.
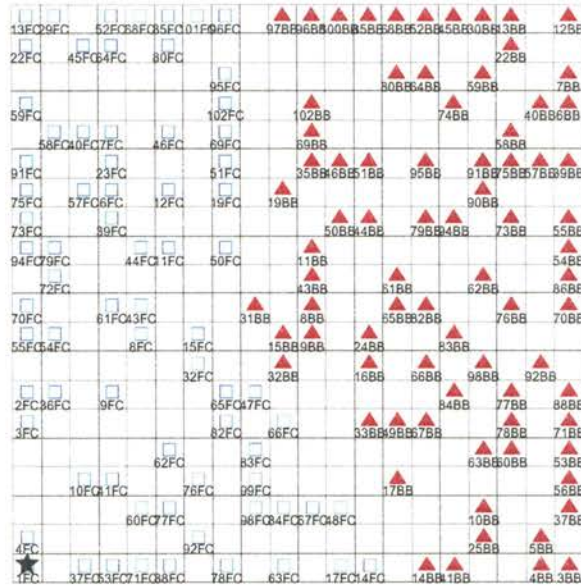
BB – Blank and Burst   FC – FEC Blank and Burst

Figure 9.28 SD$_{fw}$ QoS Map: FNBDT with and without FEC – Female.
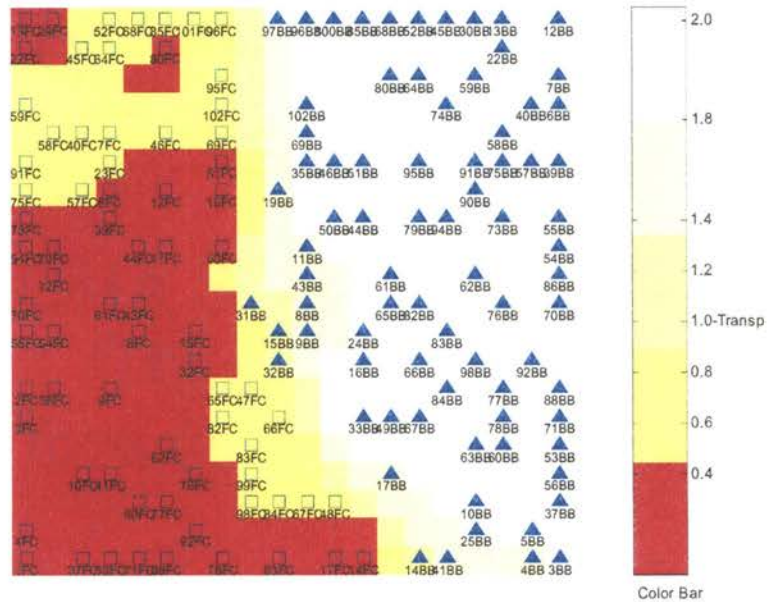
BB – Blank and Burst   FC – FEC Blank and Burst

Figure 9.29 SD$_{fw}$ Color Map: FNBDT with and without FEC – Female.

162

▲ BB – Blank and Burst   ☐ FC – FEC Blank and Burst
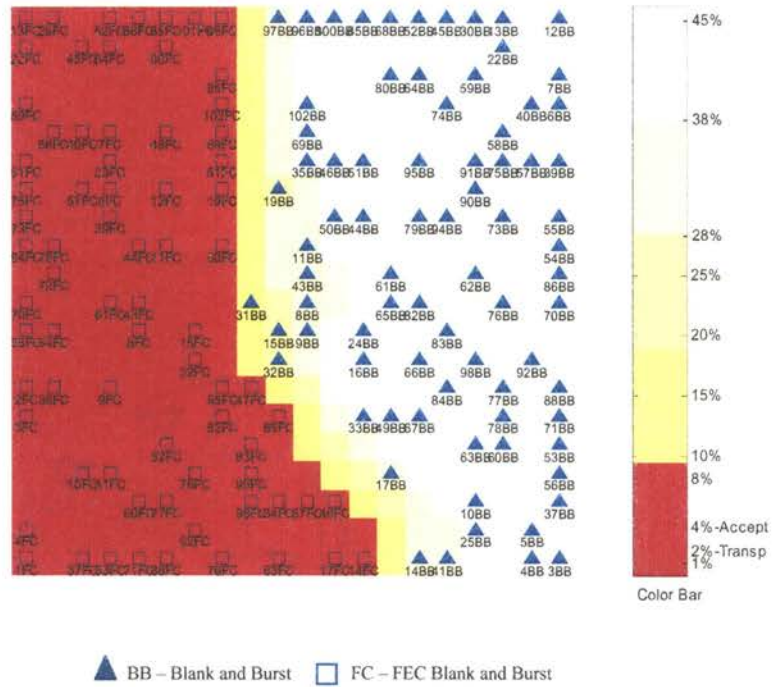
Figure 9.30 Percent Outliers Color Map: FNBDT with and without FEC – Female.

▲ BB – Blank and Burst   ☐ FC – FEC Blank and Burst
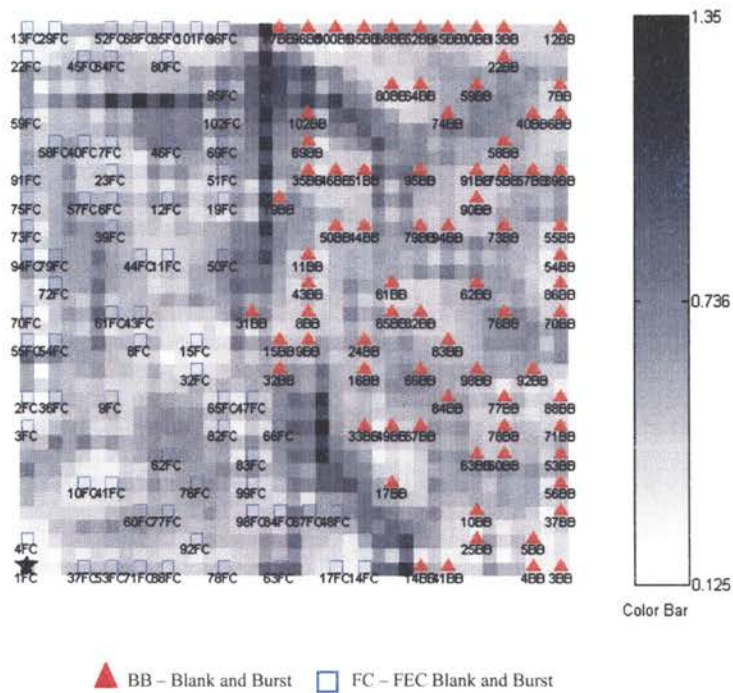
Figure 9.31 U-Matrix: FNBDT with and without FEC – Female.

163

Jitter Buffering Evaluation

The goal of this buffering evaluation is to show how jitter buffering can improve FNBDT system quality in packet network environments. This analysis will show how the addition of jitter buffering, both fixed and adaptive, can reduce packet loss, while minimizing overall system delay, thus increasing system QoS. Adaptive jitter buffering requires the receiver audio playback buffer to increase or decrease based on the measured received packets inter-arrival delays. Correspondingly, the audio data contained in the playback buffer is either stretched or compressed to maintain continuous audio play-out to accommodate the buffer adjustment. The compressing and stretching of the audio data can result in degradation in output audio quality, and is an area of ongoing research [107] [108]. During active speech segments if the buffer increase is too large (resulting in the stretching of audio), the resulting audio may have long silent period or sustained sounds within words or speech segments. On the contrary, large reduction is playback buffer size can result in elimination of large parts of words or conversation. Nevertheless, this research focuses on the delay and packet loss saving capability of jitter buffering and will not consider the audio quality as a result of the buffer adjustment. The performance of adaptive buffer adjustment on audio quality is discussed in chapter 10 for future work. This analysis assumes that audio data is perfectly stretched and compressed in active voice regions. The emphasis is on the effects of loss due to inadequate buffering and the delay savings possible with jitter buffering.

Three receiver system jitter buffering scenarios are considered: no receiver buffer, fixed buffer, and an adaptive buffer. "No buffering" at the receiver provides insight on how a raw FNBDT system would perform in a packet environment, while fixed and

adaptive buffering shows potential improvements to system performance with the addition of buffering. The above buffering schemes are used with the network simulator that models network jitter based the new multi-structure network delay jitter model. The buffering schemes are tested with model parameters that reflect measurements of mean and standard deviation in network jitter recorded from trace runs using *NetPerf* for Internet, and concatenated Internet and CDMA connections. Mean and standard deviation values use in this evaluation is given in table 9.6.

Table 9.6 Jitter Model Parameters for Jitter Buffer Evaluation

Jitter Model Parameters						
Mean	45	45	48	52	53	81
Stdev	1	5	59	69	84	208

Simulations used the 8 male/female speech files described above based on the PESQ recommendation. During simulations packets, containing speech frames, are considered late when the inter-arrival delay for the packet is greater than the scheduled playback time for the packet. The scheduled playback time for the "no buffering" case is a function of the MELP frame interval (2*22.5ms=45ms, where 2 is the number of frames per packet and 22.5ms is MELP's frame interval). The fixed buffer will include a buffer length or playback time of 90 ms or 2 packets intervals [6]. The adaptive buffer uses the adaptive playback buffering algorithm to calculate the playback time based in incoming packet inter-arrival delays. Late packets are considered lost and the equivalent amount silence is substituted at the audio output. The output audio quality is measured using PESQ.

Jitter Buffering Analysis using QoS mapping

Parameter vectors consisting of mean, standard deviation and PESQ score for the no buffering, fixed buffering and adaptive buffering simulations (male and female) are used to train a QoS map neural network. The resulting QoS map of the jitter buffering data set is shown in Figure 9.32.

Figure 9.32 QoS Map Jitter Buffering Evaluation.

The spatial location on the map represents the overall QoS level. In the lower left corner of the map is the reference quality point, which represents the highest possible

quality value (no error case) and is represented with a star. As the map points move away

from the reference point the corresponding quality values decrease.
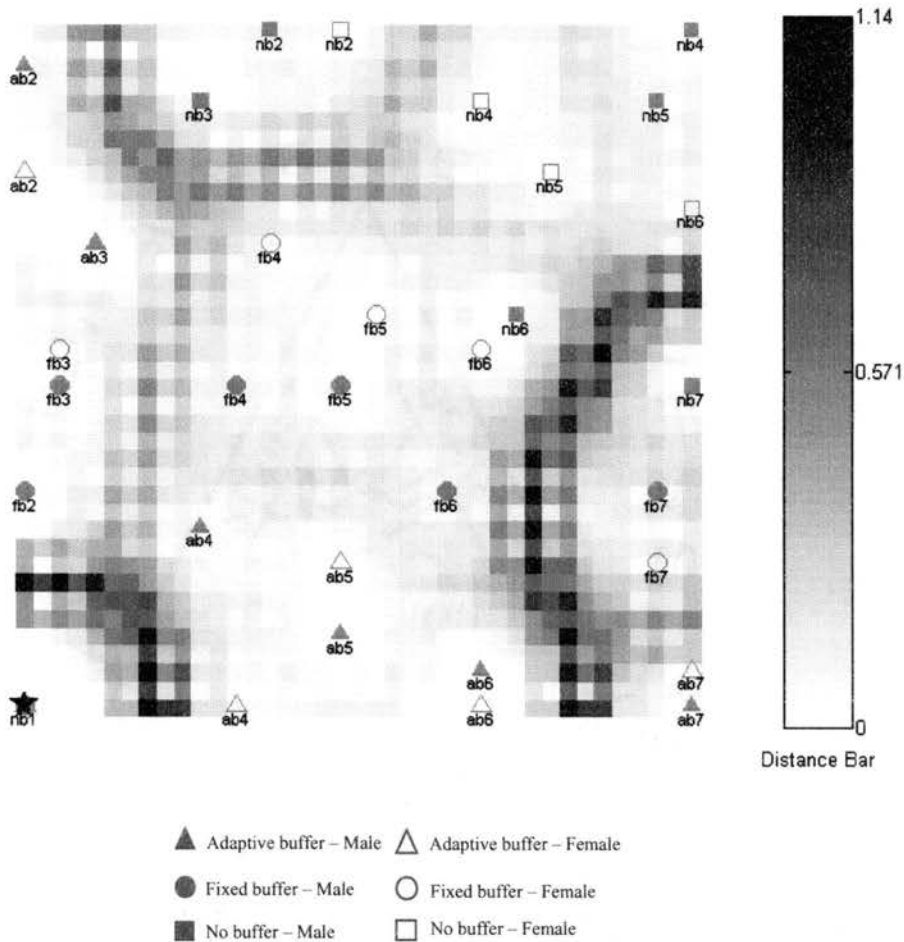
Figure 9.33 Unit Distance Map with Distance Color Bar.

The unit distance map, figure 9.33, show there are some definite cluster structures

for the different buffering schemes on the distance map indicated by large distance

values. The no buffering data is clustered in the upper right corner, the fixed buffering

data in the center diagonal, and the adaptive buffering is spread along the lower left-

center of the map. Although clusters appear close spatially on the map the distance,

167

particularly as it relates to quality, in some cases are quite large. For example, in the

upper left corner of the unit distance map the adaptive buffering case labeled *ab2* is in

close proximity to the no buffering case *nb3*, however the unit distance between them is

large and is reflected in the PESQ quality scores of 3.2 (good quality for *ab2*) and 1.6

(poor quality for *nb3*).

On the PESQ color map, figure 9.34, the direction of increasing quality is from

the lower left corner to the upper right corner of the map. It can be seen from the values

on the map that the fixed buffer achieved better quality over no buffering, and the

adaptive buffering achieve higher quality over fixed buffering. The addition of adaptive

buffering moves the voice quality from poor to good. Overall this map shows the how

jitter buffering can significantly improves FNBDT system performance.
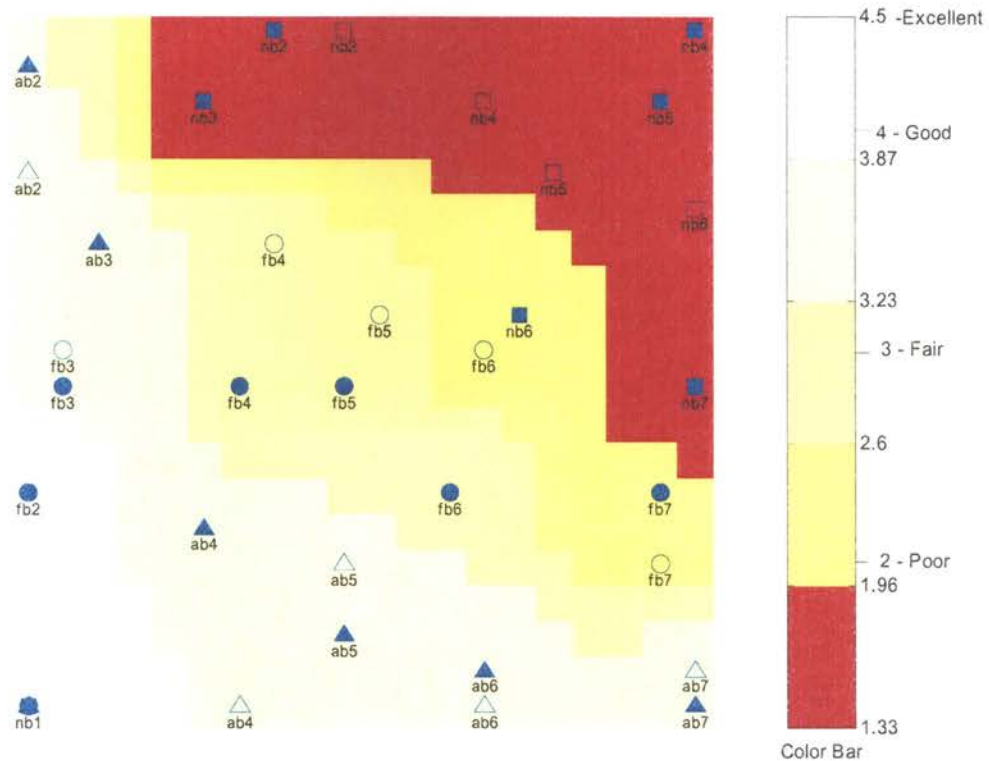
Figure 9.34 PESQ Color Map with Color Bar.

