Quality of Service Analysis for Audio over Cellular Voice Networks and Cellular Wireless Wide Area Networks

by

Omair S. Malik

Submitted to the

Department of Electrical Engineering and Computer Science

in partial fulfillment of the requirements for the degree of

Master of Engineering in Electrical Engineering and Computer Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

September 2007

Author ....
Department of Electrical Engineering and Computer Science
August 21, 2007

Certified by ....
John V. Guttag
Professor, Computer Science and Engineering
Thesis Supervisor

Accepted by ....
Arthur C. Smith
Chairman, Department Committee on Graduate Students

Quality of Service Analysis for Audio over Cellular Voice Networks and Cellular Wireless Wide Area Networks

by

Omair S. Malik

Submitted to the Department of Electrical Engineering and Computer Science
on August 21, 2007, in partial fulfillment of the
requirements for the degree of
Master of Engineering in Electrical Engineering and Computer Science

Abstract

Cellular Wireless Wide Area Networks (WWANs) are most prevalent and offer high-bandwidth data transfer. We believe WWANs can be availed for voice communications employing Voice Over IP technologies. Such a communication will be of better quality and offer higher resilience compared to voice communication over a cellular phone (using cellular voice networks).

We present a Quality of Service analysis of one-way voice communication over cellular voice networks and cellular WWANs. By studying different quality metrics we test if WWANs offer a better solution for voice communications compared to traditional cellular voice networks. We also study if employing more than one WWANs or cellular voice networks leads to a higher resilience in voice communication.

Thesis Supervisor: John V. Guttag
Title: Professor, Computer Science and Engineering

Acknowledgments

I would like to thank John Guttag and Asfandyar Qureshi for their valuable support and guidance. This thesis would not have been possible without them.

This thesis is dedicated to my family. Their love, passion, sacrifice and support kept me going in the darkest of times.

I would also like to thank Zaid Samar, Zeeshan H. Syed, Ebad Ahmed, Asif Khan and Dilan Jayawardane for their friendship, support and encouragement.

Finally to the human mind: a complex, powerful organ, yet so fragile.

Contents

List of Figures

Chapter 1

Introduction

This thesis describes an experimental setup to analyze voice audio quality over Wireless Wide Area Networks (WWANs) in comparison to traditional cellular voice networks. Current cellular voice networks suffer from low quality voice communication and unexpected call terminations (also referred to as "*dropped calls*"). WWANs are most prevalent and offer high-bandwidth data transfer. By porting existing Voice-over-IP technologies to WWANs we can create a voice communications network that offers higher voice quality and reliability.

Our research leverages the widespread cellular wireless data networks to develop and test a resilient high quality voice communications network. This thesis develops a testbed for comparing this voice network to the traditional cellular phone network, discussing its design and evaluating the performance of both the networks under different conditions.

1.1 Motivation

Our work was motivated by a need for a mission critical voice communications system. As part of a telemedicine project, we wanted to transmit bi-directional audio from a moving ambulance using public carrier networks.

We realized the traditional cellular voice networks were not suitable for our application. The loss in voice quality over such networks (especially when transmitting from a moving van) makes it harder for the doctor at the other end to discern between the various auditory symptoms. Also, the phenomenon of "dropped calls" meant an unexpected abruption in the voice communication that is undesirable.

In most urban areas, there are a large number of public carrier wireless channels providing mobile connectivity to the Internet, using standard cellular technologies such as `GSM` and `CDMA2000`. Notably, these providers have overlapping coverage areas, allowing us to connect to more than one provider at the same time. Other researchers have investigated the use of higher bandwidth wireless technologies, such as 802.11 [4, 28], for public networks with varying degrees of success.

Individual cellular Wireless Wide Area Network(WWAN) channels provide enough bandwidth for our audio application. The upstream bandwidth[1] offered by these channels is limited in comparison to a wireless local area network (e.g. `802.11`). However, the best WWAN channels available to us in the greater-Boston area provide a couple of hundred kilobits per second of *peak* bandwidth which is more than sufficient for audio transmission[2].

WWANs also provide us with the ability to use intelligent scheduling schemes to send voice data over these channels. E.g., in case one network is facing high latency, we can adapt our application on the fly to send data on another overlapping network.

1.2 Challenges

WWAN channels provide little in the way of quality of service guarantees for data channels. We have observed that WWANs with overlapping coverage have widely varying channel characteristics. Motion and location cause many additional complications, the details depending on the network technology and service provider. WWAN channels are handicapped by high and variable round trip times, occasional outages, considerable burstiness, and much instability when moving [53, 54, 25]. For

[1] Here the term upstream bandwidth here refers to how much data we can *send* and the term downstream bandwidth refers to how much data we can *receive*

[2] We use an encoding bitrate for the WWAN transmission comparable to the throughput of a traditional cellular voice network

instance, average packet round-trip-times on different channels can easily differ by $500ms$. Even on a single channel, at different times and locations, individual packet times can vary by this much.

In most areas the bandwidth/latency/loss characteristics of the available public carrier wireless channels vary by more than an order of magnitude (such as among a set of channels containing both 2.5G and 3G cellular data channels). We also expect there to be a high degree of dynamic variation in each wireless channel's *quality*—partly due to the motion of the vehicle. There is evidence of high variability in bandwidth and packet-latency on real WWAN channels [53]. Spatial variation in the quality of service provided by different carriers is well-known and largely depends on the carrier's placement of cell-towers relative to the terminal. One of the reasons for temporal variation is that demand varies over time, while the total bandwidth in any given cell remains roughly constant.

These issues led us to consider using redundancy. By taking advantage of service provider diversity, overlapping coverage, and network technology diversity (e.g. GPRS, CDMA, UMTS), we attempt to provide the illusion that a reliable stable high-bandwidth channel is available.

In our environment, however, the underlying links are neither stable nor homogeneous. Therefore, the manner in which we decide to schedule the transmission of voice data packets can have a large influence on observed latencies, bandwidth, and loss rate. Furthermore, we care more about average latency and its variance than packet loss for voice transmission.

Thus, these redundancy schemes involve sending voice data in some order over multiple physical channels, possibly reassembling the data in the correct order at the other end before passing it on. Unfortunately, the scope of this thesis does not involve creating/testing complex redundancy schemes. We use the simple scheme of transmitting data on more than one physical channel.

Our decision to use WWANs also demands a robust buffering algorithm to decrease the variance in average latency. This introduces another type of latency at both ends. However, unlike the inherent latency linked to the physical network, we can decrease the quantity of this secondary latency by fine tuning our buffering algorithm.

For traditional cellular voice networks, we had to ensure that any degradation in voice quality was due to the network, and not because of our hardware. Extra care was taken to transfer uncompressed voice data to the input, and to record voice data without any sort of compression at the output.

1.3 Goals

Our primary goal is to prove that Wireless Wide Area Networks are capable of voice communication that has the following properties:

High Quality

We gauge the quality of the received audio in terms of metrics including frequency and phase. High quality audio means less frequency filtering and very low (preferably none) phase shift. This requires the use of an efficient voice codec.

Reliability

Voice transmission is more reliable compared to the traditional cellular voice networks. This means there are fewer call drops and even in the worst case scenario, the voice quality on WWANs is atleast as good as that on cellular voice networks. This requires the use of the overlapping WWANs as well as a robust buffering scheme.

Low Latency

The network is good enough for real-time voice communication. This means an average latency less than 250ms [53]. Even though we are testing uni-directional audio, a latency comparatively higher than 250ms is undesirable. We can not change the value of the network latency, but we can average out the latency value by the use of an efficient buffering scheme.

Voice communication must be as insensitive to motion as possible. Unfortunately, even in the best case scenario, some degradation in link quality due to motion may occur. Doppler effects, resulting from the motion, can be correlated across all channels. Similarly if provider base-stations are co-located, loss and bandwidth on the different channels may exhibit correlation.

Finally, our research intends to highlight the problems with the current cellular voice networks.This thesis does not focus on improving such networks; we tackle the problem of developing a better voice communications system, using widely available technology.

1.4 Thesis Contributions

This thesis makes the following major contributions:

Cellular Voice Networks' Characteristics

We explore the characteristics of cellular voice networks through real-world experiments. We show that these networks degrade the quality of the audio and suffer from occasional unexpected termination of calls.

WWAN Channel Characteristics

Through experiments and discussions of WWAN technology standards, we explore the characteristics of real WWAN channels. We show that `CDMA2000 1xEV-DO` and `GSM EDGE/EGPRS` channels are handicapped by high and variable round trip times, occasional outages, considerable burstiness, and much instability when moving.