Figures 9.35 and 9.36 are the color map for the mean and standard deviation parameter values. Figure 9.35 shows that the direction of increasing mean delay is from left to right. Looking at both figure 9.34 and 9.35, the mean delay and PESQ score has an inverse relationship with quality. As the mean jitter delay increases, the quality deceases. Standard deviation indicates the level of burstiness of jitter delay. Figure 9.36, for standard deviation show the same trend, the direction of increased standard deviation results in decrease in quality. Therefore, the burstiness of delay causes decrease in quality. This is particularly apparent in fixed buffering case where as the standard

deviation increases the resulting quality of the fixed buffer decreases, going from

excellent to good to fair to poor.
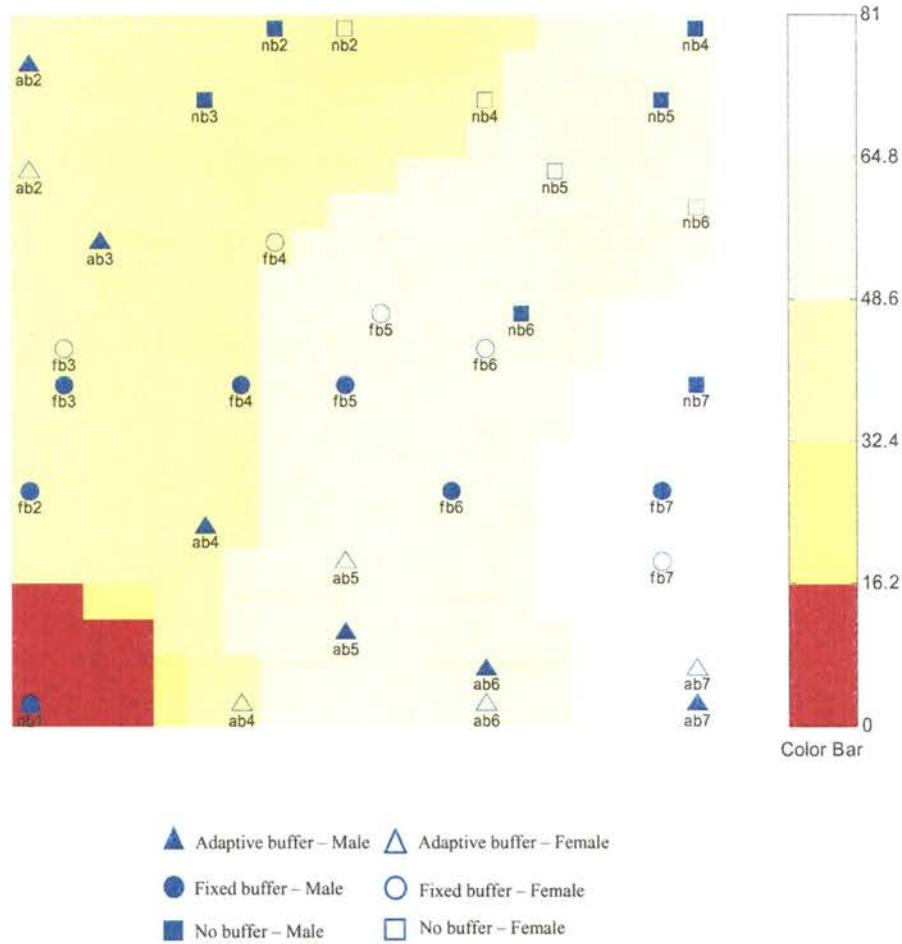
Figure 9.35 QoS Map of Mean Delay with Color Bar.
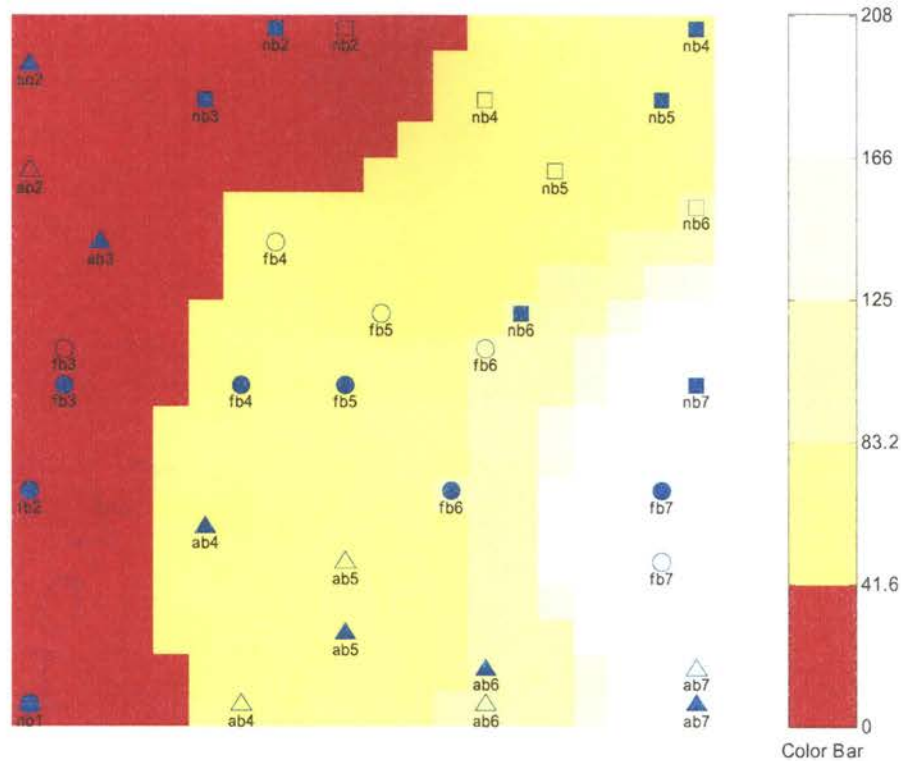
Figure 9.36 QoS Map of Standard Deviation with Color Bar.

To show how each parameter impacts the system quality and the relative placement of the data on the map, the color maps for the individual parameters and the distance pie chart maps is used in conjunction with the PESQ color map. Figure 9.37 shows the pie distance chart for the mean and standard deviation parameters for the jitter buffering analysis.
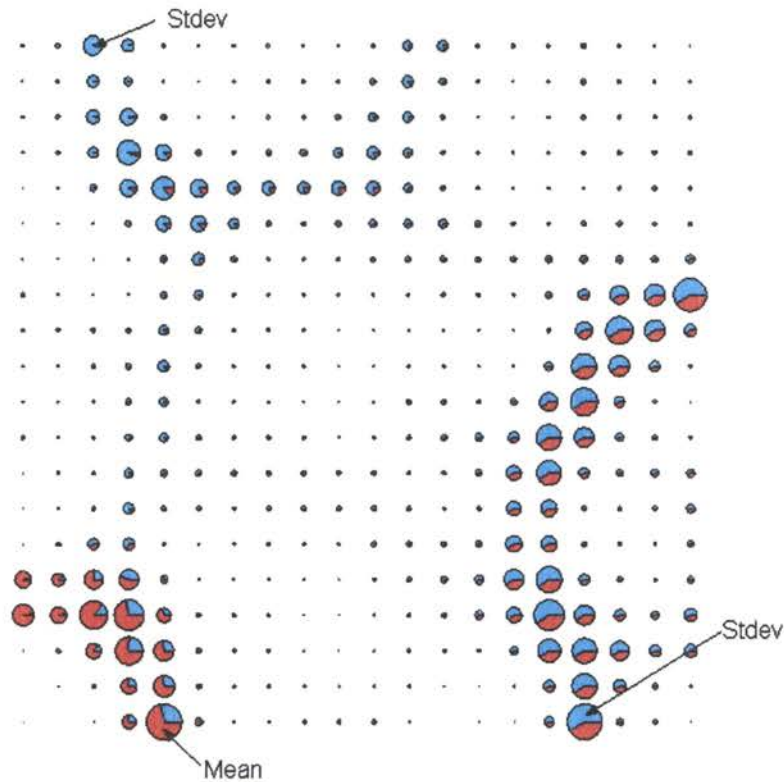
Figure 9.37 QoS Map of Relative Importance of Mean and Standard Deviation.

In figure 9.37 the cluster borders are well defined by the pie charts of mean and standard deviation. Of particular interest is the cluster separation as a result of rapid changes in standard deviation on the right and top of the map. This further reiterates the cluster separation of the low quality - no buffering case from the other schemes. This gives a strong indication that a raw FNBDT system without jitter buffering is adversely affected by bursty delays. However, the absence of large values of standard deviation around the adaptive buffering clusters, center and lower middle of figure 9.33 shows that adaptive buffering is resilient to bursty delays and is a viable improvement to an FNBDT system.

System Delay Analysis with Jitter Buffering

To provide insight as to the delay performance of the jitter buffering schemes,

figures 9.38 and 9.39 are presented. The figures highlight the fixed and adaptive

buffering for mean jitter of 45 and 81 ms, and standard deviation of 1 and 208, for the

male talker simulation runs. Each illustration shows the jitter model output jitter values

and the corresponding buffer length for each jitter value. Simulated packets with jitter

values greater than the buffer length are considered late and are discarded.
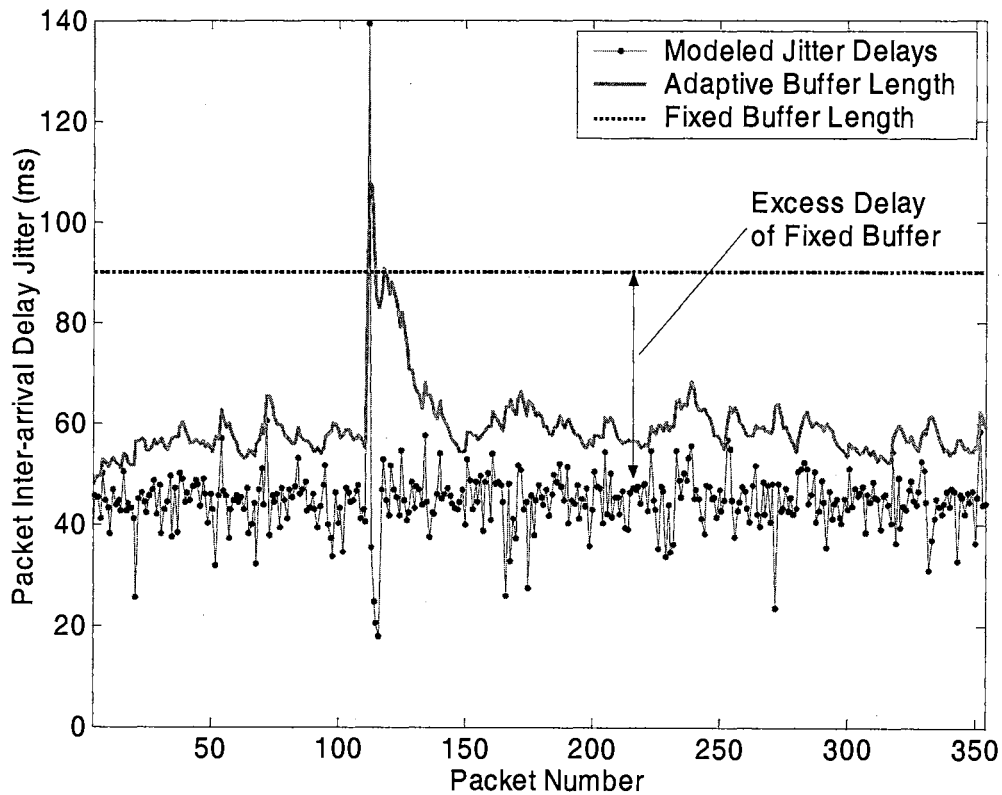
Figure 9.38 Delay Jitter vs. Jitter Buffer Length.

It is apparent from Figure 9.38 that for low network jitter delays, both fixed and adaptive

buffer schemes work well for preventing late loss packets. However, the fixed buffer of

90 ms adds additional delay to the system than required to accommodate the jitter delays.

The adaptive buffer minimizes the overall system delay by saving as much as 40 ms over

the fixed buffer.
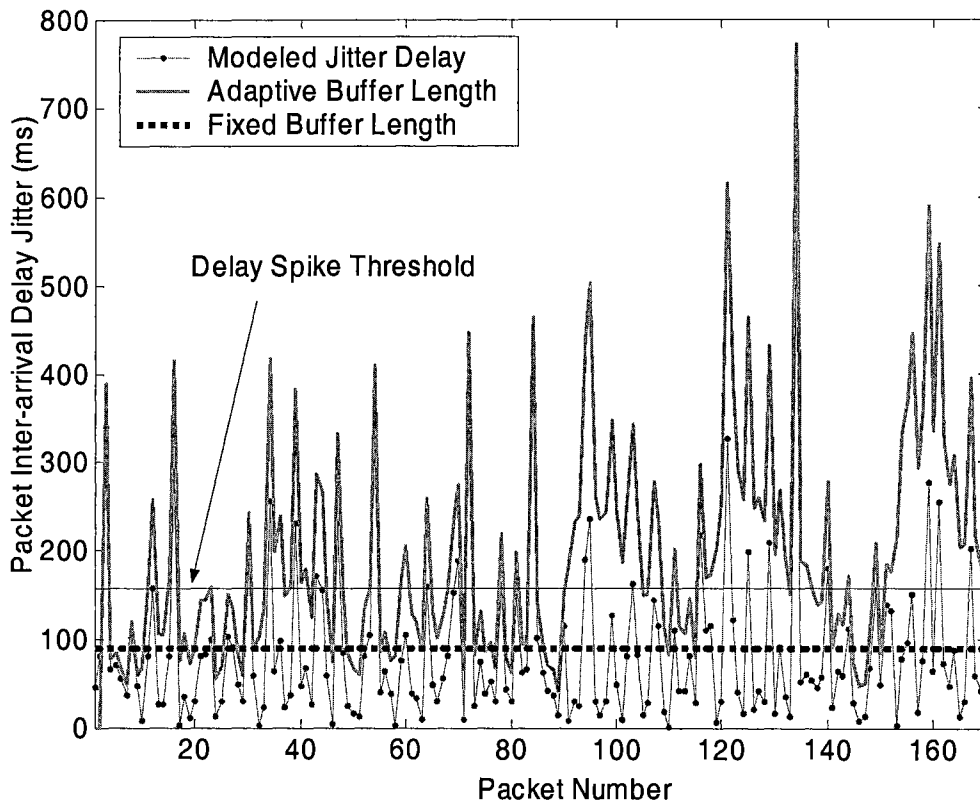
Figure 9.39 Jitter Delay vs. Jitter Buffer Length.

Figure 9.14 represents a more extreme case of jitter delay, recreating the trace run from

LAN to CDMA cellular data with mean delay jitter of 81 ms and standard deviation of

208. Under these jitter conditions, real-time communication would be difficult, however

large buffer lengths of the adaptive buffer enable the conversation to get through in

opposed to the fixed buffer which losses the majority of the packets to late arrivals (based

on buffer length). The total late loss and loss rate for the fixed buffer is 107 out of 354

total packets and 30.22%, respectively. The adaptive buffer adjusts fairly well to the jitter delay trends, even in high burst delay areas, yielding no late losses for the entire trace. To match the packet loss savings of the adaptive buffer, the equivalent fixed buffer length would be approximately 800 ms. A fixed buffer of 800 ms would give favorable late loss results, but the delay savings of the adaptive buffer would go unmatched. Consequently, the additional delay imposed on the overall system would be too great for practical real-time communication.

CHAPTER X

SUMMARY AND FUTURE WORK

Findings

- For Internet and CDMA networks, extremely high packet loss rates are possible (>20%), but lower loss rates are most common (<10%). Packet loss is bursty with the majority of loss burst consisting of one to two packets. Packet inter-arrival delay jitter is bursty with delay bursts reaching as high as 1 second. Finally, out-of-order packet delivery seldom occurs.

- Clear MELP data streams that experience degraded quality due to packet loss can achieve quality improvement with packet loss recovery schemes. Frame replication yields better speech quality results than silence substitution. Informal listening test results showed acceptable quality was achieved at high loss rates of 20% with frame replication for both male and female speakers. However, performance degrades to unacceptable at extremely high loss rates (>20%). MELP is robust to burst loss at low loss rates with frame replication because of MELP's internal parameter smoothing. Informal listening tests for male and female speakers returned similar results. Informal listening tests agreed with results returned by the spectral distortion calculation, thus re-enforcing the validity of MELP quality measurements.

- Secure FBNDT (Secure Blank and Burst Mode) exhibits poor overall quality compared to Clear MELP mode in error prone mixed networks. Secure modes experience sync loss due to loss of sync management frames that results in unacceptable speech quality. Increases in packet loss rates (ulp) for the Internet and frame error rates (FER) caused by multi-path fading for CDMA decrease overall FNBDT quality. Bursty packet loss of the Internet is the major cause of synchronization loss and thus has the greatest impact on quality degradation. However, for CDMA channels the correlation in the fading process due to Doppler frequency shift experienced by the mobile unit for stationary to driving speeds (fdT values of 0.02 to 3.5), does not contribute significantly to the degradation in quality. Simulation results showed that changes over the range of the normalized Doppler frequency parameter, fdT, had no effect on the resulting output speech quality. Bursty packet inter-arrival delay jitter degrades the quality of a raw FNBDT system without sufficient buffering. Finally, simulations with male and female speakers yield similar results.

- Enhancement to Secure FNBDT Blank and Burst mode yields improved system performance. Adaptive jitter buffering improves system quality from poor quality to good quality according to MOS PESQ scores. Adaptive jitter buffering reduces packet loss due to inadequate buffering over fixed and no buffering, while minimizing overall system delay. Adaptive Forward Error Correction with rate control feedback significantly reduces sync loss through the recovery of sync

177

management frames. Sync loss is reduced at extremely high loss rates, >20%, by as much as 88%. Burst spacing of redundant sync management frames (distributing redundant copies with equal spacing within superframes) showed resilience to bursty packet loss, the major cause of sync loss. Adaptively sending redundant frames conserves bandwidth, with a maximum bandwidth increase of 12%. Overall adaptive forward error correction of sync management frames for Secure Blank and Burst achieves quality result comparable to Clear MELP. Male and female simulations yield similar results.

- Probabilistic models including the Gilbert model for Internet packet loss, fading channel model for CDMA channels, the new multi-structure delay jitter, and the FNBDT sync loss model capture the characteristics of the networks and FNBDT. The Gilbert model is commonly used to model Internet packet loss because of its simplicity and ease of implementation. It sufficiently models the temporal dependency of packet loss of the Internet. The CDMA fading channel model is based on Rayleigh distributed channel fading of CDMA air links from base station to mobile host. It captures the error pattern of CDMA channels including loss burstiness. The new multi-structured jitter model captures the multi-structure of packet inter-arrival jitter by modeling the short and large time scale jitter delays using a Laplacian and Gaussian distributions, respectively. To achieve the correlation between delay values, an exponential decaying function is used to model the delay spike phenomena found in measure network delays. The modeled jitter delays follow mean and burst properties of "real-life" collected jitter delay

values. Finally, the FNBDT sync loss model sufficiently models the loss of sync (MELP voice frames) when simulated network packet loss results in loss of sync management frames.

Recommendations

The favorable results of the proposed FNBDT enhancement techniques make them a natural choice as recommendations for a FNBDT system deployed over mixed packet networks. In addition to the proposed enhancements, there are others that may be applicable to an FNBDT implementation. This section highlights a few ideas for improving FNBDT performance in addition to the enhancements already presented in this dissertation.

For an FNBDT/MELP implementation, frame repetition is a good candidate for loss recovery in IP network environments. The analysis in Chapter 9 of Clear MELP using frame repetition to recover packet loss gave favorable results over silence substitution. However, when loss rates were high, frame repetition and silence substitution separately did not yield acceptable quality according to informal listening tests, because silence substitution produces "pops" and abrupt silence between active speech, while the continued replication of packets performed by frame replication proved annoying to listeners. Therefore, a good compromise between simple frame duplication and silence substitution proposed by [98] is frame duplication with successive fading. This technique involves fading the replicated frame over a time period to gradually transition into a silent or comfort noise period. This alternative on average would provide better performance over abrupt insertion of silence or continued replication of packets during sync loss with secure FNBDT modes. According to [98], frame replication

gives acceptable quality, and frame replication with successive frame fading gives better quality.

Interpolation also has potential for improved quality in the wake of packet loss. Interpolation adds additional system complexity, but improves quality [98] over replication techniques. Sridhara and Fuja [93] demonstrated in their evaluation of MELP's performance to cell loss over ATM networks that frame interpolation provides improvement in spectral distortion over frame replication and silence substitution. These findings also make frame interpolation a good prospective candidate for this implementation.

Retransmission is yet another form of error correction that may be applicable to sync management frame recovery. If a sync management frame, in particular the SMF1[10], loss is detected, this scheme simply sends out a request for the retransmission of the lost sync management frame and the transmitter retransmits the lost sync management frame. Under normal circumstances of a real-time voice communication application, this scheme will not suffice because of the delay it imposes on the system. However, with the unique case of sync management frames, particularly SMF1, if lost, three superframes of data will be lost, which translates to 1.5 seconds of audio data. If the sync management frame (SMF1) can be recovered, FNBDT can potentially recover (decrypt) at least the second and third superframe of data, gaining one second worth of recovered data. This is assuming the loss of SMF1 can be detected, retransmitted, and received within one superframe or approximately 0.5 second. This analogy can also apply to the loss of

---

[10] The encryption counter is divided into 3 consecutive sync management frames, with sync management frame 1 being the first of 3 divided counter values. Each consecutive sync management frame depends on the preceding sync management frame for the continuation of its counter to decrypt the data.

SMF2, which can potentially reduce the overall loss from one second worth of speech to approximately 0.5 second.

Under conditions where loss rates are extremely high, synchronization will be hard to maintain making communication difficult with an FNBDT system, particularly in secure modes. Such conditions require drastic adjustments such as resynchronization, adding additional delay by buffering to avoid congestion, or simply dropping the connection and reconnecting. In applications where getting the message through takes priority over real-time communication, a TCP connection can be established to utilize its guaranteed throughput property with the expectation of high delays.

When attempting to compensate for out-of-order packets there are several things to consider. The number of packets that can be corrected is directly proportional to the additional delay that is introduced to the system. In order to correct out-of-order packets, additional buffers are required to hold and resequence packets as they arrive from the network. Figure 10.1 shows a buffer organization scheme for reordering out-of-order packets for a packet-voice receiver taken from [109].
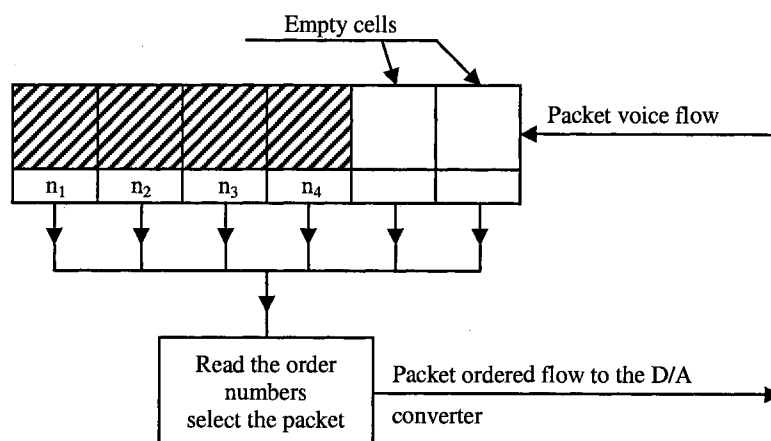
Figure 10.1 Buffer Reorganization Scheme with Reordering Mechanism at the Buffer Output.

To resequence n packets, $n+1$ additional buffers are needed. The additional delay added to the overall system is τ_{delay}, with Δ_{Pkt}, as the packet accumulation time (10.1).

$$\tau_{delay} = (n+1) * \Delta_{Pkt} \qquad (10.1)$$

When determining the number of out-of-order packets to correct, the trade-off of delay added by resequencing out-of-order packets versus the quality of reproduced speech as a result of the packet loss recovery scheme must be considered. For example, if the loss recovery technique produces acceptable speech quality for n lost packets, correcting for only n out-of-order packets would yield, at most, an incremental increase in audio quality; at the expense of adding a significant $n+1$ packet length delay to the overall system delay. Therefore, the number of out-of-order packets to correct should be more than the amount of lost packets the system can recover with reasonable quality. The trade-off of delay versus improved audio quality in correcting for out-of-order packets often falls in favor of the delay savings. For this reason, it is recommended that out-of-order packets are not recovered. If a packet is received out-of-order, it should be regarded as lost and the appropriate packet loss recovery technique be used.

Future Work

Silence vs. Talkspurt Playback Time Adjustment

The adaptive jitter buffering algorithm used in this research provides a means of estimating network delay to adjust the audio playback time. Audio playback is adjusted without regard to audio data within the buffer. Typically, applications that apply jitter buffer adjustments, considering the audio data within the buffer, use talkspurt detection

and only adjust playback time during silence periods. In order to adjust the playback time

for talkspurts the speech signal must be extended or compressed to accommodate the

increase or reduction in playback time; this must be done without reducing output speech

quality, figure 11.1. However, recent studies have applied adaptive schemes that adjust

playback times for both talkspurts and silence periods [107] [108]. They contend that

allowing playback adjustment only during silence periods prevents the application from

dealing with large delay spikes that occur within talkspurts, so packets are lost because

the playback time cannot be adjusted until the next silence period.

The algorithm Liang *et al.* ([107]) proposes adjusts playback time for each

individual packet that requires extension and compression of the audio signal. This is

done using a time-scale modification technique referred to as the Waveform Similarity

Overlap-Add (WSOLA) algorithm. He reports that this method does not degrade audio

quality, but significantly improves the trade-off between buffering delay and late loss.
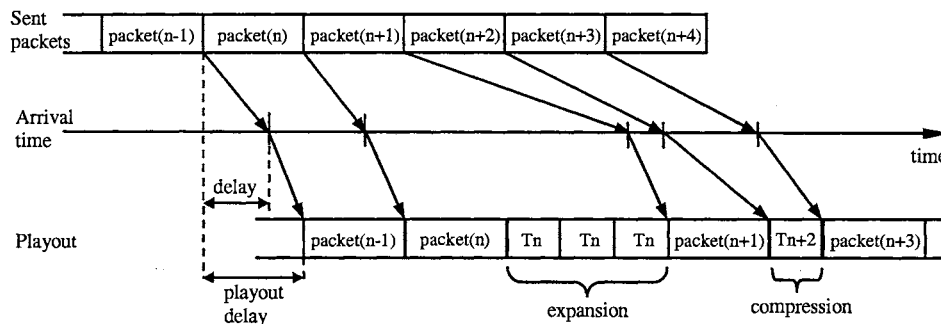
Figure 10.2 Expansion and Compression of Playout Voiced Signal. *Tn* and *Tn+2* Denotes the Pitch Lag for Packet *N* and *N+2*, Respectively [108].

Anandakumar *et al.* [108] allows playback delay to increase during both silence

and talkspurts, but playout decreases only during silence periods. Increasing playback

delay during talkspurts also requires extension of speech. They accomplish this using a voicing-classification based speech extension algorithm for code-excited linear predictive (CELP) coders. Their speech extension algorithm is performed on the parameter-level at the decoder and involves replication of parameters and the generation of the excitation signal for the extension speech. They attest that their algorithm improves system quality by reducing packet loss due to delay spikes during talkspurts and by not decreasing playback delay during talkspurt avoids audio quality degradation. The CELP based extension algorithm seems a likelier candidate, with modification, for this FNBDT implementation using MELP (with voice activity detection). It is simpler to implement and provides a means to maintain system quality during jitter buffer adjustments.

Additional Simulation Study

Simulations can be conducted with varying packet sizes and send rates to monitor the effects on network performance; with the intention of investigating if larger packet sizes (slower send rates) will improve network performance during congested periods. By sending more frames per packet, increasing packet size and sending packet less often because of frame accumulation time, can congested periods of the network be avoided? Are larger packets more network efficient? Will they have a higher probability of getting through network nodes than smaller ones? Simulations will provide answers to these question which may lead to system improvements. For example, if the FNBDT receiver system detects high packet losses, it can signal the transmitter to slow down its send rate by buffering up more voice frames (increasing packet size), thus avoiding the congestion period, preventing packet loss.

Additional simulations for the network models are also warranted. A random number generator is being used to condition the probability of packet loss for all network models, Internet burst loss model (Gilbert model), CDMA fading channel model, multi-structured network delay jitter model, and FNBDT sync loss model. To improve simulation using the random number generator, multiple different seeds for each conditioned probability can be used for the respective loss models. The average throughput, distortion calculation, etc. for the simulation will be used as characteristic for that probability. This will account for the wide range of packet loss types and loss patterns for the respective loss probabilities. Also, voice activity detection can be incorporated into packet loss model simulations, by calculating packet loss only during active speech. It can be assumed that loss during silence periods is imperceptible. Therefore, during simulations packets would enter the loss state only during talkspurts and non-loss state during non-losses and silent frames.

Additional Application of QoS Mapping

There are several applications of a QoS map in a FNBDT/VoIP application. For example, in a "real-life" implementation, if a quality measurement is made on the input data, the distance measured from a reference point can determine the current level of QoS for the system. If the QoS level of the receiver system is not adequate based on the relative placement on the QoS map indicated by distance from a reference quality point, an adaptive technique can be initiated by the system in an attempt to improve the overall quality. This process of measuring QoS and system adaptation can continue until the desired QoS level is attained.

Other Applications

Although this work entails the analysis of FNBDT using voice data (MELP), the ultimate objective of FNBDT is to provide secure multi-media communications. A natural extension of this work would be to explore different media types, such as video, data, fax, etc. Several aspects of this research have particular applications for interactive video applications. To conserve bandwidth, video codecs typically send a key frame that contains an entire video image, followed by difference frames which provide updates the key frame based on changes in the scene. In most cases the changes in the scene are small and the difference in successive frames is minimal. Less information bits are required and bandwidth is minimized. Video key frames have the same impact on system quality as sync management frames with FNBDT. If a key frame is lost, then subsequent difference frames are unusable and sync is lost until the receipt of the next key frame. An adaptive FEC scheme such as presented in this research has the potential to improve the performance of an interactive video application such as teleconferencing or distance learning using video.

Other data formats such as data and fax do not have the constraints of real-time voice or video applications. Many of the challenges of this research are not relevant. For example, data (documents, emails, text messages) do not have the delay constrains of real-time voice. However, data integrity is important, so a more reliable transport protocol such as TCP can be used. TCP guarantees throughput with potentially longer delays. So, the loss recovery techniques and buffering issues presented in this research are not relevant.

FNBDT is intended to be interoperable over different networks. So, different

network types (ATM, ISDN, PSTN, etc.) should be explored as they have different

characteristics and error types. Models of other networks can be investigated to determine

their effects on FNBDT. Other wireless standards should be explored (GSM, GPRS,

Wireless LAN, Satellite, etc.) as well as other speech coding algorithms. Various other

types of terminal equipment can be considered for testing and demonstration of an

FNBDT system, for example, the QSEC-800 cellular CDMA phone with FNBDT

capability developed by Qualcomm. Additional terminal equipment might include

wireless PDA's or hand held computing devices.

CHAPTER XI

CONTRIBUTION AND CONCLUSIONS

Research Contribution

The contributions made by this research are as follows:

1) Identified packet network limitations by conducting a literature search in conjunction with collection of "real-life" network statistics.

2) Projected network limitations on the FNBDT system using network performance models based on network statistics. Developed a new multi-structured network delay jitter model and a new FNDT synchronization loss model. Designed a network simulator application in MATLAB® that included FNBDT simulation.

3) To evaluate the performance of FNBDT/MELP the following quality measures were introduced: the frequency weighted distortion measure, informal listening tests, and the perceptual evaluation of speech quality (PESQ) measure. A QoS mapping technique was also introduced to analyze (Secure and Clear) FNBDT performance over mixed packet networks, enabling the analysis of a multi-dimensional parameter space.

4) Proposed FNBDT system quality enhancements for quality improvements including frame replacement strategies for packet loss, Forward Error Correction (FEC) for recovery of FNBDT encryption synchronization loss, and an adaptive delay jitter buffering scheme to accommodate network delay jitter.

5) Evaluated the performance of raw FNBDT system performance as well as FNBDT system performance with enhancements in mixed Internet and CDMA

cellular data networks. Showed performance improvements to MELP using

packet loss recovery techniques and FNBDT improvement by using adaptive jitter

buffering, and an adaptive FEC scheme that minimizes bandwidth and is resilient

to burst loss.

6) Finally, identified performance issues, determined feasibility, and contributed to

formulation of recommendations necessary for designing a more robust FNBDT

and other secure (and non-secure) voice-over-packet-network applications.

Conclusions

FNBDT represents a radical departure from previous government secure

communications schemes. FNBDT is an open signaling plan that is designed to leverage

future, existing, and legacy government and commercial communications networks and

technologies. FNBDT is appropriate for both wireless and wired networks and offers the

flexibility necessary for providing future secure multimedia communications. However,

FNBDT poses several challenges when deployed over packet networks. These challenges

include the FNBDT system response to packet loss, varying packet inter-arrival delays,

out-of-order packets, and others. These network imposed challenges affect the overall

system quality causing degradation in the audio quality, loss of synchronization due to

loss of sync management frames, and late packet loss due to inadequate receiver

buffering. As a result of these challenges experiments show that a raw secure FNBDT

system yields unacceptable communication quality under moderate to high loss rates and

jitter delays.

This report introduced several solutions to these problems, which improved

FNBDT system performance to an acceptable communication level. These solutions

include the use of packet loss recovery strategies for correcting packet loss seen by MELP, adaptive FEC for sync management frames to recover from loss of synchronization, and incorporating an adaptive playback buffering scheme to accommodate packet inter-arrival delay jitter.

This research showed how FNBDT performance is degraded in harsh network environments and how the quality can be improved to an acceptable level. FNBDT is still in its development stages and information gained by this research is important to its future development. The information gained from this research provides the recommendations necessary for designing a more robust FNBDT system.

CHAPTER XII

REFERENCES

[1] M. E. Dark, R. A. Dean, "Security Architecture for Tactical/Strategic Interoperability", *Proceedings of the 1994 IEEE Tactical Communications Conference*, pp. 245-254, Vol. 1, 1994.

[2] J. S. Collura, "Noise Pre-Processing for Tactical Secure Voice Communications", *NATO Tactical Mobile Communications Symposium TMC-99*, Lillehammer Norway.

[3] J. S. Collura, D. J. Rahikka, "Interoperable Secure Voice Communications in Tactical Systems", *Speech Coding for Algorithms for Radio Channels (Ref. No. 200/012), IEE Seminar*, pp 7/1 – 7/13, April 2000.

[4] L. Supplee, R. Cohn, and J. Collura, "MELP: The New Federal Standard at 2400bps," *IEEE International Conference on Acoustic, Speech, and Signal Processing ICASSP'97*, pp. 1591-4, 1997.