Transmitting Audio over Heterogeneous and Unstable Channels

We assume that the characteristics of WWAN channels can change rapidly (e.g. due to motion relative to a base-station), and show that we can still transfer high quality audio with high reliability. Even though we use a simple redundancy scheme, the final results prove that voice transmission over WWAN channels is possible.

1.5 Thesis Organization

This thesis begins by providing some background on WWAN channels and cellular voice networks in chapter 2. We also talk about our choice of voice quality metrics. We provide an overview of the testbed architecture in chapter 3, and follow in chapter 4 with a discussion of the experiments we performed with their results. Chapter 5 evaluates the results of the experiments and debates the feasibility of a WWAN based voice communications system in comparison to the traditional cellular voice networks. Finally, chapter 6 provides conclusions and directions for future work.

Chapter 2

Background

This chapter covers some background material.

2.1 Wireless WAN's

Figure 2-1 enumerates some of the various wireless data networking technologies that are available.

The cellular WWAN technologies are the most prevalent. In most urban areas, there are a large number of public carrier wireless channels providing mobile connectivity to the Internet[1], using standard cellular technologies such as `GSM EDGE/EGPRS` and `CDMA2000 1xEV-DO`[2]. Notably, there are multiple providers and these providers have overlapping coverage areas, allowing us to connect to more than one provider at the same time.

In connections over these WWAN's the wireless link dominates in determining the

[1]based networks
[2]Referred to as `EGPRS` and `EV-DO` respectively in the thesis.

Channel	Latency	Loss	Throughput/Down	Throughput/Up
GSM EDGE/EGPRS	High	Low	240	100
CDMA2000 1xEV-DO	High	Low	300	120
802.11b	Low	Medium	5000	5000

Figure 2-1: Characteristics of wireless data network channels. The throughputs are estimated averages in kilobits-per-second.

quality of the connection. Also, Doppler effects due to motion and the positioning of the terminal relative to the provider's base-stations can significantly reduce the available throughput.

The last IP hop's latency accounts for over 99% of the overall latency of the entire route taken by packets sent from a WWAN terminal [53]. Providers using EGPRS use IP tunnels to transfer a terminal's data packets through the provider network. The last IP hop in such a situation is the entire provider network, which might explain why it dominates.

WWAN channels are handicapped by high and variable round trip times, occasional outages, considerable burstiness, and much instability when moving. The quality of a WWAN link is highly dependent not only on the technology used, but perhaps even more so on the way the provider has decided to set up the network (e.g. the distribution of a provider's base-stations).

EV-DO channels and EGPRS channels have comparable speeds, as noted in figure 2-1. For small TCP-SYN packets and stationary terminals, the measured average packet round-trip-times on a EV-DO link is around $315ms$ and $550ms$ on an EGPRS link. For larger UDP packets and stationary terminals, both types of channels have an average packet latency of around $800ms$ with a standard deviation of around $100ms$ for 768-byte packets [53].

Motion causes the quality of these links to degrade. Average throughputs arent affected much by motion, but on both channels the standard deviation of the throughput gets multiplied by a factor of five. When moving, the average latency rises to $760ms$ and the standard deviation jumps to $460ms$ [53].

Disconnections on WWAN channels are rare and uncommon. In case of a disconnection we can reconnect immediately without changing location. However, occassional service disruptions while moving are unavoidable.

Finally, the WWAN extensively uses link-layer retransmissions and forward-error-correction (FEC) for data packets, increasing packet delays and delay variances, but keeping the bit-error-rate for the channel at a low level.

Other researchers have investigated the use of higher bandwidth wireless technologies, such as 802.11, for public networks [4, 28, 3] with varying goals and varying degrees of success. 802.11 was not designed to be used over large distances, in multi-hop wireless networks, and with vehicular mobile terminals. Therefore its use in this way poses a number of technological challenges. For instance, technologies like `EV-DO` have been designed to seamlessly switch base-stations whenever appropriate.

In this thesis the only wireless data network connections available to us are a `CDMA 1xEV-DO` link and a `GSM EDGE/EGPRS` link.

2.2 Cellular Voice Networks

We use the term *Cellular Voice Networks* to refer to the mobile telephony networks, specifically those that employ second-generation wireless telephone technology (also referred to as `2.5G` networks). For this thesis we restrict ourselves to `CDMA 1xRTT` and `GSM`[3], the two most widely used `2.5G` networks in the world. Figure 2-2 compares the two networks in terms of bandwidth available.

Both `CDMA` and `GSM` are highly optimized for low bandwidth voice communication. The GSM Enhanced Full Rate speech codec (`EFR`) uses 12.2 kbit/s for speech coding and 10.6 kbit/s for error correction [44]. Even though `CDMA 1xRTT` offers higher bandwidth, it transfers voice data at a speed comparable to `GSM` [29].

`CDMA` and `GSM` channels are prone to degradation in voice quality. Speech codecs like `EFR` are lossy codecs that encode speech at high levels of compression at the expense of quality degradation. The saved bandwidth is used for error correction, and its bitrate is comparable to the encoding bitrate. This way the voice stream is reconstructed perfectly[4], though with a loss of tonality, even in situations of bad network connectivity [44].

A bigger problem with cellular voice networks is *dropped calls*. Dropped call is the common term for a wireless mobile phone call that is terminated unexpectedly as a result of technical reasons. Areas where users experience a large number of dropped

[3]Referred to as `CDMA` and `GSM` respectively in the thesis

[4]By perfect we mean the user at the other end is able to hear the speech correctly

Channel	Latency	Loss	Throughput/Down	Throughput/Up
GSM	Low	Low	40	20
CDMA2000 1xRTT	Low	Low	120	120

Figure 2-2: Characteristics of wireless data network channels. The throughputs are estimated averages in kilobits-per-second.

calls are commonly referred to as dead zones.

One reason for a dropped call is when the mobile phone moves out of range of a wireless network. An active call cannot usually be maintained across a different company's network (as calls cannot be re-routed over the traditional phone network while in progress), resulting in the termination of the call once a signal cannot be maintained between the phone and the original network. Another common reason is when a phone is taken into an area where wireless communication is unavailable, interrupted, interfered with, or jammed. From the network's perspective, this is the same as the mobile moving out of the coverage area.

Occasionally calls are dropped upon handoff between cells towers within the same provider's network. This may be due to an imbalance of traffic between the two cell sites' areas of coverage. If the new cell site is at capacity, it cannot accept the additional traffic of the call trying to "hand in." It may also be due to the network configuration not being set up properly, such that one cell site is not aware of the cell the phone is trying to hand off to. If the phone cannot find an alternative cell to move to that can take over the call, the call is lost.

Co-channel and Adjacent channel interference can also be responsible for dropped calls in a wireless network. Neighbor cells with the same frequencies interfere with each other, deteriorating the quality of service and producing dropped calls. Transmission problems are also a common cause of dropped calls.

Areas where users experience a large number of dropped calls are commonly referred to as dead zones. Apart from cell tower issues, dead zones are also created due to geographical reasons. E.g., cellular signal strength drops if the area is surrounded by tall buildings or in a tunnel.

A great amount of money and time is invested by wireless operators in order to

Category	Metric	Description
Timeliness	Latency	Time taken for a message to be transmitted
	Response Time	Round trip time from request transmission to reply receipt
	Jitter	Variation in delay or response time
Bandwidth	Throughput	Bandwidth required or used, in bits or bytes per second.
Reliability	Loss or corruption rate	Proportion of total data which does not arrive as sent, e.g. network error rate
Perceived QoS (Audio Quality)	Encoding Rate	Audio sampling rate and number of bits
	High Fidelity	Frequency response and phase variation

Figure 2-3: Description of QoS metrics for audio transmission

improve the network quality of service to acceptable values. Dropped calls along with congestion are the two most important customer perceived problems that affect the quality.

2.3 Voice Quality of Service Metrics

Reliable message transfer with error control and notification of non-delivery is common in many modern communication systems. However, the ability to specify timeliness, and the perceived quality of the data arriving is also important, particularly where more complex multimedia are being used. The underlying concepts of bandwidth, timeliness (including jitter), reliability, perceived quality and cost are the foundations of what is known as Quality of Service (QoS).

While systems are often defined in terms of their functionality, QoS defines non-functional characteristics of a system, affecting the perceived quality of the results. In multimedia this might include audio quality, or speed of response, as opposed to the fact that a sound was produced, or a response to stimuli occurred. Much work has been done in the past on establishing the QoS metrics for multimedia[27, 20]. Figure 2-3 delineates the various metrics that apply to audio.

The particular problems of WWAN's highly variable connection quality means that latency, response time and jitter are adversely affected. Our experiments involve uni-directional audio. Thus, we will not be analyzing the response time metric in this thesis. However, latency is a considerable problem for all type of cellular networks (especially WWAN). End-to-end latency is one of the most important performance parameters for a voice communication system. It includes the time the signal takes to traverse the network as well as any delay caused in the hardware (e.g. delay caused by buffering). At present there is some agreement that an end-to-end latency of no more than $250ms$ is acceptable[37].

Jitter is an unwanted variation of one or more signal characteristics. For audio, jitter defines how jerky it is, and how much crackle is there. Cellular voice networks are optimized for low jitter. A robust buffering scheme decreases jitter at the expense of added delay/latency.

Throughput defines the amount of data we can send per unit time. The higher the throughput, the better quality of the audio transmitted. However, the throughput rate of cellular voice networks is set by cellular companies and can not be set independently. To compensate for this limitation we restrict the bandwidth available for audio to 22kbits/s. This value for the bandwidth is comparable to the throughput available on cellular voice networks. Unfortunately this means that throughput is *not* one of the QoS metrics we wil analyze in this thesis.

Reliability is a desired property for any real-time application. For voice communication reliability means a *best effort* network that adapts on-the-fly to any changes in the underlying links.

Cellular voice networks are circuit switching networks, and the notion of reliability involves decreasing the probability of dropped calls. Again, this probability depends on the underlying link and is determined by various factors like the cell tower belong at its capacity, etc. On the other hand, WWAN are packet switching networks and they do not face the problem of dropped calls unless there is no coverage[5]. However, WWAN suffer from the problem of lost packets aggravated by the varying nature of the underlying links. Hence, a voice communication system over WWAN should employ some sort of traffic loss recovery scheme. We discuss such a scheme in the

[5]We assume that the sender is always in an area where at least one WWAN has coverage

next section.

Perceived QoS metrics refer to those audio properties that are distinguishable by a human being. Encoding and audio sampling rate determines the ability of the digital system to recreate the analog audio signal upon replay. These values are set inside the network for cellular voice networks. We set comparable values for the encoding and audio sampling rate when transferring audio over WWAN. High fidelity means how succesful we are when creating the digital audio signal at the receiving end. The underlying links, in all type of networks, determine what frequencies do and do not pass, and which frequencies get attenuated and which get stronger (phase variation). A phase skew distorts the audio making it incomprehensible for the receiver.

2.4 Voice Encoding and Streaming

We are motivated by the need to encode voice data in WWAN at almost the same bitrate and sampling rate as available to cellular voice networks. This section provides some background to the area of audio encoding.

Cellular voice networks make use of optimized speech codecs for voice transmission. Example includes the Enhanced Full Rate (`EFR`) codec used in the `GSM` network. (`EFR`) uses 12.2 kbit/s for speech coding and 10.6 kbit/s for error protection. The compression scheme is based on a sophisticated Code Excited Linear Prediction (CELP) algorithm. Also, `GSM` networks are optimized for transmitting audio sampled at 8kHz[6][46].

Unlike cellular voice networks where the codec is hard coded in the hardware, we have to determine what codec to use for transmitting audio over WWAN. There are several speech codecs available this purpose. We did a comparison of various speech codecs for the experiments including internet Low Bit Rate Codec (`iLBC`)[5], Advanced Multi-Band Excitation (`AMBE`)[2] and `Speex`[12].

We decided to encode our audio with `Speex` for multiple reasons. `Speex` is a royalty free software speech codec that claims to be free of patent restrictions. `Speex`

[6]Referred to as narrowband

is extensively used on Voice over IP (VoIP) applications and podcasts and, unlike other speech codecs, is not targeted at cell phones but rather at VoIP. Similar to the (EFR) codec used in GSM networks, Speex employs the Code Excited Linear Prediction (CELP) algorithm to compress the data[60]. However, Speex is a lossy format, meaning quality is permanently degraded to reduce file size.

Every codec introduces a delay in the transmission. For Speex, this delay is equal to the frame size, plus some amount of "look-ahead" required to process each frame. In narrowband operation (8 kHz) of our system, the delay is $30ms$[60]. This delay is unfortunate, but its value is less than the delay introduced by the other codecs we tested.

Designing for VoIP instead of cell phone use means that Speex is robust to lost packets. However, Speex does not handle the case of corrupt packets but it relies on the User Datagram Protocol (UDP) to ensure that packets either arrive unaltered or don't arrive. WWAN extensively uses link-layer retransmissions and forward-error-correction (FEC) for data packets, increasing packet delays and delay variances, but keeping the bit-error-rate for the channel at a low level.We can also use a robust packet loss recovery scheme to fix the corrupt packets. Such schemes include specialized forward error correction (FEC), interleaving, error concealment, etc[51]. For our experiments we decided to rely on WWAN's retransmissions for lost/corrupted packets. This meant a low overhead and allowed us to determine the variables associated with WWAN without any cover up done by the schemes.

Additionally, with network streams, there are likely to be hard deadlines for the arrival of audio packets. Generally, audio data is useless if it arrives too late into the decoding process at the receiver's end. Modern systems use a *playout buffer* to compensate for delay. Such buffers can help mitigate for network delay, variance in this delay (*jitter*), and even to provide enough time to retransmit lost packets [36].

Chapter 3

Experimental Setup

This chapter delineates the testbed designed for this thesis. Our testbed has two distinct setups, one for analyzing cellular voice networks and the other for analyzing WWANs. Both these setups are discussed below.

3.1 WWAN: Experimental Setup

3.1.1 Hardware

The experiments were carried out using a Linux laptop and hosts on the MIT ethernet. The laptop was running an updated 2.6.x kernel from the Fedora Core 5 distribution. The Linux point-to-point protocol daemon (pppd) was used to establish connections over each WWAN interface.

For the CDMA2000 experiments, we used an Airprime-5220 PCMCIA card, enabled with a Verizon[15] wireless data service plan which used the CDMA2000 1xEV-DO option on the card. The Verizon plan gave us a routable IP address and the ability to send and receive ICMP, UDP, and TCP packets.

Similarly, for the GSM EDGE experiments, we used a Sony Ericsson GC82 EDGE PCMCIA Card, enabled with a T-Mobile[13] wireless data service plan. It also provided us with an IP based network similar to Verizon's.

On the receiving end was a Linux host connected to the MIT Ethernet. Both the sender and the receiver have their clocks synced through an NTP server.

3.1.2 Sending Voice Data

Voice over IP technologies were used to transmit audio through the WWAN links. As mentioned in Section 2.4 `Speex` was used to encode voice data. Input data was an uncompressed PCM file in the WAV container with a sampling frequency of 96kHz and a bitrate of 4608 kbits/s[1]. This data was encoded by `Speex` with a sampling rate of 8kHz and a bitrate of 18kbits/s. These values are comparable to the network values for cellular voice networks.

We decided against keeping a buffer at the sender's end because of the effective use of forward error correction and link-layer acknowledgments in WWANs. Also, for a real-time application like audio, lost packets are abandoned after a certain period of time for the sake of continuity.

`Speex` also provided us with the ability to create Application Data Units (`ADU`) of the same size as of the data packets sent at the link layer. This got rid of added complexity involved with creating complete `ADUs` at the receiving end, especially in the case of lost/corrupt packets.

Work has been done in the past to determine the optimum packet size for transmitting data over WWAN. It has been found that the round-trip-time distribution of small and medium packets (768 bytes and smaller) is different from the round-trip-time distribution for larger packets (1024 bytes or more). The average median round-trip-time for large packets is over $500ms$ larger than the average median round-trip-time for smaller packets and the inter-quartile range for larger packets is $100ms$ less than this range for smaller packets. Also, medium and small packet transmissions are highly inefficient on the link, but for packets larger than 768 bytes, packet size has no impact on throughput.[53].

For the purpose of our experiments we found that packet size of 512 bytes was optimum. We did not experience any sort of inefficient transfer or extra latency, and `Speex` provided us with a better audio encoding for this size compared to 768 bytes.