[5] E. J. Daniel, K. A. Teague, "Federal Standard 2.4k bps MELP over IP," *Proceedings of the 43rd IEEE Midwest Symposium on Circuits and Systems MWSCAS '2000*, Vol. 2, pp 568-571, Aug. 2000.

[6] E. J. Daniel, *A Voice over IP Framework and Simulation for Low Rate Speech and the Future NarrowBand Digital Terminal*, Oklahoma State University, Stillwater, OK, July 2000, M.S. Thesis.

[7] CCITT Yellow Book, *Recommendation* G.721, Vol. 3.

[8] J. Deller, J. Proakis and J. Hansen, *Discrete-Time Processing of Speech Signals*, Macmillan Publishing Company, New York, p. 410, 1993.

[9] A. McCree and T. Barnwell, "A Mixed Excitation LPC Vocoder Model for Low Bit Rate Speech Coding," *IEEE Transactions of Acoustics, Speech, and Signal Processing*, Vol. 3, No. 4, pp. 242-50, July 1995.

[10] A. V. McCree, T. P., III, Barnwell, "Implementation and Evaluation of a 2400 bit/s Mixed Excitation LPC vocoder," *1993 IEEE International Conference on Acoustics, Speech, and Signal Processing ICASSP-93*, Vol. 2, pp. 159-162, 1993.

[11] A. V. McCree, T. P., III, Barnwell, "A 2400 bps Mixed Excitation LPC Vocoder," *Conference Record IEEE Military Communications Conference MILCOM'92*, Vol. 1, pp. 381-384, 1992.

[12] D. Griffin and J. Lim, "Multiband Excitation Vocoder," *IEEE Transactions of Acoustics, Speech, and Signal Processing,* Vol. 36, No. 8, pp. 1223-35,August 1988.

[13] D. Griffin, *Multiband Coder,* MIT, Cambridge, MA, 1987, Ph.D. Dissertation.

[14] J. C. Campbell, T. E. Tremain and V.C. Welch, "The Federal Standard 1016 4800 bps CELP Voice Coder," *Digital Signal Processing,* Vol. 1, 1991.

[15] J. Hardwick and J. Lim, "A 4800 bps Multiband Excitation Speech Coder," *Proceedings of ICASSP-88,* pp. 374-7, April 1988.

[16] K. Teague, B. Leach and W. Andrews, "Development of a High-Quality MBE Based Vocoder for Implementation at 2400bps," *Proceedings of IEEE Wichita Conference on Communications, Networking and Signal Processing,* April 1994.

[17] T. E. Tremain, "The Government Standard Linear Predictive Coding Alogrithm: LPC-10," *Speech Technol.,* pp. 40-49, Apr. 1982.

[18] T. Unno, T. P. Barnwell III, and K. Truong, "An Improved Mixed Excitation Linear Prediction (MELP) Coder," *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing,* Vol. 1, pp 245-248, 1999.

[19] J. S. Collura, D. F. Brandt, D. J. Rahikka, "The 1.2Kbps/2.4Kbps MELP Speech Coding Suite with Integrated Noise Pre-Processing," *Proceedings of IEEE Military Communications Conference MILCOM 1999,* Vol. 2, pp 1449-1453, 1999.

[20] T. Wang, K. Koishida, V. Cuperman, and A. Gersho, J. S. Collura,"A 1200 BPS Speech Coder Based on MELP," *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing ICASSP'00,* Vol. 3, pp 1375-1378, 2000

[21] D. Rahikka, T. E. Fuja, T. Fazel, "Enhanced Error Correction of the US Federal Standard MELP Vocoder Employing Residual Redundancy for Harsh Tactical Applications," *NATO Tactical Mobile Communications Symposium TMC-99,* Lillehammer, Norway.

[22] J. S. Collura, "Speech Enhancements and Coding in Harsh Acoustic Noise Environments," *IEEE Workshop on Speech Coding Proceedings,* pp 162-164, 1999

[23] D. J. Rahikka, J. S. Collura, T. E. Fuja, T. Fazel, "US Federal Standard MELP Vocoder Tactical Performance Enhancement Via MAP Error Correction," *Proceedings of IEEE Military Communications Conference MILCOM '99,* Vol. 2, pp 1458-1462, 1999.

[24] GTE Government Systems Corp., "*FNBDT Signaling Plan*", Rev. 1.0, Dec. 1998.

[25] A. H. Levesque, R. A. Dean, "Secure Communications Interoperability In Mixed Analog and Digital Networks", *Proceedings of IEEE Military Communications Conference MILCOM'91*, pp.303-307, Vol. 1, 1991.

[26] J. Walrand, *Communication Networks: A First Course*, Richard D. Irwin Inc., and Aksen Associates, Inc., USA, pp.46-51, 1991.

[27] A. Duke, personal communication, GTE Government Systems Corp, August 2001.

[28] D. Su, J. Srivastava, and J-H Yao, "Investigating Factors Influencing QoS of Internet Phone" *IEEE International Conference on Multimedia Computing and Systems*, pp. 308-313, 1999.

[29] T. Yensen, M. Parperis, I. Lambadaris, and R. Goubran, "Determining Acoustic Round Trip Delay for VoIP Conferences", *IEEE Second Workshop on Multimedia Signal Processing*, pp. 161-166, 1998.

[30] M. S. Borella, A. Sears, J. A. Jacko, "The Effects on Internet Latency on User Perception of Information Content," *IEEE Global Telecommunications Conference*, Vol. 3, pp. 1932-1936, Nov. 1997.

[31] T. J. Kostas, M. S. Borella, I. Sidhu, G. M. Schuster, J. Grabiec, and J. Mahler, "Real-Time Voice Over Packet-Switched Networks," *IEEE Network*, pp. 18-27, Jan/Feb 1998.

[32] M. Borella, D. Swider, S. Uludag, and G. Brewster, "Internet Packet Loss: Measurement and Implication for End-to-End QoS," *Proc. IEEE ICPP Workshop '98*, pp. 3-12, 1998.

[33] D. Bertsekas, R. Gallager, *Data Networks*, Prentice-Hall, New Jersey, 1992.

[34] J. Walrand, P. Varaiya, *High-Performance Communication Networks*, Morgan Kaufmann Publishers, San Francisco, CA, 2000.

[35] T. S. Rappaport, *Wireless Communications, Principles and Practices*, Prentice-Hall, New Jersey, 1996.

[36] J. C. Bolot, "Characterizing End-to-End Packet Delay and Loss in the Internet", *Journal of High Speed Networks*, pp. 305-323, Vol. 2, 1993.

[37] M. Yajnik, S. Moon, J. Kurose, and D. Towsley, "Measurement and Modeling of the Temporal Dependence in Packet Loss," *Proceedings of IEEE Conference of the IEEE Computer and Communications Societies INFOCOM '99*, pp. 345-352, June 1999.

[38] M. Khansari, V. Bhaskaran, "A low-complexity error-resilient H.263 coder", *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp 2737-2740, 1997.

[39] D. Sanghi, A. K. Agrawala, O. Gudmundsson, B. N. Jain, "Experimental assessment of end-to-end behavior on Internet," *Proceedings of INFOCOM 1993*, Volume 2, pp 867-874, March 1993.

[40] M. A. Kohler, *Compensation for Missing Voice Coding Frames in Packet Transmission Systems*, Oklahoma State University, Stillwater, OK, May 2000, Ph.D. Dissertation.

[41] V. Paxson, "End-to-End Internet Packet Dynamics", *IEEE/ACM Transactions on Networking*, pp 277-292, Vol. 7, No. 3, June 1999.

[42] N. F. Maxemchuk and S. Lo, "Measurement and Interpretation of Voice Traffic on the Internet", *IEEE International Conference on Communications ICC'97*, Vol. 1, pp. 500-507, Aug. 1997.

[43] M. Arai, A. Chiba, K. Iwasaki, "Measurement and Modeling of Burst Packet Losses in Internet End-to-End Communications", *Proceedings of IEEE Pacific Rim International Symposium on Dependable Computing*, pp 260-267, 1999.

[44] B. J. Dempsey and Y. Zhang, "Destination Buffering for Low-Bandwidth Audio Transmission using Redundancy-Based Error Control," *Proceedings of the 21st IEEE Conference on Local Computer Networks*, pp. 345-354, May 1996.

[45] J. C. R. Bennett, C. Partridge, N. Shectman, "Packet Reordering is Not Pathological Network Behavior", *IEEE/ACM Transactions on Networking*, Vol. 7, No. 6, pp 789-798, December 1999.

[46] S. Floyd, V. Paxson, "Difficulties in Simulating the Internet", *IEEE/ACM Transactions on Networking*, Vol. 9, Issue 4, pp 392-403, Aug. 2001.

[47] W. E. Leland, M. S. Taqqu, W. Willinger, and D.V. Wilson, "On the Self-Similar Nature of Ethernet Traffic (Extended Version)," *IEEE Transactions on Networking*, Vol.2, No. 1, pp. 1-15, Feb 1994.

[48] Z. Sahinoglu, S. Tekinay, "On Multimedia Networks: Self-Similar Traffic and Network Performance," *IEEE Communications Magazine*, pp. 48-52, January 1999.

[49] M. E. Crovella, A. Bestavros, "Self-Similarity in World Wide Web Traffic: Evidence and Possible Causes", *IEEE/ACM Transactions on Networking*, Vol. 5, No. 6, pp 835-846, December 1997.

[50] V. Paxson, S. Floyd, "Wide-Area Traffic: The Failure of Poisson Modeling", *IEEE/ACM Transactions on Networking*, Vol. 3, No. 3, June 1995.

[51] M. S. Borella, S. Uludag, G. B. Brewster, I. Sidhu, "Self-similarity of Internet Packet Delay", *Proceedings of IEEE International Conference on Communications ICC '97*, Vol. 1, pp 513-517, 1997.

[52] M. S. Borella, G. B. Brewster, "Measurement and Analysis of Long-Range Dependent Behavior of Internet Packet Delay", *Proceedings of IEEE INFOCOM '98*, Vol. 2, pp 497-504, 1998.

[53] Q. Li, D. L. Mills, "On The Long-Range Dependence of Packet Round-Trip Delays In Internet", *Conference Record IEEE International Conference on Communications ICC' 98*, Vol. 2, pp 1185-1191, 1998.

[54] TIA/EIA/IS-95, "Data Services Option Standard for Wideband Spread Spectrum Digital Cellular System," May 1995.

[55] TIA/EIA/IS-99, "Data Services Option Standard for Wideband Spread Spectrum Digital Cellular System," July 1995.

[56] TIA/EIA/IS-707, "Data Services Option Standard for Wideband Spread Spectrum Digital Cellular System," February 1998.

[57] P. Karn, "The Qualcomm CDMA Digital Cellular System", *Proceedings of the USENIX Mobile and Location-Independent Computing Symposium*, pp 35-39, August 1993.

[58] G. Bao, "Performance Evaluation of TCP/RLP Protocol Stack over CDMA Wireless Link", *Wireless Networks*, Vol. 2, no. 3, pp 229-237, 1996.

[59] G. Xylomenos, G. C. Polyzos, "Internet Protocol Performance over Networks with Wireless Links", *IEEE Network*, pp. 55-63, July/August 1999.

[60] W. R. Stevens, *TCP/IP Illustrated*, Vol. 1, Addison Wesley, 1994.

[61] F. Yegenoglu, F. Faris, O. Qadan, "A Model for Representing Wide Area Internet Packet Behavior", *Conference Proceeding of Performance, Computing, and Communications Conference IPCCC '00*, pp. 167-173, 2000.

[62] M. Zorzi, R. R. Rao, L. B. Milstein, "On the Accuracy of a First-Order Markov Model for Data Transmission on Fading Channels", *Proceedings of IEEE ICUPC' 95*, pp. 211-215, Nov. 1995.

[63] K. Thompson, G. J. Miller, R. Wilder, "Wide-Area Internet Traffic Patterns and Characteristics", *IEEE Network*, Vol. 11, Issue 6, pp 10-23, November/December 1997.

[64] V. Paxson, "Empirically Derived Analytic Models of Wide-Area TCP Connections," *IEEE/ACM Transactions on Networking*, Vol. 2, No. 4, August 1994.

[65] G. Crouch, "Considerations for the Specification of IP Performance Objectives," *U.S. contribution to the Feburary 1999 meeting of ITU-T SG 13*, T1A1.3/98-059.

[66] M. Zorzi, R. R. Rao, L. B. Milstein, "Error Statistics in Data Transmission Over Fading Channels", *IEEE Transactions on Communications*, Vol. 46, Issue 11, pp 1468-1477, November 1998

[67] H. S. Wang, "On Verifying The First-Order Markovian Assumption For a Rayleigh Fading Channel Model", *Proceedings of IEEE ICUPC'94*, pp 160-164, San Diego, CA, September 1994.

[68] A. Chockalingam, G. Bao, "Performance of TCP/RLP Protocol Stack on Correlated Fading DS-CDMA Wireless Links", *IEEE Transactions on Vehicular Technology*, Vol. 49, No. 1, pp 28-33, January 2000.

[69] E. N. Gilbert, "Capacity of a Burst-Noise Channel", *Bell System Tech. Journal*, Vol. 39, pp. 1253-1266, September 1960.

[70] J. G. Proakis, *Digital Communications*, McGraw Hill, 2001.

[71] P. M. Soni and A. Chockalingam, "Analysis Of UDP With Link Layer Backoff On Markov Fading Channels," *Tech. Rep. WRL-IISc-TR-102*, IISc, Bangalore, 1999.

[72] E. Chan, X. Hong, "Analytical model for an Assured Forwarding Differential Service Over Wireless Links", *IEEE Proceedings on Communications*, Vol. 148, No. 1, pp 19-23, February 2001.

[73] Y. Y. Kim, S. Li, "Modeling Multipath Fading Channel Dynamics for Packet Data Performance Analysis", *Wireless Networks*, Vol. 6, No. 6, pp 481-492, December 2000.

[74] M. Zorzi, R. Rao, "On Channel Modeling of Delay Analysis Of Packet Communications Over Wireless Links", *36th Annual Allerton Conference on Communications, Control and Computing*, Monticello, IL, September 1998.

[75] E. J. Daniel, K. A. Teague, "Performance of FNBDT and Low Rate Voice (MELP) Over Packet Networks," *Proc. 35th Asilomar Conference on Signals, Systems, and Computers*, Pacific Grove, California, Nov. 4-7, 2001.

[76] H. Lipmaa, P. Rogaway, D. Wagner, "Comments to NIST Concerning AES Modes of Operations: CTR-Mode Encryption", *Contribution to the NIST Modes of Operation Workshop (unpublished)*.

[77] W. Willinger, M. S. Taqqu, R. Sherman, and D. V. Wilson, "Self-Similarity Through High Variability: Statistical Analysis of Ethernet LAN Traffic at the Source Level" *IEEE/ACM Transactions on Networking*, February 1997.

[78] TIA/EIA/IS-2000.2, "Cdma200 Standard for Spread Spectrum Systems", March 2000.

[79] TIA/EIA/IS-2000.3, "Cdma200 Standard for Spread Spectrum Systems", March 2000.

[80] 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 Communicaitons ICC 2001*, Vol. 1, pp. 122-126, 2001.

[81] C. Fulton, S. Li, "Delay Jitter First-Order and Second-Order Statistical Functions of General Traffic on High-Speed Multimedia Networks" *IEEE/ACM Transactions on Networking*, Vol. 6, No. 2, April 1998.

[82] Q. Li, D. L. Mills, "Jitter-Based Delay-Boundary Prediction of Wide-Area Networks" *IEEE/ACM Transactions on Networking*, Vol. 9, No. 5, October 2001.

[83] J. C. Bolot, A. Vega-Garcia, "Control Mechanisms for Packet Audio in the Internet", *Proceedings IEEE INFOCOM '96*, Vol. 1, pp 232-239.

[84] D. Mills, "Internet Time Synchronization: the Network Time Protocol" *RFC-1129*, Internet Engineering Task Force, Oct. 1989.

[85] W. Jiang, H. Schulzrinne, "Modeling of Packet Loss and Delay and Their Effect on Real-Time Multimedia Service Quality", *NOSSDAV 2000*, Chapel Hill, NC, June 2000.

[86] M. Yajnik, S. Moon, J. Kurose and D. Towsley, "Measurement and Modelling of the Temporal Dependence in Packet Loss", *Proceedings of IEEE INFOCOM '99*, New York, Vol. 1, pp 345-352, March 1999.

[87] H. Sanneck, G. Carle, and R. Koodi, "A Framework model for Packet Loss Metrics Based on Loss Runlengths", *In SPIE/ACM SIGMM Multimedia Computing and Networking Conference*, January 2000.

[88] H. Sanneck, L. Le, and A. Wolisz, "Intra-flow Loss Recovery and Control for VoIP," *Proc. ACM Multimedia*, Ottawa, Canada, 2001.

[89] Y. Bai, P. Zhu, A. Rudrapatna, A. Ogielski, "Performance of TCP/IP over IS-2000 Based CDMA Radio Links," *IEEE-VTS Fall VTC 2000*, Vol. 3, pp. 1036-1040, 2000.