[1]This low compression meant input data was almost as good as analog sound in quality

3.1.3 Receiving Voice Data

It has been shown that packet loss rate on WWAN links is very low ($\approx 1\%$)[53]. On the other hand, WWAN links suffer from high and variable round trip times, occasional outages and considerable burstiness. Hence, we require a robust buffering scheme on the receiver's end to decrease jitter and have a smooth playback. We also want the ability to turn off the buffering scheme for some of the experiments.

The voice data arrives at the receiver as a User Datagram Protocol (UDP) stream through the Ethernet. Even though Speex provides us with an in-built jitter buffer, we decided to disable that option and implement our own buffer. This way we have more control over the parameters associated with the buffer.

Considerable amount of research has been done on buffering schemes associated with multimedia streaming[51, 36, 55]. Our implementation was optimized for network parameters associated with WWAN[53]. The buffering scheme added an extra $50ms$ to the playback delay (and end-to-end latency) of the system. This may seem too high a delay, however, it was required to counter the burstiness of the WWAN link.

Speex is optimized for fast decoding of the audio frames. For the narrowband operation (8 kHz) of our system, decoding takes $20ms$[60]. This is a small and acceptable delay and its value is less than the delay introduced by the other codecs we tested. Instead of playing back the received data, we save the decoded audio stream on a hard drive with the associated timestamps.

3.2 Cellular Voice Networks: Experimental Setup

3.2.1 Hardware

We used a Motorola RAZR V3m cellular phone, enabled with a Verizon wireless service plan, for the CDMA experiments. Motorola offers a software suite[9] that facilitates interfacing with the phone. For the GSM experiments, we used a Nokia N80 cellular phone, enabled with a Motorola wireless service plan. This phone runs Symbian OS that supports Python applications.

On the receiving end was a host running Windows XP SP2. It was interfaced with two external USRobotics 56K USB Voice Modems which connected to the public switched telephone network (PSTN) through two separate landlines. Both modems featured V.92 enhancements that facilitated call recording by sending uncompressed PCM data to the computer.

The drivers for many internal modems cannot tolerate more than one of the same device inside a single computer (on both Linux and Windows XP). Symptoms of incompatibility include crashes, blue screens of death, or simple inoperability of all but a single modem. External serial modems do not have this limitation because each modem contains its own microprocessor and is unaware of other modems on the same host. USB modems may or may not have this problem, because some USB modems are simply serial modems with a "USB-to-serial" converter chipset (in which case there should be no problem), and other USB modems are "host-controlled" and are essentially externally-attached internal modems (in which case the problem may persist). We chose the external USRobotics 56K USB Voice Modems because they have a "USB-to-serial" converter chipset that allowed us to receive two calls at the same time. Unfortunately the drivers for the modems failed to work on Linux and we were forced to use Windows XP for the experiments.

The sender, the receiver and the Nokia N80 cellular phone have their clocks synced through an NTP server.

3.2.2 Sending Voice Data

For cellular voice networks like `CDMA2000` and `GSM`, the compression scheme for audio transmission is hard coded into the hardware (cellular phones). Transmitting voice data involved sending the voice audio to the cellular phone with minimum delay. We achieved this on the Motorola RAZR V3m cellular phone by connecting it to the computer through USB 2.0[2] and sending uncompressed voice audio to the phone's input. This ensured that all signal degradation happened inside the network. Also,

[2]USB 2.0 offers a bandwidth of 480 Mbit/s which is higher than the bitrate of the voice audio

because of the high transfer rate associated with USB 2.0, there was virtually no delay involved with the transfer of the audio to the phone. It was easier to send voice audio through the Nokia N80 cellular phone. The uncompressed voice audio was transferred to N80's memory. Using the Python bindings to the underlying Symbian OS, the audio was directly fed into the phone input. This eliminated any delay in audio transfer, and any signal degradation before the data entered the cellular network.

Both cellular phones were operated by hand. In case of a dropped call, a new call to the network was placed manually. This process could have been automated on the Nokia N80 but not on Motorola RAZR V3m. We decided to do it manually for the sake of consistence.

3.2.3 Receiving Voice Data

Using external modems allows us the ability to record two incoming calls simultaneously. We originally intended to implement an application to capture data from both the modems and save it on a hard drive. However, a quick search led to many telephone recording software including "Yet Another Telephony Engine" (Yate)[16], "TRx Phone Recorder"[14] and "Modem Spy"[8]. We decided to use Modem Spy for our experiments because it has a small footprint (unlike Yate which is a telephony engine), is free for academic use, and it lets us save the incoming audio as uncompressed PCM (along with the associated timestamps). Thus, no audio compression happens either when the modems transmit the voice data to the computer or when the application saves this data to the hard drive. This ensures that any signal degradation present in the data is due to the characteristics of the network.

3.3 Motion Experiments

Experiments to determine the effects of motion were carried out by placing the laptop inside the MIT "Boston-West" safe-ride (route shown in Figure 3-1). The safe-ride is a shuttle service that follows a fairly constant path, making frequent short stops,

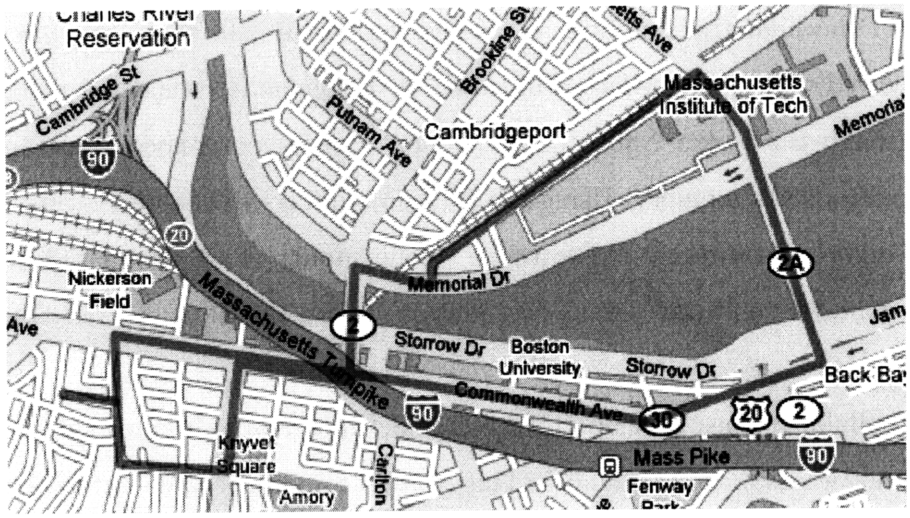

Figure 3-1: The Boston-West saferide route

moving from the MIT campus into Boston over the Harvard bridge, moving east onto the Boston University campus, and then back to MIT over the BU bridge. Since all motion experiments were carried out late in the evening on a weekday, no adverse traffic conditions hampered the motion of the vehicle.

Chapter 4

Experiment Design and Results

This section describes the various experiments we performed using our testbed and their results. As discussed in Chapter 2, the Quality of Service metrics we will be studying are latency/jitter, reliability and perceived QoS. We decided not to analyze bandwidth because it is hard coded into the cellular voice networks. We are also unable to accurately measure response time because our test system is uni-directional. However, latency will provide us with a good estimate about the response time as well.

The aim of our experiments was to clearly differentiate between the two types of networks in terms of the QoS metrics. We were also interested in studying how mobility affected our results because the general user of a cellular network is most likely to be on the move when communicating with someone.

We divide our experiments into two distinct groups. All experiments related studying frequency response and phase variations are grouped together under perceived QoS. Experiments studying delay, jitter and reliability are banded together under latency. We are including reliability in the second group because it is a discrete value (dropped calls) under cellular voice networks, and it gets translated into latency (buffering, retransmissions) in WWAN.

As discussed in Chapter 3, cellular networks encode voice data at $\approx$ 23 kbit/s (e.g., the GSM Enhanced Full Rate speech codec (EFR) uses 12.2 kbit/s for speech coding and 10.6 kbit/s for error correction) with a sampling rate of 8kHz. To have a fair comparison, we encode voice data with `Speex` at 18kHz (with a bandwidth of 5kbits/s overhead for retransmissions) with a sampling rate of 8kHz in WWAN.

4.1 Latency related Experiments

These experiments were designed to establish the inherent latency associated with both types of networks. For a realtime application like voice communication, an end to end latency of more than $250ms$ is not considered good[37]. Our aim in designing these experiments was to minimize the latency by as much as possible. This involved avoiding slow codecs, fast data transfers to and from the network interfaces, etc.

As latency is our biggest concern here, we ignore issues about data quality (they are dealt with in the next section) and concentrate on data arrival patterns. To simplify our task, we input a specialized signal to the networks and study the variations in the output. All of the experiments in this section use the same clip for consistency. The testbed setup remains the same as well; only the way we send and receive our data varies.

4.1.1 Input Data

Our specialized input is a amplitude varying blip signal with a frequency of 2Hz. The reason we chose such a signal was the ease of analyzing latency by comparing the time difference between adjacent blips. By studying these time intervals we can ascertain the latency characteristics of the underlying networks. We vary the amplitude to make it easier for use to differentiate between the different blips on a cellular voice network (we use timestamps for this purpose on WWAN).

We calculate average latency between two blips for each set of 5 blips (varying in amplitude from 0.2 to 1.0), and plot the results for each experiment. This way we can observe how the networks changes over time with respect to latency and reliability.

Note about the graphs

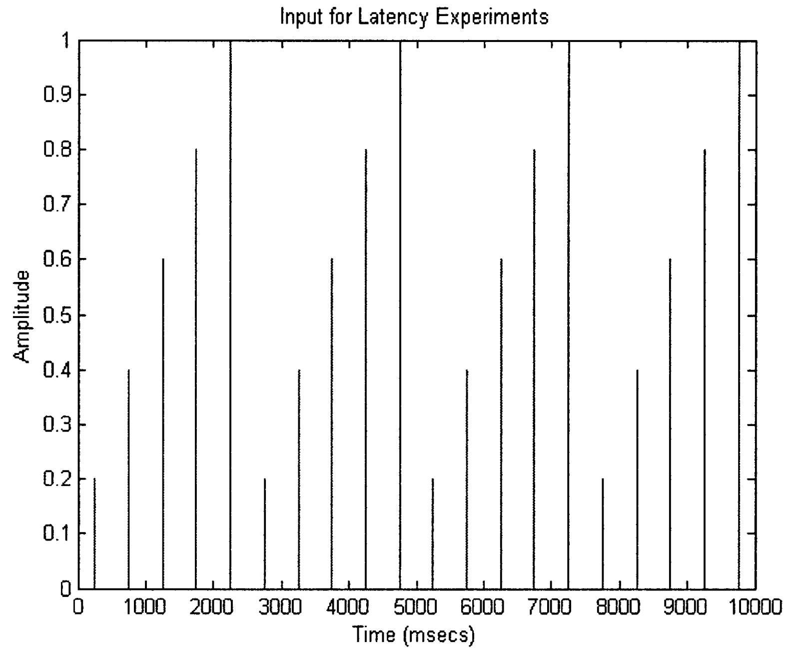

Figure 4-1: Input Data for Latency Experiments

For all the experiments in this section we output graphs plotting the time between blips in the received data against time at the receiver. We chose to represent our results this way because it helps us visualize the QoS metrics we are studying. E.g., in all graphs, the very first spike represents the initial end to end latency of the system. It is the time from the instance a packet is sent by the sender to the instance it is played back at the receiver[1] (or saved to the hard drive). Both the sender and the receiver have synchronized clocks by connecting to an NTP server before hand. After the first spike, the rest of the graph displays the variation in this latency of the system, and also how the system reacts to it.

4.1.2 Experiment 1: Stationary Source, Single Network

These experiments involved transmitting voice data from a stationary source to a stationary destination for both types of networks. We use the results of these experiments as a baseline for the experiments involving motion.

[1]It includes the initial delay caused by buffering for the relevant systems

We run these experiments on single networks (we do not take advantage of network overlap). This is because both WWAN and cellular voice networks perform very well when the source is stationary. Also, latency can not be decreased by using more than one WWAN because of the real-time nature of the application.

Even though our goal is more about comparing the two different types of networks than all the individual networks, we run our experiment on all four networks available to us: `CDMA2000 1xRTT`, `GSM`, `CDMA2000 1xEV-DO` and `GSM EDGE/EGPRS`.

Cellular Voice Networks

For both `CDMA200 1xRTT` and `GSM`, the results of the experiments are attached in Figure 4-2. As we can observe, there is very little difference in the behavior of both networks. There is jitter spread above the $500ms$ line (the actual time interval between blips), but it is at all times below $250ms$.

`GSM` is a little better than `CDMA200 1xRTT` in terms of worse latency. However, the mean latency in case of `CDMA200 1xRTT` ($558ms$) is better than that of `GSM` ($591ms$).

It is also worth mentioning that there were no dropped calls during our experiments.

WWAN

We first ran the experiment without any sort of buffering at the receiver. There was a huge variation in the latency over time, and the whole system was highly unsuitable for a real-time application.

With the use of buffering we were able to steam out the huge jitter involved with the network. However, buffering meant there was an initial $50ms$ of delay added to the latency.

Figure 4-3 graphs the average latencies for both WWAN networks. The graphs have two interesting features. The first is the very small jitter during playback time. This jitter is actually even less than that of cellular voice networks ($\approx 15ms$). The second are the blips spread throughout the graphs. The blips represent the time

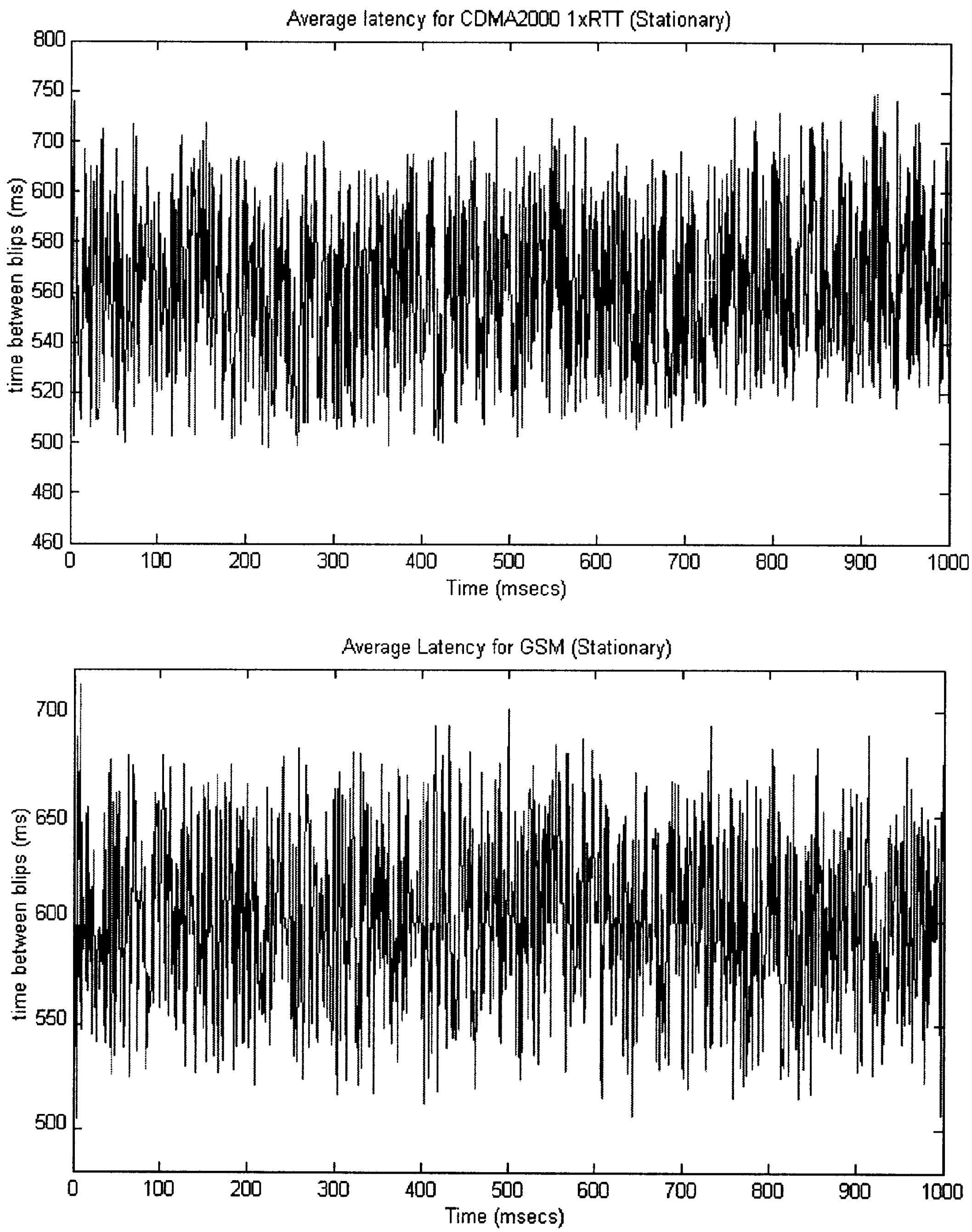

Figure 4-2: Latency for the blips in cellular voice networks with a stationary source.

waiting for the next packet while the buffer is empty. These values are the sum of the latency of the underlying network and a latency of $50ms$ introduced by the buffering algorithm.