[90] Q. Li, D. L. Mills, "Investigating the Scaling Behavior, Crossover and Anti-persistence of Internet Packet Delay Dynamics," *Proceedings of IEEE Globalcom*, Vol. 3, pp. 1843-1852, Rio de Janeiro, Brazil, 1999.

[91] T. Yletyinen, R. Kantola, "Voice Packet Inter-arrival Jitter Over IP Switching," *SBT/IEEE International Telecom Symposium ITS '98*, Vol. 1, pp. 16-21, 1998.

[92] R. Ramjee, J. Kurose, D. Towsley, and H. Schulzrinne, "Adaptive Playout Mechanisms for Packetized Audio Applications in Wide-Area Networks," *Proceedings IEEE Networking for Global Communications INFOCOM '94*, Vol. 2, pp. 680-688, 1994.

[93] D. Sridhara, T. Fuja, "Performance of the Federal Standard 2.4kbps MELP Vocoder Over ATM Networks", *Proceedings of 2000 Conference on Information Science and Systems*, Princeton University, March 2000.

[94] K. Paliwal, B. Atal, "Efficient Vector Quantization of LPC Parameters at 24 Bits/Frame", *IEEE Trans. on Speech and Audio Processing*, vol. 1, no. 1, January 1993.

[95] W. Jiang, H. Schulzrinne, "Modeling of Packet Loss and Delay and Their Effect on Real-Time Multimedia Service Quality", *NOSSDAV 2000*, Chapel Hill, NC, June 2000.

[96] Y. J. Liang, E. G. Steinbach, and B. Girod, "Real-Time Voice Communication over Internet using packet path diversity," *Proc. ACM Multimedia*, Sept/Oct 2001.

[97] D. Sridhara, *Performance of U.S. Federal Standard 2.4 KBPS Speech Coder Over ATM Networks*, University of Notre Dame, Notre Dame, IN, April 2000, M.S. Thesis.

[98] C. Peerkins, O. Hodson, and V. Hardman, "A Survey of Packet Loss Recovery Techniques for Streaming Audio," *IEEE Network*, pp. 40-48, Sept. 1998.

[99] B. W. Wah, X. Su, and D. Lin, "A Survey of Error-Concealment Schemes for Real-Time Audio and Video Transmissions over the Internet," *Proceedings of IEEE International Symposium on Multimedia Software Engineering*, Dec. 2000.

[100] P. Deleon, and C. J. Sreenan, "An Adaptive Predictor for Media Playout Buffering", *Proceedings IEEE International Conference on Acoustic, Speech, and Signal Processing*, pp. 3097-3100, Mar. 1999.

[101] B. J. Dempsey and Y. Zhang, "Destination Buffering for Low-Bandwidth Audio Transmission using Redundancy-Based Error Control," *Proceedings 21st IEEE Conference on Local Computer Networks*, pp. 345-354, May 1996.

[102] A. Kansal, A. Karandikar, "Adaptive Delay Estimation for Low Jitter Audio over Internet," *IEEE Global Telecom Conference GLOBECOM '01*, Vol. 4, pp.2591-2595, 2001.

[103] S. B. Moon, J. Kurose, D. Towsley, "Packet Audio Playout Adjustment: Performance Bounds and Algorithms," *ACM/Springer Multimedia Systems*, Vol. 6, pp. 17-28, Jan. 1998.

[104] J. Postel, editor, "Transmission Control Protocol Specification," *ARPANET Working Group Requests for Comment*, RFC 793, Sept. 1981.

[105] V. Jacobson, "Congestion Avoidance and Control," *Proceedings 1988 ACM SIGCOMM Conf.*, pp. 314-329, Aug. 1988.

[106] D. Mills, "Internet Delay experiments," *ARPANET Working Group Requests for Comment*, RFC 889, Dec. 1983.

[107] Y. J. Liang, N. Farber, and B. Girod, "Adaptive Playout Scheduling using time-scale modification in packet voice communications," *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, Vol. 3, pp. 1445-1448, 2001.

[108] A. K. Anandakumar, A. McCree, and E. Paksoy, "An Adaptive Voice Playout Method for VOP Applications," *IEEE Global Telecom Conference*, Vol. 3, pp. 1637-1640, 2001.

[109] G. Barberis, "Buffer Sizing of a Packet-Voice Receiver," *IEEE Transactions on Communications*, Vol. COM-29, No. 2, Feb. 1981.

[110] L. Sun, and E. C. Ifeachor, "Perceived Speech Quality Prediction for Voice over IP-based Networks," *IEEE International Conference on Communications*, Vol. 4, pp. 2573-2577, 2002.

[111] ITU-T Recommendation P.862. "Perceptual evaluation of speech quality (PESQ), an objective method for end-to-end speech quality assessment of narrow-band telephone networks and speech codecs," Feb. 2001.

[112] A. W. Rix, J. G. Beerends, M. P. Hollier, A. P. Hekstra, "PESQ – The New ITU Standard for End-to-End Speech Quality Assessment," *AES 109th Convention*, Los Angeles, CA, Sept. 2000.

[113] S. K. Das, E. Lee, K. Basu, S. K. Sen, "Performance Optimization of VoIP calls over Wireless Links using H.323 Protocol," *IEEE Transactions on Computers*, Vol. 52, Issue 6, pp. 742-752, June 2003.

[114] W. Stallings, *High-Speed Networks and Internets*, Prentice-Hall, New Jersey, 2002.

[115] ITU-T Recommendation P.800. "Methods for Subjective Determination of Transmission Quality," August 1996.

[116] M. Masugi, "QoS Mapping of VoIP Communication using Self-Organizing Neural Network," *IEEE Workshop on IP Operations and Management*, pp. 13-17, 2002.

[117] T. Kohonen, "The Self-Organizing Map," *Proceedings of IEEE*, Vol. 78, Issue 9, pp. 1464-1480, Sept. 1990.

[118] M. T. Hagan, H. B. Demuth, M. H. Beale, *Neural Network Design*, PWS Pub., Boston, 1996.

[119] J. C. Bolot, S. Fosse-Parisis, D. Towsley, "Adaptive FEC-Based Error Control for Internet Telephony", *Proceedings of IEEE 8th Annual Joint Conference of the Computer and Communications Societies INFOCOM '99*, Vol. 3, pp. 1453-1460, March 1999.

APPENDIX

Supporting Clear MELP vs. Secure Blank and Burst Mode – Male Tables

Table A.1 Clear MELP Mode Simulations - Male

Index	ulp	clp	FER	fdT	mean	stdev	PESQ
1fr	0	0	0	0	x	x	4.5
2fr	0.005	0.1	0.005	3.5	x	x	3.864
3fr	0.005	0.1	0.01	2	x	x	3.864
4fr	0.005	0.1	0.05	0.2	x	x	3.8035
5fr	0.005	0.1	0.1	0.02	x	x	3.421
6fr	0.005	0.1	0.15	3.5	x	x	3.2629
7fr	0.005	0.1	0.2	2	x	x	2.9799
8fr	0.005	0.4	0.005	3.5	x	x	3.9674
9fr	0.005	0.4	0.01	2	x	x	3.9674
10fr	0.005	0.4	0.05	0.02	x	x	3.7864
11fr	0.005	0.4	0.1	3.5	x	x	3.435
12fr	0.005	0.4	0.15	2	x	x	3.2384
13fr	0.005	0.4	0.2	0.2	x	x	2.945
14fr	0.005	0.7	0.005	0.02	x	x	3.9674
15fr	0.005	0.7	0.01	3.5	x	x	3.9674
16fr	0.005	0.7	0.05	2	x	x	3.7864
17fr	0.005	0.7	0.1	0.2	x	x	3.435
18fr	0.005	0.7	0.15	0.02	x	x	3.2334
19fr	0.005	0.7	0.2	3.5	x	x	2.945
20fr	0.01	0.1	0.01	2	x	x	3.864
21fr	0.01	0.1	0.1	0.2	x	x	3.3859
22fr	0.01	0.1	0.15	0.02	x	x	3.1911
23fr	0.01	0.1	0.2	3.5	x	x	2.9799
24fr	0.01	0.4	0.005	2	x	x	3.864
25fr	0.01	0.4	0.01	0.2	x	x	3.864
26fr	0.01	0.4	0.05	0.02	x	x	3.8035
27fr	0.01	0.4	0.1	3.5	x	x	3.3859
28fr	0.01	0.4	0.15	2	x	x	3.2629
29fr	0.01	0.4	0.15	0.2	x	x	3.2629
30fr	0.01	0.4	0.2	0.02	x	x	3.0427
31fr	0.01	0.7	0.005	3.5	x	x	3.9674
32fr	0.01	0.7	0.01	2	x	x	3.9674
33fr	0.01	0.7	0.05	0.2	x	x	3.7864
34fr	0.01	0.7	0.1	0.02	x	x	3.4682
35fr	0.01	0.7	0.2	3.5	x	x	2.945
36fr	0.05	0.1	0.005	2	x	x	3.7465
37fr	0.05	0.1	0.01	0.2	x	x	3.7465
38fr	0.05	0.1	0.05	0.02	x	x	3.617

39fr	0.05	0.1	0.1	3.5	x	x	3.3883
40fr	0.05	0.1	0.15	2	x	x	3.3145
41fr	0.05	0.4	0.005	0.2	x	x	3.7676
42fr	0.05	0.4	0.01	0.02	x	x	3.7676
43fr	0.05	0.4	0.05	3.5	x	x	3.6738
44fr	0.05	0.4	0.1	2	x	x	3.4117
45fr	0.05	0.4	0.15	0.2	x	x	3.3327
46fr	0.05	0.4	0.2	3.5	x	x	3.0785
47fr	0.05	0.7	0.005	2	x	x	3.5656
48fr	0.05	0.7	0.01	0.2	x	x	3.5656
49fr	0.05	0.7	0.05	0.02	x	x	3.3751
50fr	0.05	0.7	0.1	3.5	x	x	3.1271
51fr	0.05	0.7	0.15	2	x	x	3.0191
52fr	0.05	0.7	0.2	0.2	x	x	2.8325
53fr	0.1	0.1	0.005	0.02	x	x	3.3506
54fr	0.1	0.1	0.01	3.5	x	x	3.3506
55fr	0.1	0.1	0.05	2	x	x	3.1375
56fr	0.1	0.1	0.1	0.02	x	x	3.0271
57fr	0.1	0.1	0.15	3.5	x	x	2.8151
58fr	0.1	0.1	0.2	2	x	x	2.5713
59fr	0.1	0.1	0.2	0.2	x	x	2.5713
60fr	0.1	0.4	0.005	0.02	x	x	3.7465
61fr	0.1	0.4	0.01	3.5	x	x	3.7465
62fr	0.1	0.4	0.05	2	x	x	3.617
63fr	0.1	0.4	0.1	0.2	x	x	3.3883
64fr	0.1	0.4	0.15	0.02	x	x	3.2529
65fr	0.1	0.7	0.01	3.5	x	x	3.3985
66fr	0.1	0.7	0.05	2	x	x	3.2099
67fr	0.1	0.7	0.1	0.2	x	x	3.0355
68fr	0.1	0.7	0.15	0.02	x	x	2.9609
69fr	0.1	0.7	0.2	3.5	x	x	2.6845
70fr	0.15	0.1	0.005	2	x	x	3.1762
71fr	0.15	0.1	0.01	0.02	x	x	3.1762
72fr	0.15	0.1	0.05	3.5	x	x	3.0786
73fr	0.15	0.1	0.1	2	x	x	2.8735
74fr	0.15	0.1	0.15	0.2	x	x	2.77
75fr	0.15	0.1	0.2	3.5	x	x	2.6034
76fr	0.15	0.4	0.005	2	x	x	3.1141
77fr	0.15	0.4	0.01	0.2	x	x	3.1141
78fr	0.15	0.4	0.05	0.02	x	x	2.9735
79fr	0.15	0.4	0.1	3.5	x	x	2.8342
80fr	0.15	0.4	0.15	0.2	x	x	2.7543
81fr	0.15	0.4	0.2	0.02	x	x	2.5969
82fr	0.15	0.7	0.005	3.5	x	x	3.3985
83fr	0.15	0.7	0.01	2	x	x	3.3985
84fr	0.15	0.7	0.05	0.2	x	x	3.2099
85fr	0.15	0.7	0.2	0.02	x	x	2.7159
86fr	0.2	0.1	0.005	3.5	x	x	2.7866

87fr	0.2	0.1	0.01	2	x	x	2.7866
88fr	0.2	0.1	0.05	0.2	x	x	2.7298
89fr	0.2	0.1	0.1	0.02	x	x	2.6844
90fr	0.2	0.1	0.15	3.5	x	x	2.5644
91fr	0.2	0.1	0.2	2	x	x	2.4706
92fr	0.2	0.4	0.005	0.2	x	x	3.0278
93fr	0.2	0.4	0.05	0.02	x	x	2.9194
94fr	0.2	0.4	0.1	3.5	x	x	2.7584
95fr	0.2	0.4	0.15	2	x	x	2.6251
96fr	0.2	0.4	0.2	0.2	x	x	2.5071
97fr	0.2	0.4	0.2	0.02	x	x	2.5534
98fr	0.2	0.7	0.005	0.02	x	x	3.1231
99fr	0.2	0.7	0.05	2	x	x	3.011
100fr	0.2	0.7	0.1	0.2	x	x	2.8701
101fr	0.2	0.7	0.15	0.02	x	x	2.8479
102fr	0.2	0.7	0.2	3.5	x	x	2.5819

Table A.2 Secure Blank and Burst Mode Simulation – Male

Index	ulp	clp	FER	fdT	mean	stdev	PESQ
1fr	0	0	0	0	x	x	4.5
2fr	0.005	0.4	0.005	3.5	x	x	3.1375
3fr	0.005	0.7	0.005	0.02	x	x	3.1375
4fr	0.01	0.7	0.005	3.5	x	x	3.1375
5fr	0.005	0.4	0.01	2	x	x	2.8185
6fr	0.005	0.7	0.01	3.5	x	x	2.8185
7fr	0.01	0.7	0.01	2	x	x	2.8185
8fr	0.005	0.1	0.005	3.5	x	x	3.9674
9fr	0.005	0.1	0.01	2	x	x	3.435
10fr	0.01	0.1	0.01	2	x	x	3.435
11fr	0.01	0.4	0.005	2	x	x	3.435
12fr	0.01	0.4	0.01	0.2	x	x	3.435
13fr	0.005	0.4	0.05	0.02	x	x	3.2384
14fr	0.005	0.7	0.05	2	x	x	3.2384
15fr	0.01	0.7	0.05	0.2	x	x	3.2384
16fr	0.005	0.1	0.05	0.2	x	x	1.8028
17fr	0.01	0.4	0.05	0.02	x	x	1.8028
18fr	0.05	0.1	0.005	2	x	x	1.8028
19fr	0.05	0.1	0.01	0.2	x	x	1.8028
20fr	0.05	0.4	0.005	0.2	x	x	3.617
21fr	0.05	0.4	0.01	0.02	x	x	3.617
22fr	0.1	0.4	0.005	0.02	x	x	2.1989
23fr	0.1	0.4	0.01	3.5	x	x	2.1989
24fr	0.05	0.7	0.005	2	x	x	2.7199
25fr	0.05	0.7	0.01	0.2	x	x	2.7199
26fr	0.05	0.1	0.05	0.02	x	x	2.945