These blips introduce huge latencies in the system, worst being $362ms$ in the case of `CDMA200 1xRTT` and 319 in the case of `GSM`. These latencies are more than our threshold of $250ms$.

4.1.3 Experiment 2: Mobile Source, Single Network

As observed in the last section, `CDMA200 1xRTT` and `GSM` have almost the same network characteristics. Similarly, `CDMA200 1xEV-DO` and `GSM EDGE` also have comparable network characteristics. Therefore, we decided to use just one networks from both the groups for this experiment.

This experiment had the same settings as the last except for the fact that the sender was in a mobile vehicle. This was motivated by the fact that most people use cellular networks while on the move.

Cellular Voice Network: CDMA2000 1xRTT

Figure 4-4 graphs the average latency observed from a moving source. There is at least 90 ms of latency in the system. Also, there are times when latency shoots up by a big amount and stays the same for a while. These were the "hand off" times when the phone was negotiating with another cell tower. This graph does not show any dropped calls as that would be represented by a period of no activity. On average, we had 3 dropped calls every 15 minutes.

WWAN: CDMA2000 1xEV-DO

Figure 4-5 shows the results for this experiment. In comparison to a stationary source case, the average jitter in the playback state was much higher ($\approx 90ms$). This was because of the huge variations in the arrival time of packets, the buffer was never

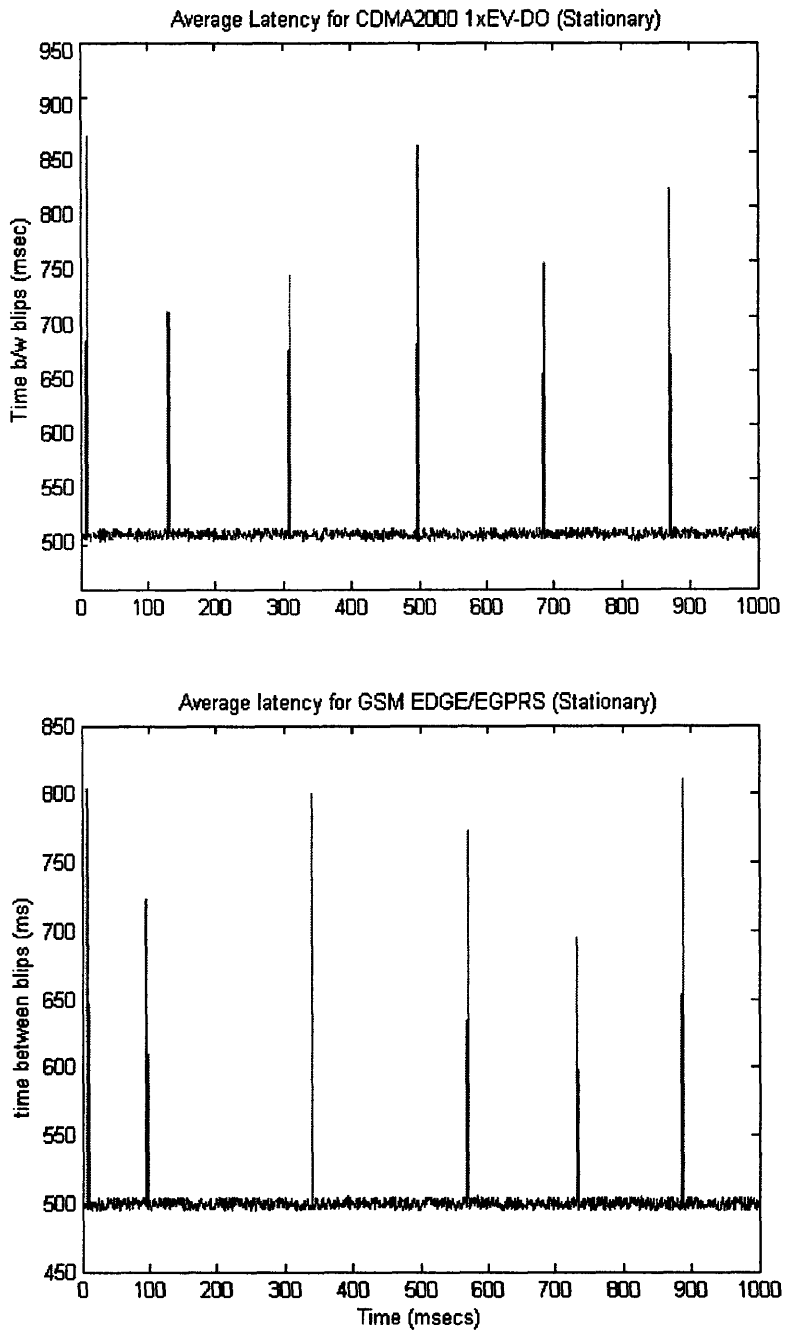

Figure 4-3: Latency for the blips in WWAN with a stationary source.

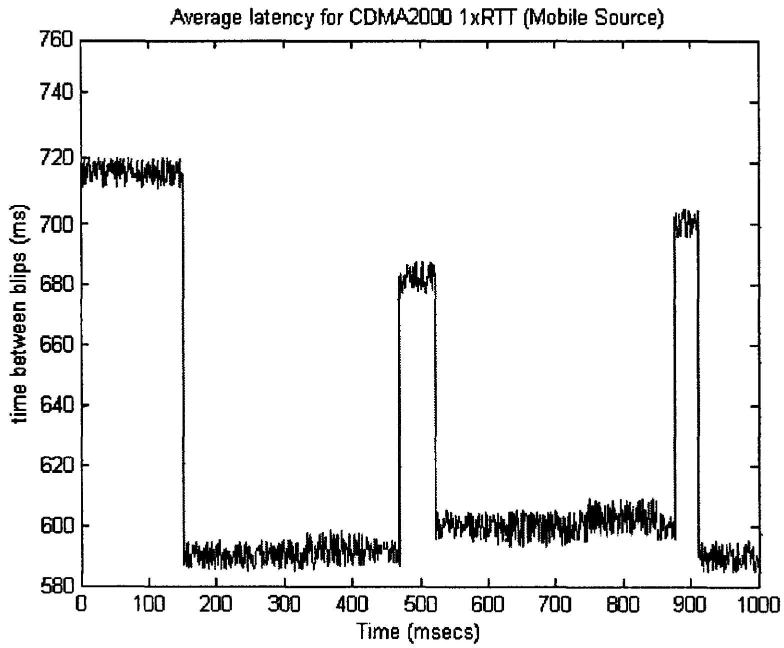

Figure 4-4: Latency for CDMA2000 1xRTT network with a mobile source

actually full. We could have decreased this jitter by increasing the initial buffering time from $50ms$, but decided against that.

On average, the waiting time for the blips also went up. In the worse case, we had a latency of $700ms$ added to the original $500ms$.

4.1.4 Experiment 3: Mobile Source, Multiple Networks

This experiment was similar to Experiment 2, but we sent data on two networks instead of one. Later on, we analyzed the received data merged the two streams into one stream with less errors. Merging was done by comparing both the streams for uniform time intervals, and choosing the "best" of the two for that interval. By "best" we mean the signal with the higher amplitude for cellular voice networks, and available packet with that time stamped for the WWAN.