27fr	0.05	0.4	0.05	3.5	x	x	1.7359
28fr	0.1	0.4	0.05	2	x	x	1.7359
29fr	0.05	0.7	0.05	0.02	x	x	2.612
30fr	0.005	0.4	0.1	3.5	x	x	2.945
31fr	0.005	0.7	0.1	0.2	x	x	2.945
32fr	0.01	0.7	0.1	0.02	x	x	2.945
33fr	0.005	0.1	0.1	0.02	x	x	2.4999
34fr	0.01	0.1	0.1	0.2	x	x	2.4999
35fr	0.01	0.4	0.1	3.5	x	x	2.4999
36fr	0.1	0.1	0.005	0.02	x	x	2.4999
37fr	0.1	0.1	0.01	3.5	x	x	2.4999
38fr	0.05	0.1	0.1	3.5	x	x	2.4119
39fr	0.1	0.1	0.05	2	x	x	2.4119
40fr	0.05	0.4	0.1	2	x	x	1.6411
41fr	0.1	0.4	0.1	0.2	x	x	1.6411
42fr	0.1	0.7	0.01	3.5	x	x	1.6411
43fr	0.15	0.7	0.005	3.5	x	x	1.6411
44fr	0.15	0.7	0.01	2	x	x	1.6411
45fr	0.05	0.7	0.1	3.5	x	x	2.5781
46fr	0.005	0.4	0.15	2	x	x	1.8374
47fr	0.005	0.7	0.15	0.02	x	x	1.8374
48fr	0.15	0.4	0.005	2	x	x	1.8374
49fr	0.15	0.4	0.01	0.2	x	x	1.8374
50fr	0.005	0.1	0.15	3.5	x	x	2.1674
51fr	0.01	0.1	0.15	0.02	x	x	2.1674
52fr	0.01	0.4	0.15	2	x	x	2.1674
53fr	0.01	0.4	0.15	0.2	x	x	2.1674
54fr	0.15	0.1	0.005	2	x	x	2.1674
55fr	0.15	0.1	0.01	0.02	x	x	2.1674
56fr	0.1	0.7	0.05	2	x	x	1.7476
57fr	0.15	0.7	0.05	0.2	x	x	1.7476
58fr	0.1	0.1	0.1	0.02	x	x	2.1016
59fr	0.15	0.4	0.05	0.02	x	x	2.1016
60fr	0.05	0.1	0.15	2	x	x	1.5205
61fr	0.1	0.4	0.15	0.02	x	x	1.5205
62fr	0.15	0.1	0.05	3.5	x	x	1.5205
63fr	0.2	0.7	0.01	3.5	x	x	1.5205
64fr	0.05	0.4	0.15	0.2	x	x	2.1924
65fr	0.05	0.7	0.15	2	x	x	2.0554
66fr	0.2	0.4	0.005	0.2	x	x	3.9674
67fr	0.01	0.4	0.2	0.02	x	x	3.1762
68fr	0.2	0.7	0.05	2	x	x	3.1762
69fr	0.005	0.4	0.2	0.2	x	x	1.6411
70fr	0.005	0.7	0.2	3.5	x	x	1.6411
71fr	0.01	0.7	0.2	3.5	x	x	1.6411
72fr	0.1	0.7	0.1	0.2	x	x	1.6411
73fr	0.005	0.1	0.2	2	x	x	1.876
74fr	0.01	0.1	0.2	3.5	x	x	1.876

|							
75fr	0.15	0.4	0.1	3.5	x	x	1.9955
76fr	0.2	0.4	0.05	0.02	x	x	1.9955
77fr	0.1	0.1	0.15	3.5	x	x	1.8155
78fr	0.15	0.1	0.1	2	x	x	1.8155
79fr	0.2	0.1	0.005	3.5	x	x	1.8155
80fr	0.2	0.1	0.01	2	x	x	1.8155
81fr	0.05	0.4	0.2	3.5	x	x	1.8758
82fr	0.1	0.7	0.15	0.02	x	x	1.8758
83fr	0.05	0.7	0.2	0.2	x	x	2.1706
84fr	0.2	0.1	0.05	0.2	x	x	2.1706
85fr	0.2	0.7	0.1	0.2	x	x	2.1706
86fr	0.15	0.4	0.15	0.2	x	x	1.7476
87fr	0.15	0.1	0.15	0.2	x	x	1.4072
88fr	0.2	0.4	0.1	3.5	x	x	1.4072
89fr	0.1	0.1	0.2	2	x	x	1.2179
90fr	0.1	0.1	0.2	0.2	x	x	1.2179
91fr	0.2	0.7	0.15	0.02	x	x	1.98
92fr	0.15	0.7	0.2	0.02	x	x	2.1244
93fr	0.2	0.1	0.1	0.02	x	x	2.1244
94fr	0.1	0.7	0.2	3.5	x	x	1.1753
95fr	0.15	0.4	0.2	0.02	x	x	1.1753
96fr	0.2	0.4	0.15	2	x	x	1.1753
97fr	0.15	0.1	0.2	3.5	x	x	1.9133
98fr	0.2	0.1	0.15	3.5	x	x	1.9709
99fr	0.2	0.7	0.2	3.5	x	x	2.5151
100fr	0.2	0.7	0.005	0.02	x	x	1.1565
101fr	0.2	0.4	0.2	0.2	x	x	1.4392
102fr	0.2	0.1	0.2	2	x	x	1.9709
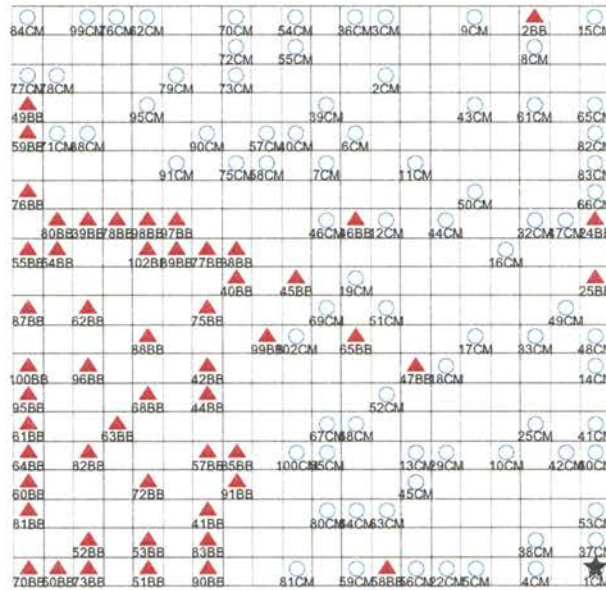
Figure A.1 SD$_{fw}$ QoS Map: Clear MELP vs. Secure Blank and Burst – Male
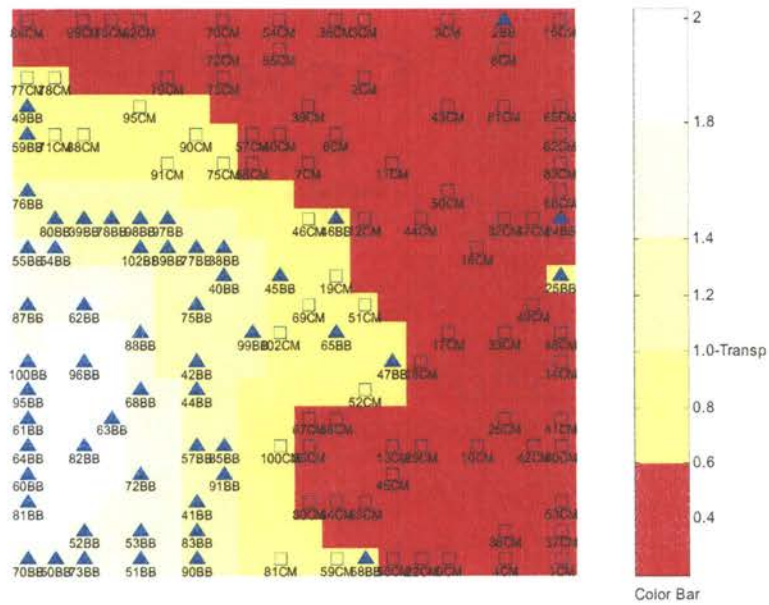
Figure A.2 SD$_{fw}$ Color Map: Clear MELP vs. Secure Blank and Burst – Male
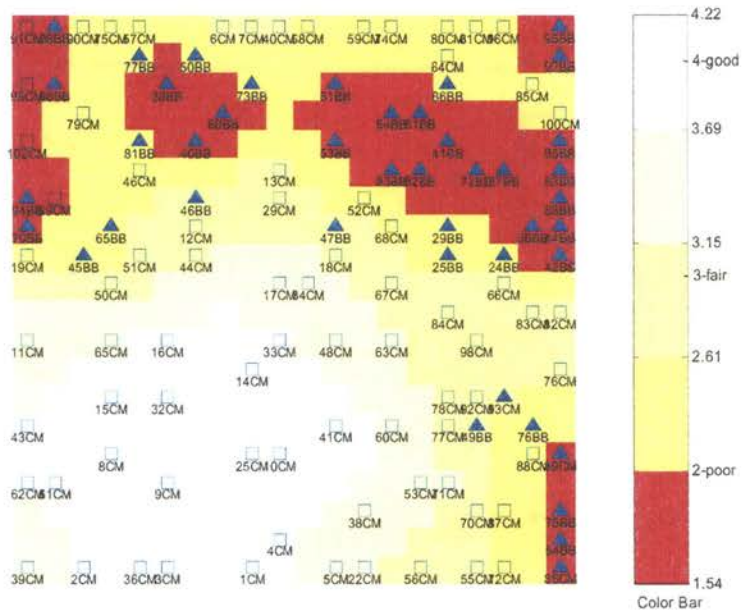
206
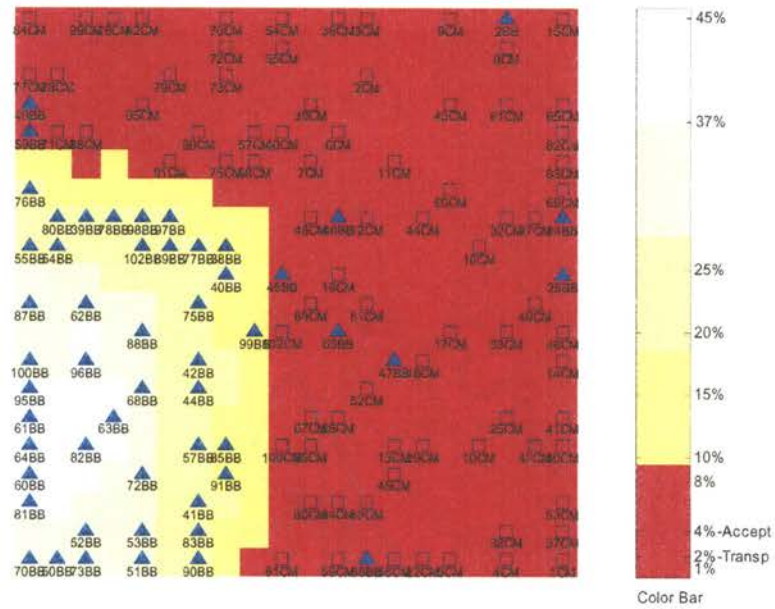
**Figure A.3 Percent Outlier Color Map: Clear MELP vs. Secure Blank and Burst –
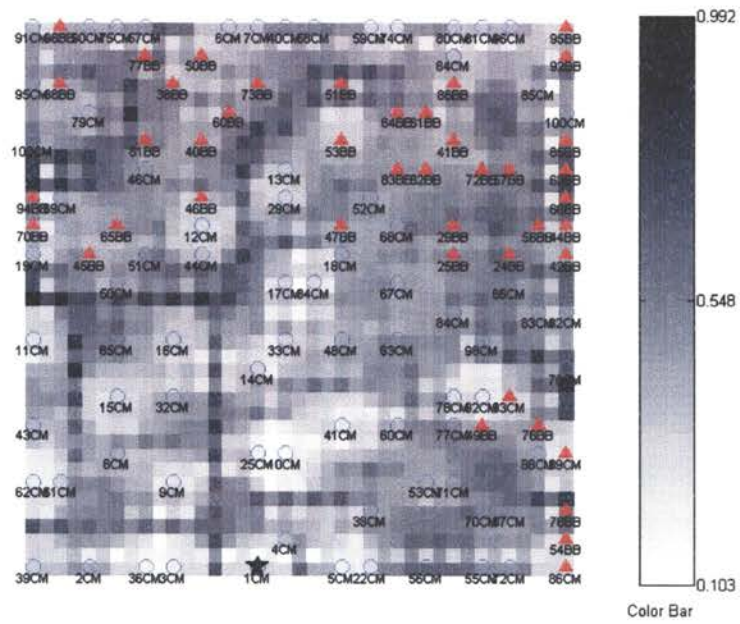Male**
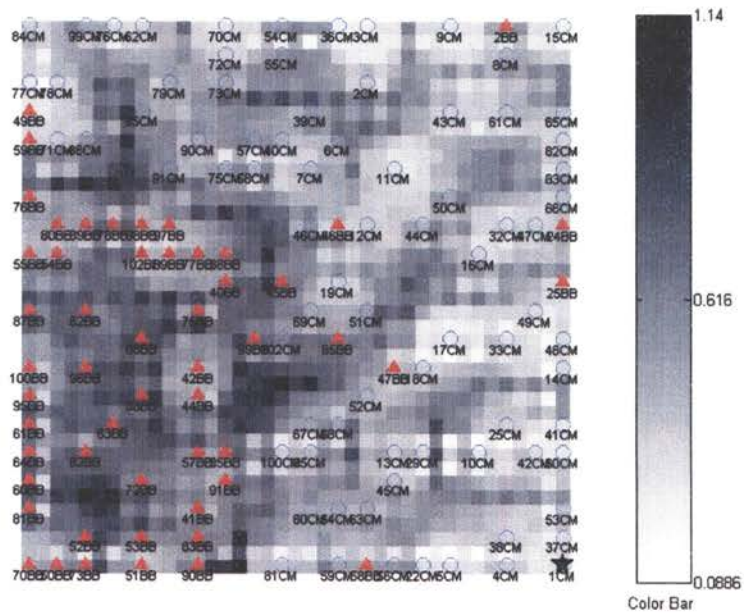
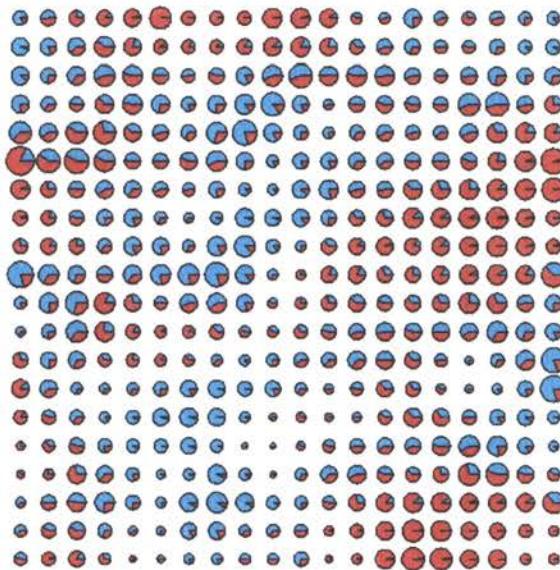**Figure A.4 SD$_{fw}$ U-Matrix QoS Map: Clear MELP vs. Secure Blank and Burst –
Male**
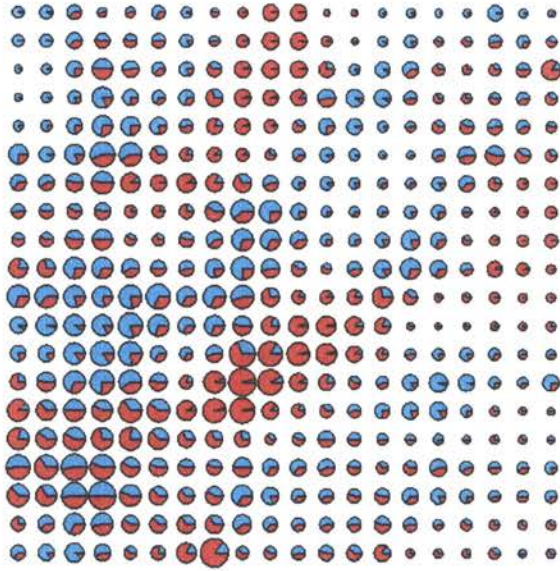
**Figure A.5 SD$_{fw}$ Relative Importance of Internet Model Parameters *ulp* and *clp* - Male**
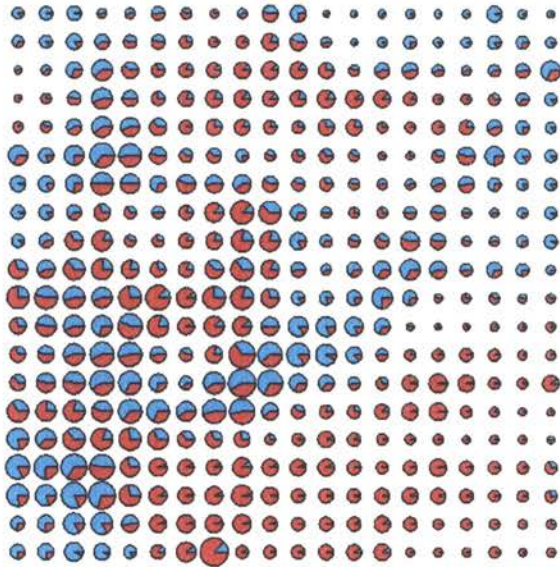
Figure A.6 SD$_{fw}$ Relative Importance of CDMA Model Parameters FER and *fdT* – Male
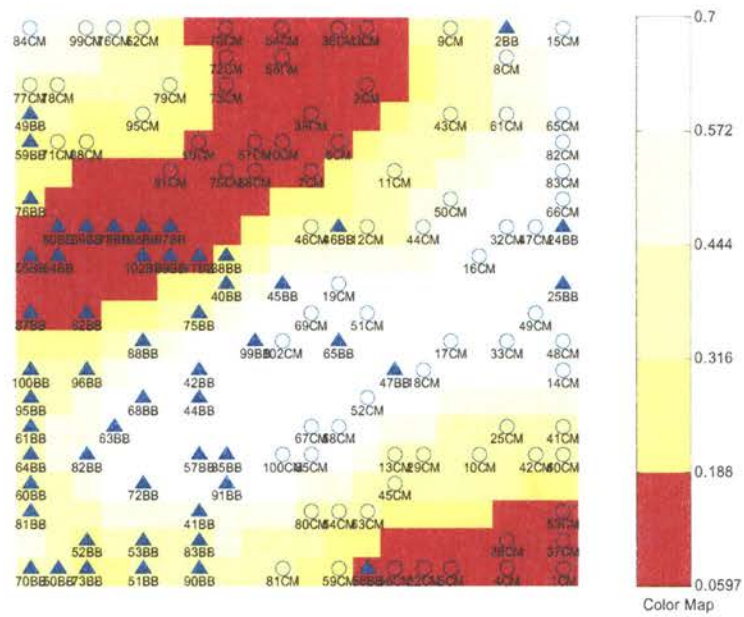
Figure A.7 SD_{fw} *Ulp* Color Map: Clear MELP vs. Secure Blank and Burst – Male
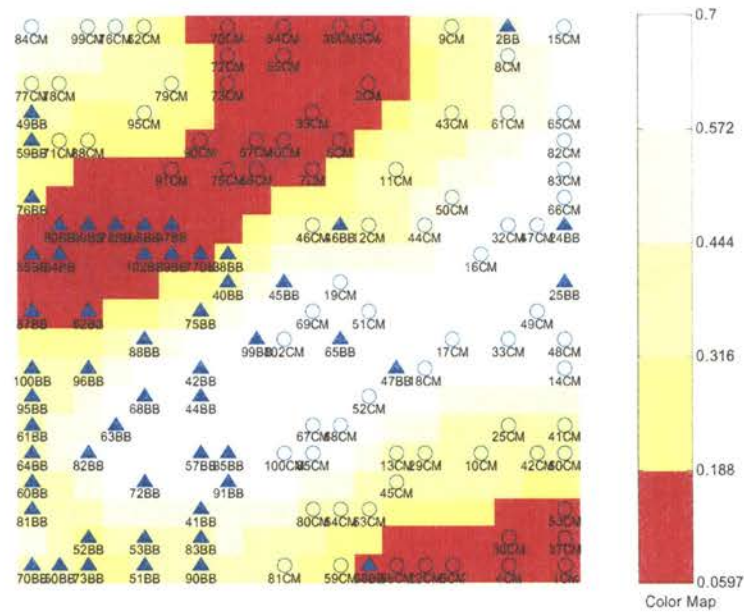
Figure A.8 SD_{fw} *Clp* Color Map: Clear MELP vs. Secure Blank and Burst – Male
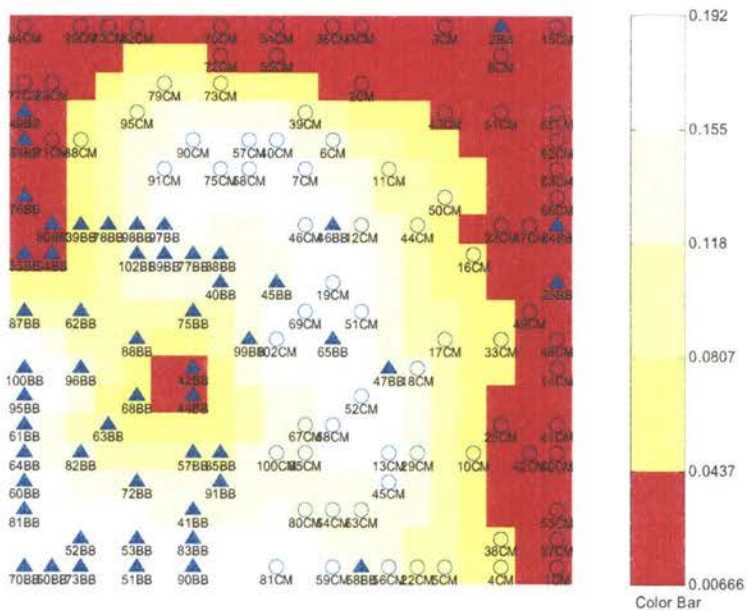
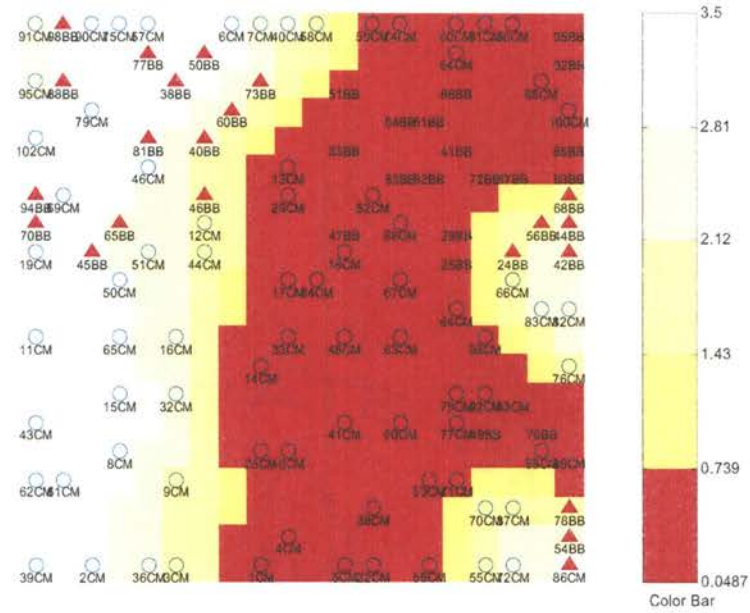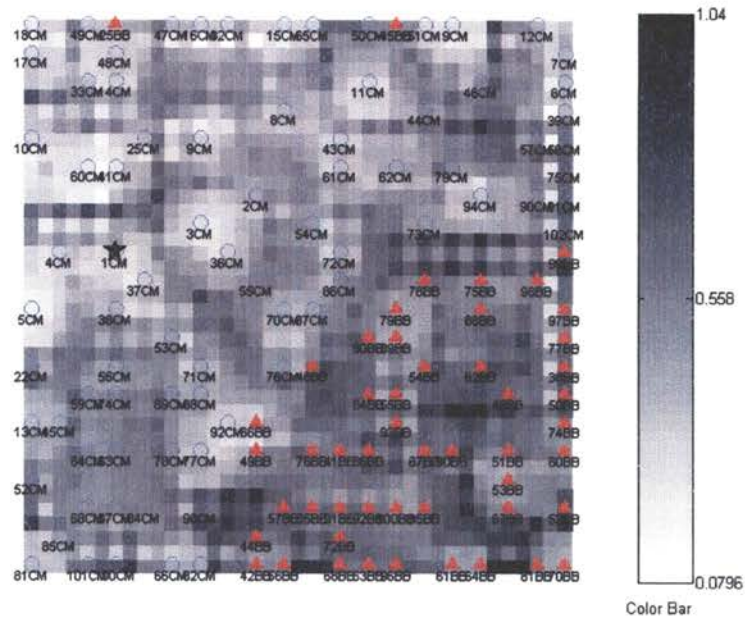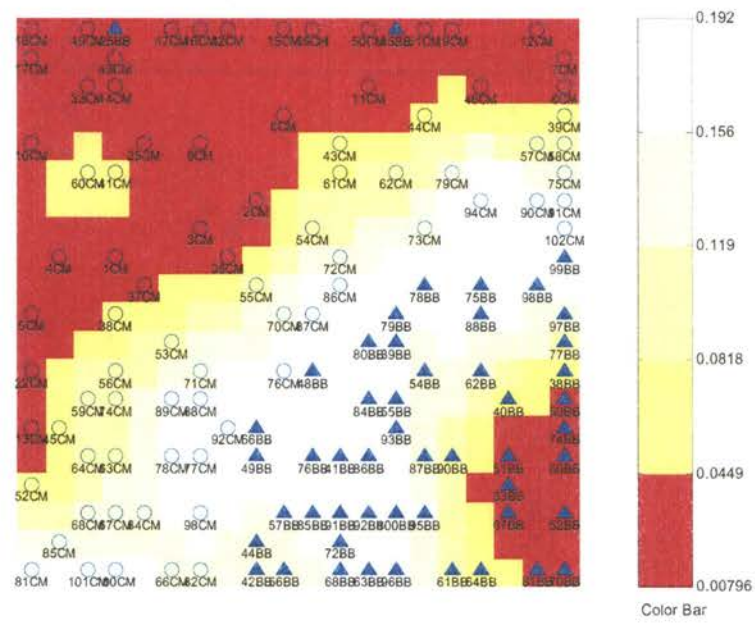**Figure A.9 SD$_{fw}$ FER Color Map: Clear MELP vs. Secure Blank and Burst – Male**
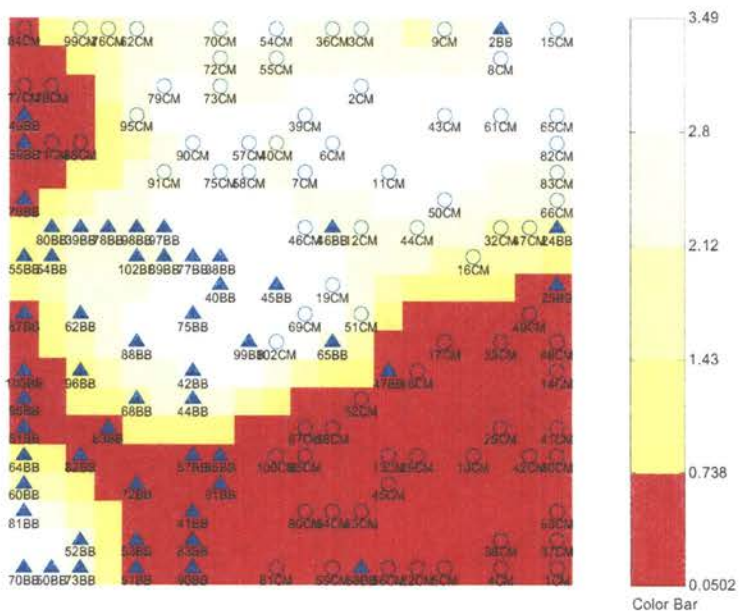
Figure A.10 SD$_{fw}$ *fdT* Color Map: Clear MELP vs. Secure Blank and Burst – Male

PESQ QoS Maps: Clear MELP vs. Secure Blank and Burst – Female
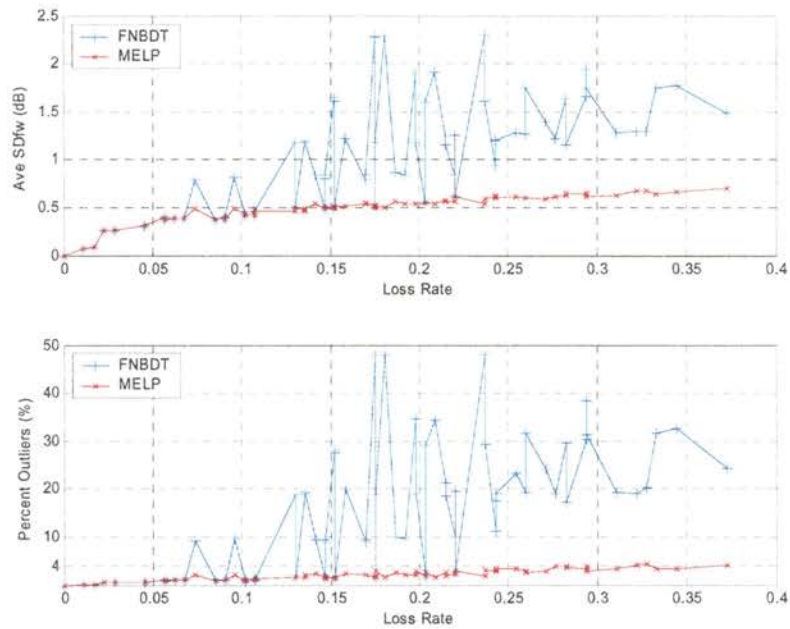
Figure A.11 Ave SD$_{fw}$ and Percentage Outlier Frames vs. Loss Rate for Clear MELP and Secure Blank and Burst FNBDT Modes – Female
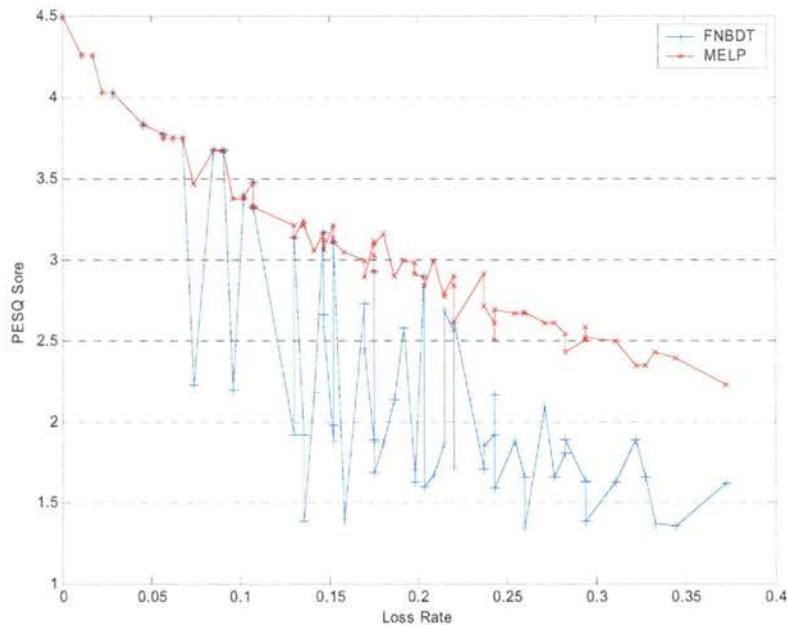
Figure A.12 PESQ Score vs. Loss Rate for Clear MELP and Secure Blank and Burst FNBDT Modes - Female
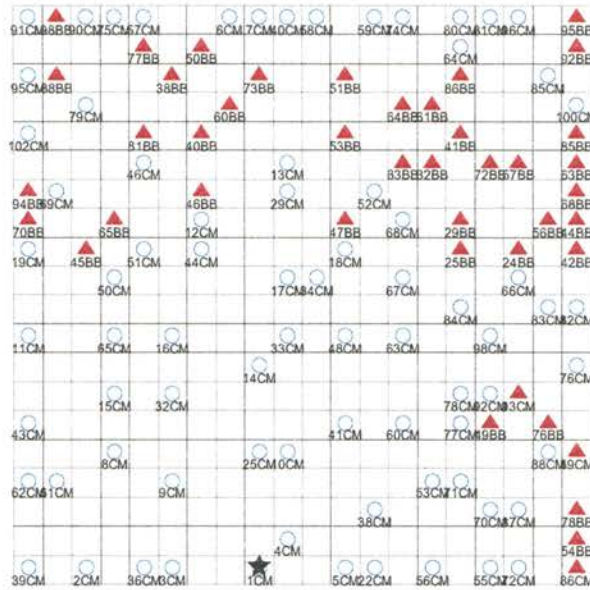
Figure A.13 QoS Map: Clear MELP vs. Secure Blank and Burst – Female

**Figure A.14 PESQ Color QoS Map for Clear MELP vs. Secure Blank and Burst –
Female**

212

Figure A.15 U-Matrix QoS Map: Clear MELP vs. Secure Blank and Burst – Female

Figure A.16 Relative Importance of Internet Model Parameters *ulp* and *clp* - Female

213
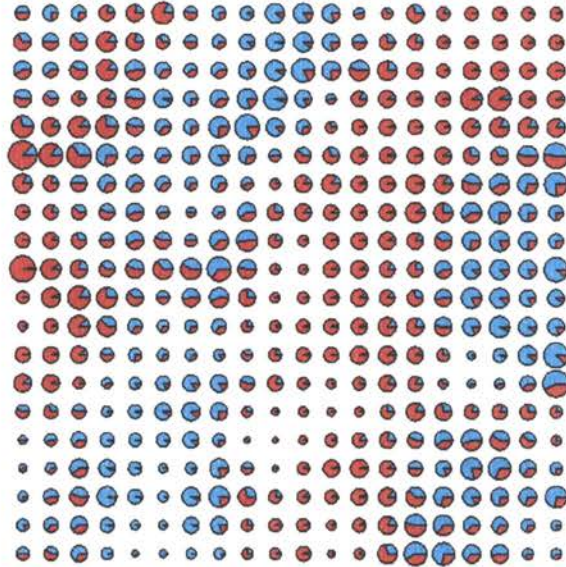
Figure A.17 Relative Importance of CDMA parameters FER and *fdT* - Female
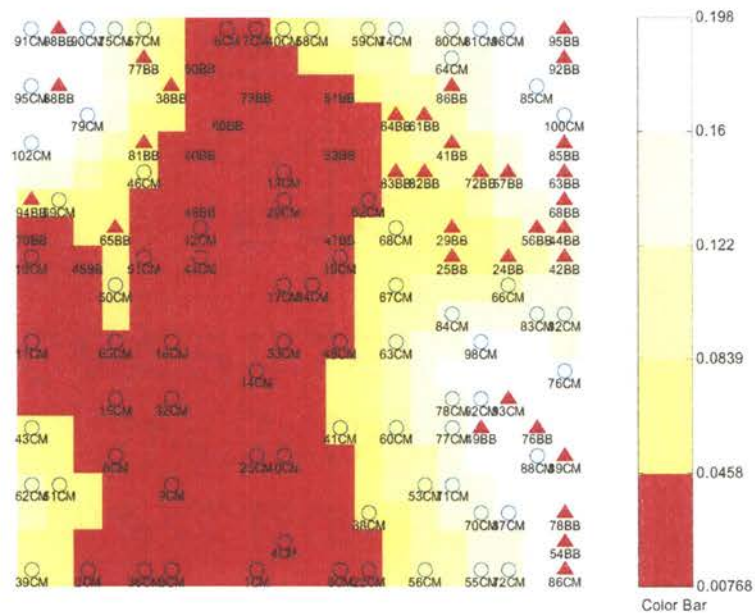
Figure A.18 *Ulp* Color Map: Clear MELP vs. Secure Blank and Burst – Female