Cellular Voice Networks

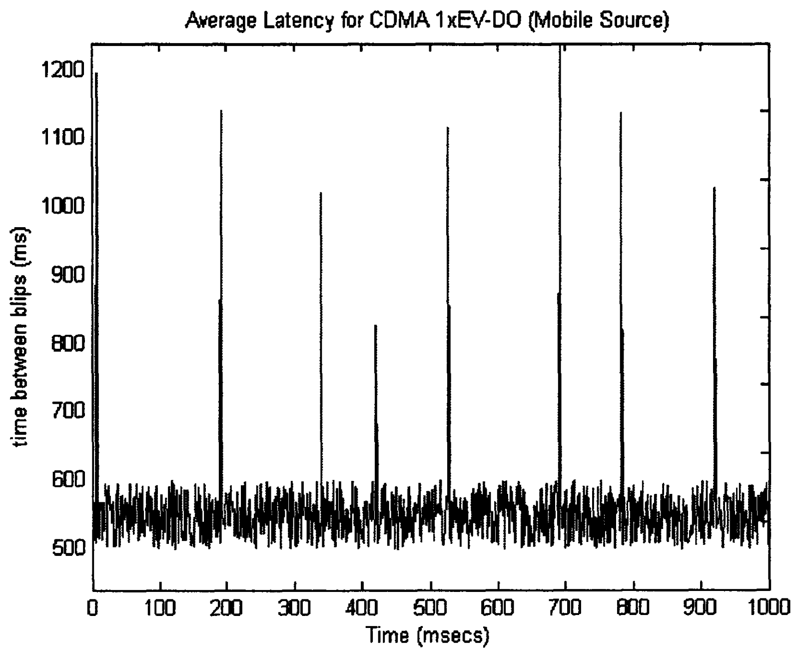

Figure 4-5: Latency for CDMA2000 1xEV-DO network with a mobile source

For this experiment both cellular phones were connected to the laptop through USB. This ensured that the data was transferred through both of them at the same time. The final merged stream (Figure 4-6) was much smoother than the result for a single mobile source. There is jitter, but it's less than 250ms and comparable to the jitter for a single stationary source.

WWAN

Same packets were sent on both the `CDMA2000 1xEV-DO` link and the `GSm EDGE` link. The end result was better than Experiment 2 (see Figure 4-7). The average jitter went down to 15ms, as in the case of a stationary source. The playback waiting periods (the blips) were lower than in case of a single mobile source, but still a lot higher than a stationary source (the worst case being 560ms).

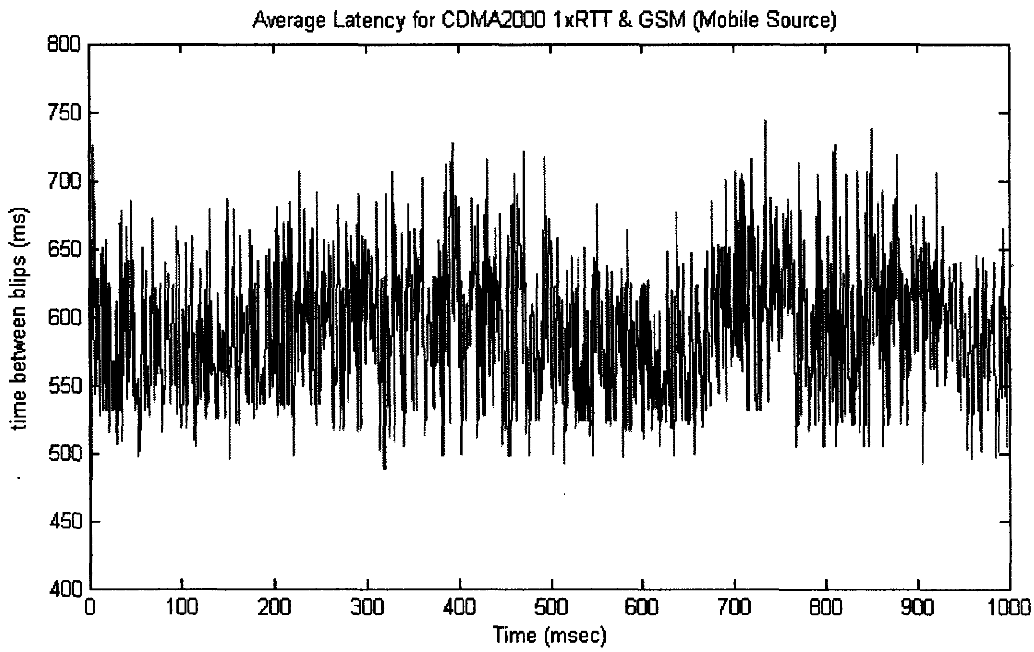

Figure 4-6: Latency for combined CDMA2000 1xRTT and GSM networks with a mobile source

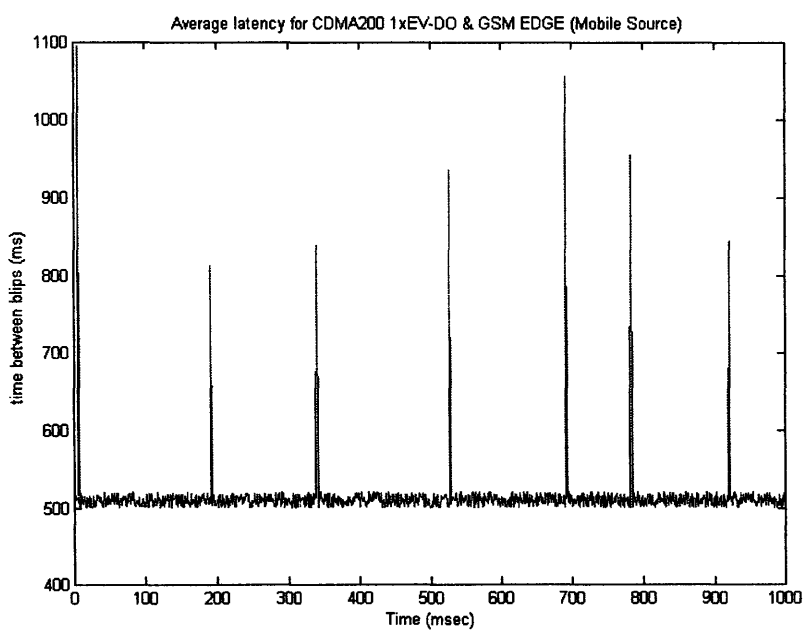

Figure 4-7: Average Latency for combined CDMA2000 1xEV-DO and GSM EDGE networks with a mobile source

4.2 Perceived Quality of Service related Experiments

These experiments were designed to study and analyze the changes in the quality of the audio as noticed by a human being. Humans auditory system is able to distinguish between the different frequencies contained in a sound. Generally referred to as timbre or tone quality, there are certain aspects of a sound that distinguishes it from another sound even if both have the same pitch and amplitude. We quantize this notion of timbre by analysis of frequencies and phase variations in the received audio.

These experiments are geared towards getting the best output and ignoring latency (we covered latency in the last section). Therefore, we will only be comparing the characteristics of the final received audio and not of the data arrival. Also, for all experiments, we will be transmitting data through two networks from a mobile source for both cellular voice networks and WWAN respectively. This section is more about comparing the two different types of networks than all the individual networks.

We utilize the buffering scheme by default for all the experiments over WWAN in this section. As shown in the last section, transmitting audio over WWAN channels without using a buffer at the receiver's end leads to undesirable jittery playback.

All our experiments in this section use the same audio clip for consistency. The testbed setup remains the same as well.

4.2.1 Voice Clip

We use a standard voice clip used by Real Networks in the comparison of their Real Media (RM) format against Microsoft's Windows Media Audio (WMA)[11]. This clip is an uncompressed PCM file in the WAV container with a sampling frequency of 96kHz and a bitrate of 4608 kbits/s. It is a 14 seconds long mono clip of a woman narrating a passage, and contains a variation of pitches as well as silence.