214
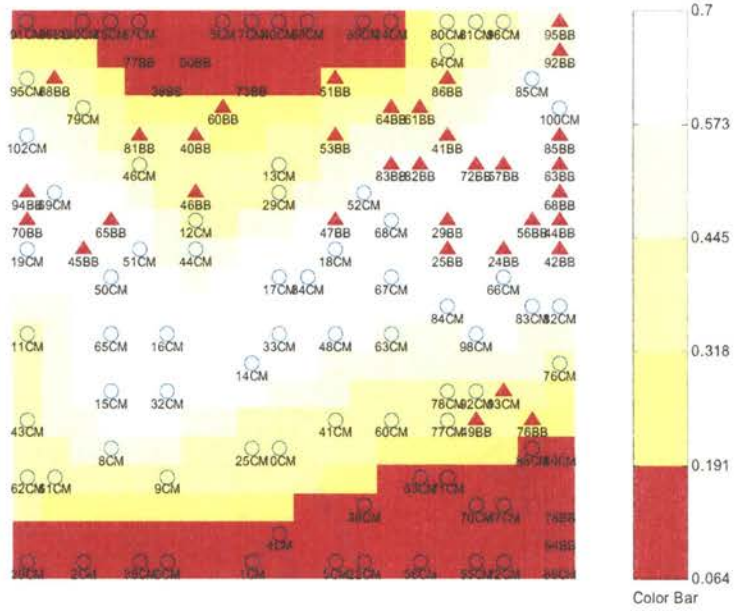
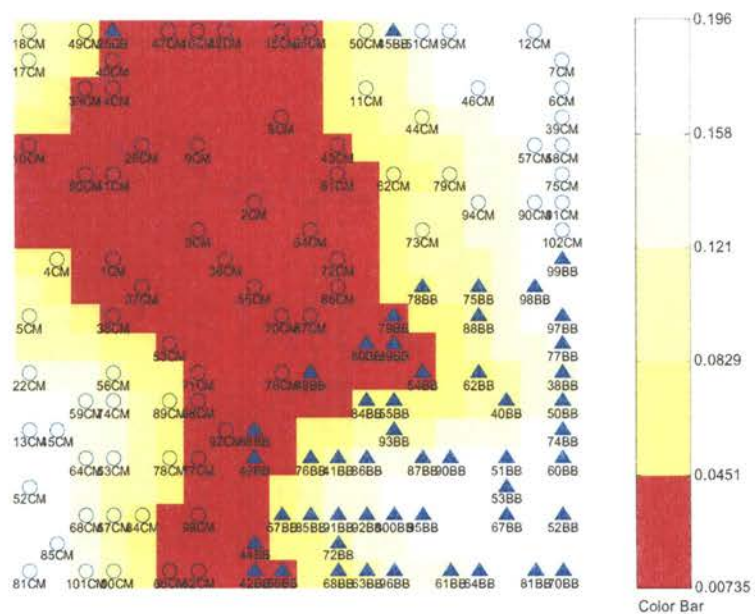**Figure A.19 *Clp* Color Map: Clear MELP vs. Secure Blank and Burst – Female**
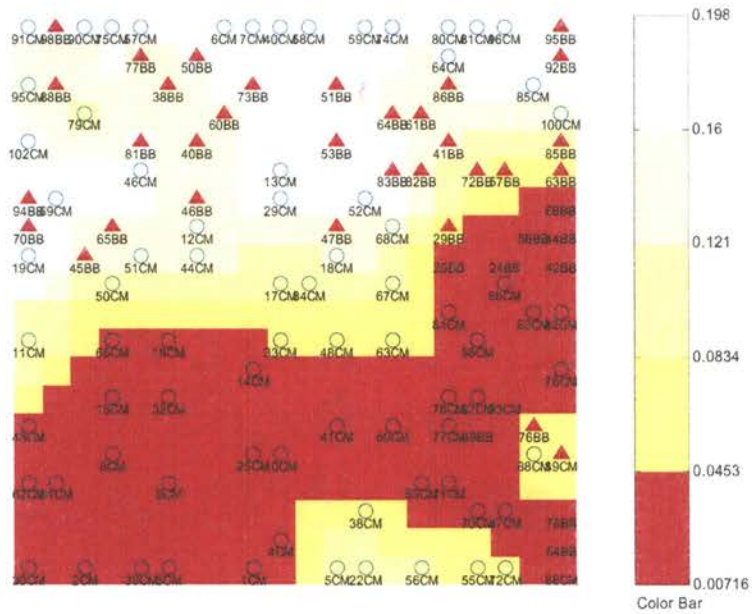
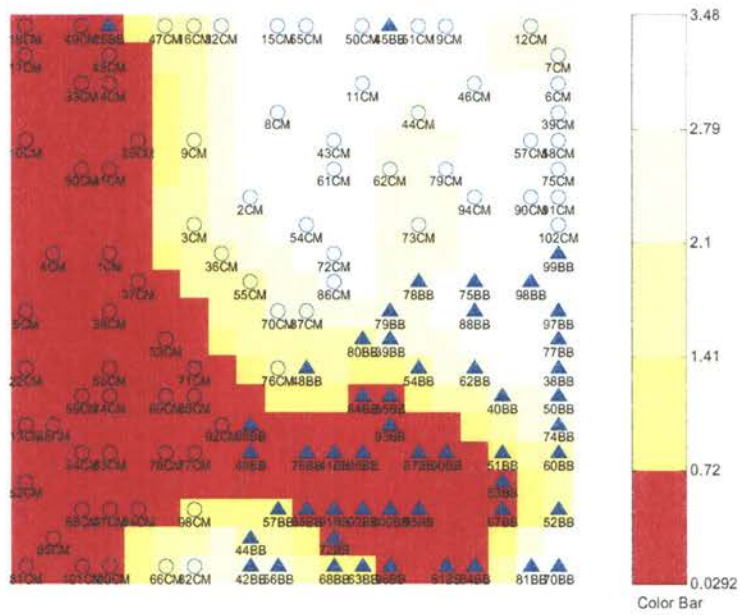**Figure A.20 FER Color Map: Clear MELP vs. Secure Blank and Burst – Female**

Figure A.21 *fdT* **Color Map: Clear MELP vs. Secure Blank and Burst – Female**

Ave SD$_{fw}$ QoS Maps: Clear MELP vs. Secure Blank and Burst – Female

Figure A.22 SD$_{fw}$ QoS Map: Clear MELP vs. Secure Blank and Burst – Female

Figure A.23 SD$_{fw}$ Color Map: Clear MELP vs. Secure Blank and Burst – Female

Figure A.24 Percent Outliers Color Map: Clear MELP vs. Secure Blank and Burst – Female

Figure A.25 SD$_{fw}$ U-Matrix QoS Map: Clear MELP vs. Secure Blank and Burst – Female

Figure A.26 SD$_{fw}$ *Ulp* Color Map: Clear MELP vs. Secure Blank and Burst – Female
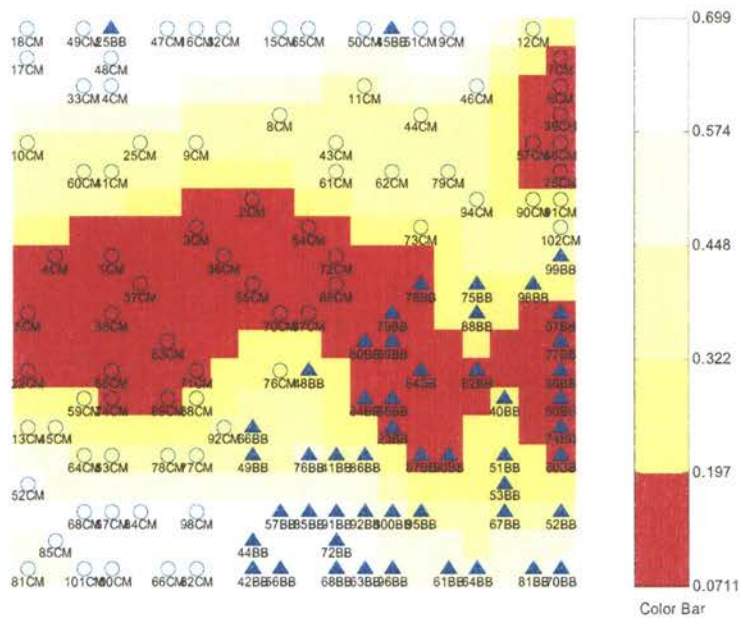
Figure A.27 SD$_{fw}$ *Clp* Color Map: Clear MELP vs. Secure Blank and Burst – Male

Figure A.28 SD$_{fw}$ FER Color Map: Clear MELP vs. Secure Blank and Burst – Female

Figure A.29 SD$_{fw}$ *fdT* Color Map: Clear MELP vs. Secure Blank and Burst – Female

Ave SD$_{fw}$ QoS Maps: FNBDT with FEC vs. FNBDT without FEC – Male
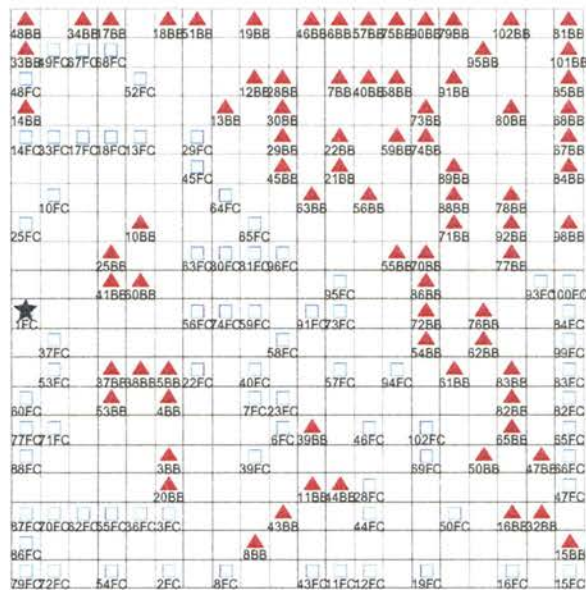
Figure A.30 SD$_{fw}$ QoS Map: FNBDT with and without FEC – Male

220
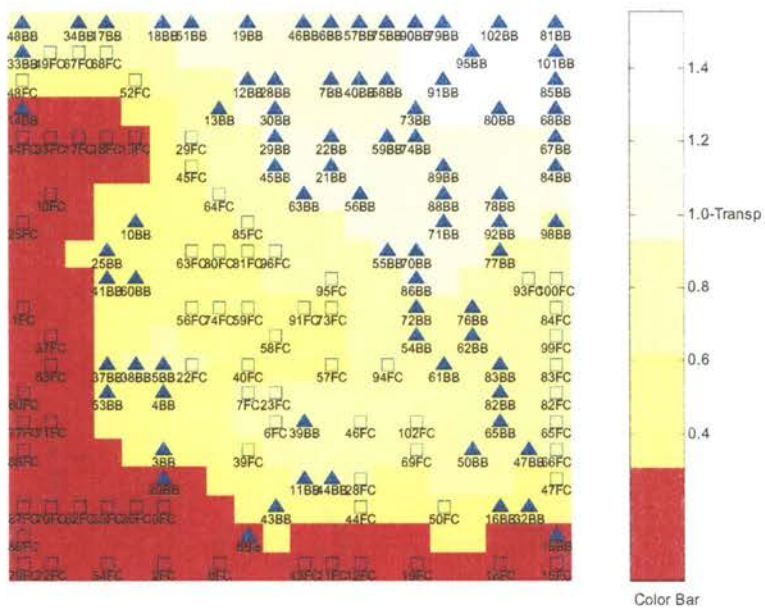
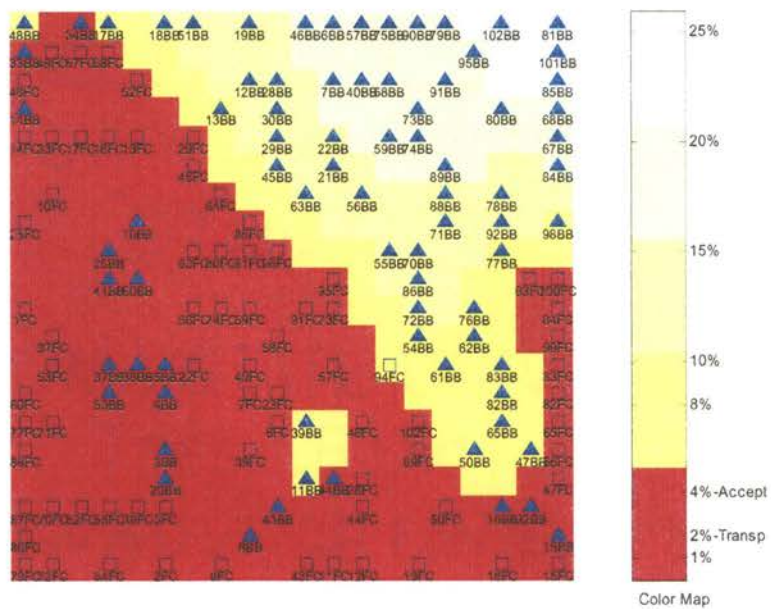**Figure A.31 SD$_{fw}$ Color Map: FNBDT with and without FEC – Male**

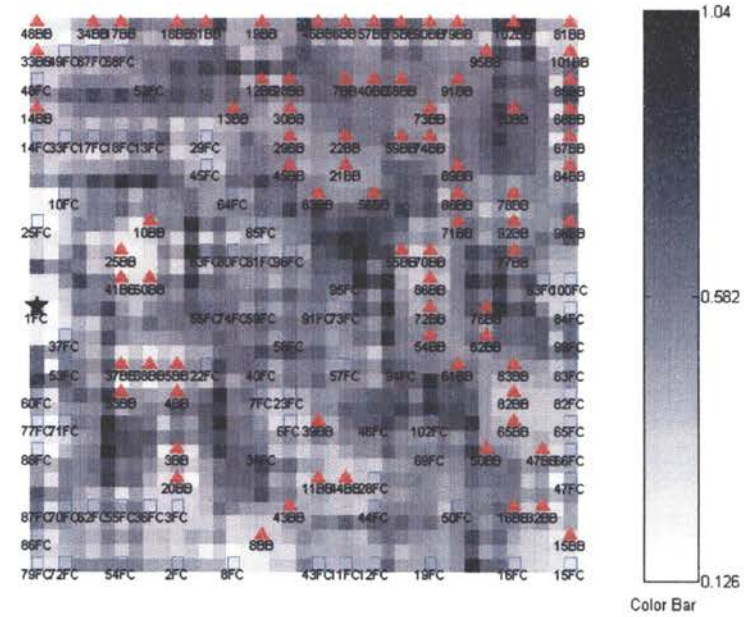**Figure A.32 Percent Outliers Color Map: FNBDT with and without FEC – Male**

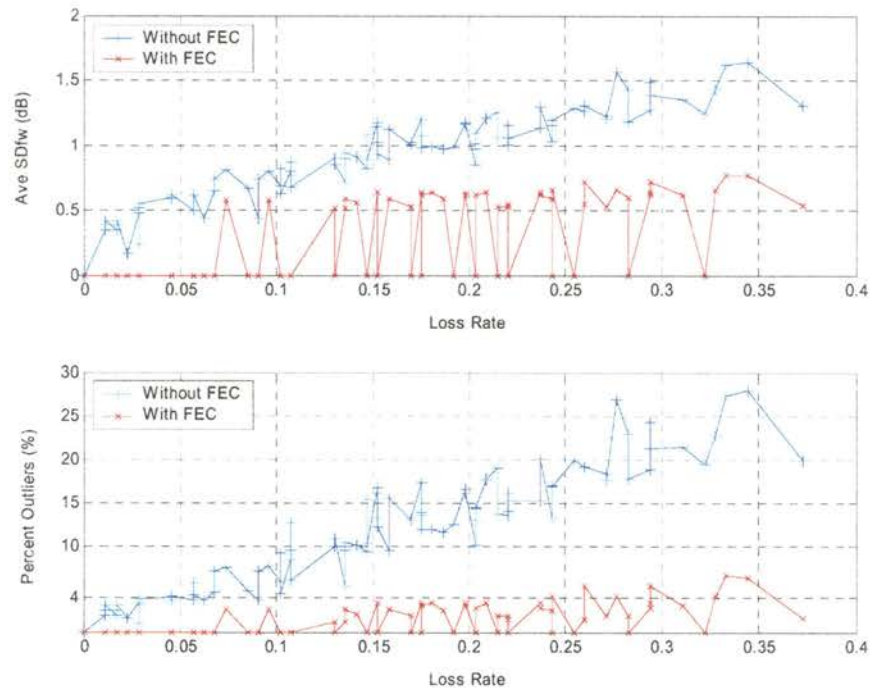**Figure A.33 U-Matrix: FNBDT with and without FEC – Male**

Figure A.34 Ave SD_fw and Percentage Outlier Frames vs.. Loss Rate for FNBDT with FEC and FNBDT without FEC - Male