Figure 4-8 graphs the clip in time domain and frequency domain. As can be seen in the frequency domain graph, most of the energy associated with the spectrum lies

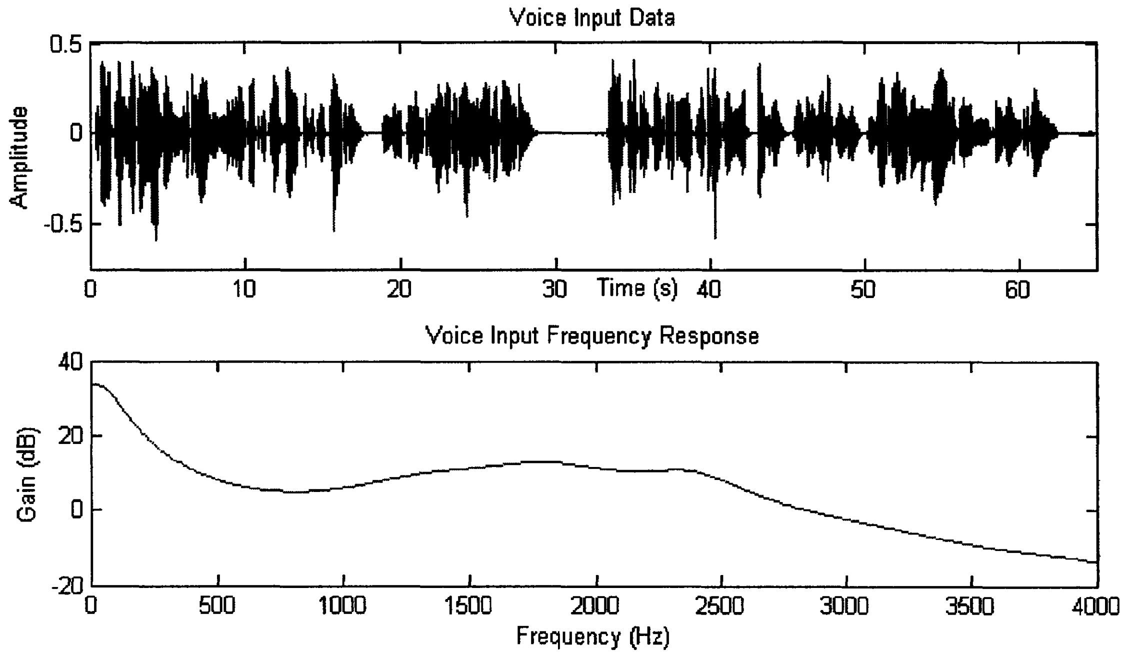

Figure 4-8: Input Voice Data, represented in Time Domain and Frequency Domain

in the to 2.5kHz range.

4.2.2 Voice received over Cellular Voice Networks

We transmitted the same voice data from two cellphones (`CDMA2000 1xRTT` and `GSM`) while on the go. The destination host saved the received data to a hard drive, and later both the received audio streams were merged into one stream with less errors. Merging was done by comparing both the streams for uniform time intervals, and choosing the "best" of the two for that interval. By "best" we mean the signal with the higher amplitude. For all our tries, there was never an instant where both the cell phones had dropped calls.

Figure 4-9 displays the time domain and frequency domain characteristics of the received data. Compared to the original clip, higher frequencies gained amplitude while smaller frequencies were attenuated. This can be observed by noticing the dip in the frequency response graph. This dip occurs at low frequencies while higher frequencies have a higher gain compared to the original clip. Also, there is a a

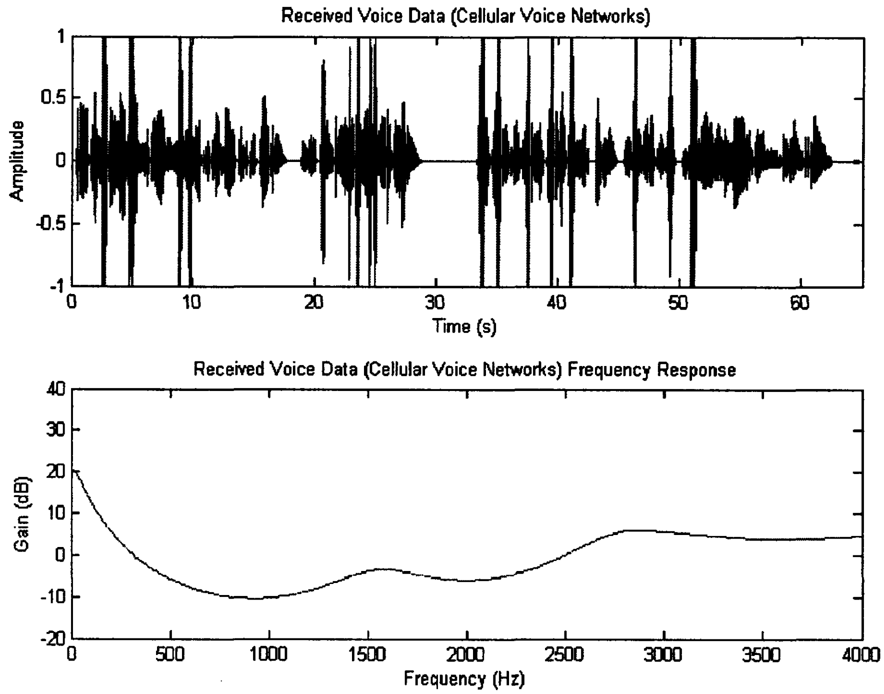

Figure 4-9: Voice Data Received over Cellular Voice Networks, represented in Time Domain and Frequency Domain

considerable amount of lost data noticeable in the time domain graph. It seems that CDMA2000 1xRTT filters out lower frequencies and boosts higher frequencies on its channels.

4.2.3 Voice received over WWAN

Identical voice data packets were sent over a CDMA2000 1xEV-DO interface and over a GSM EDGE/EGPRS interface from a moving vehicle.. The receiver saved both these data streams to hard drive, and later on both these streams were merged to build a final stream. Merging was easy because each data packet had a time stamp and all corrupt packets were already dropped by the system.

Figure 4-10 displays the time domain and frequency domain characteristics of the received data. The output signal is comparable to the original input, though there are certain issues. There is lost data (as observed in the time domain graph), and

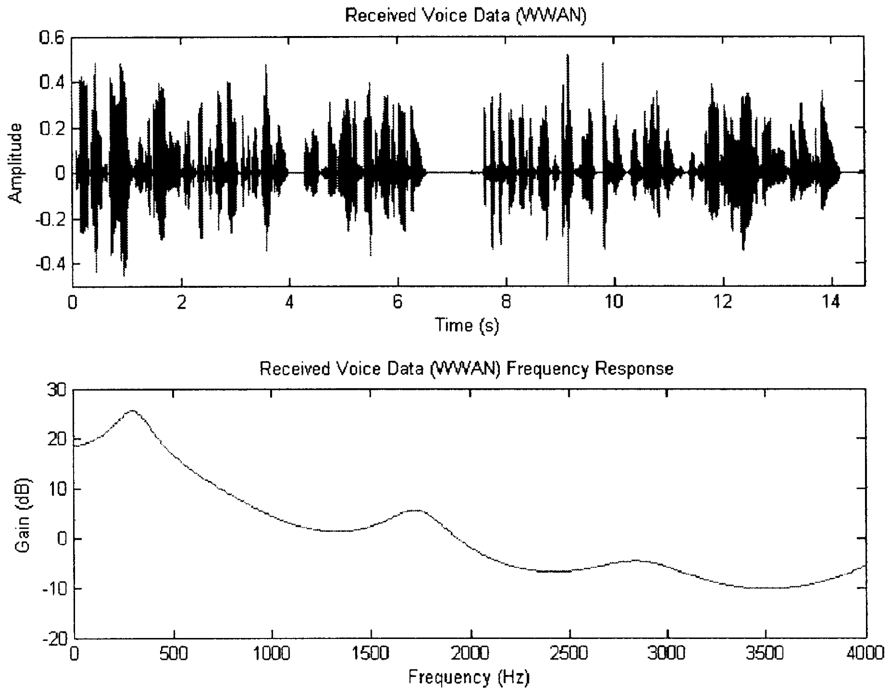

Figure 4-10: Voice Data Received over WWAN, represented in Time Domain and Frequency Domain

even though the signal's frequency response mimics the input's frequency response, it has peaks at certain frequencies ($\approx 310Hz$, $1750Hz$ and $2800Hz$). However, this output is not clipped as was the case with cellular voice networks.

Chapter 5

Evaluation

The preceding chapters have covered the overall design of the testbed including the experiments and their results. This chapter compares and evaluates the obtained results, and tries to ascertain if we were able to achieve our goals.

As described in section 1.3, we set our primary goal to prove that WWAN are capable of voice communication that is of high quality, reliable and has low latency associated with it. We look at each of these properties and determine how WWAN fared against cellular voice networks.

High Quality

Figure 4-9 and Figure 4-10 provide a comparison between a cellular voice network and a WWAN in terms of signal quality. We contrast these figures to Figure 4-8 that graphs the input signal.

Cellular voice networks are optimized for low bandwidth voice communications. This shows in Figure 4-9 where low frequencies are clipped and attenuated, and high frequencies get a power boost. There is also data missing from the output in comparison to the original.

Even though there are compression artifacts visible in Figure 4-10, the overall quality of the output for a WWAN was higher than that of a cellular voice network. Also, the frequency response of the WWAN was similar to the frequency of the input signal. Figure 5-1 graphs the frequency response of the three signals (Input, Cellular

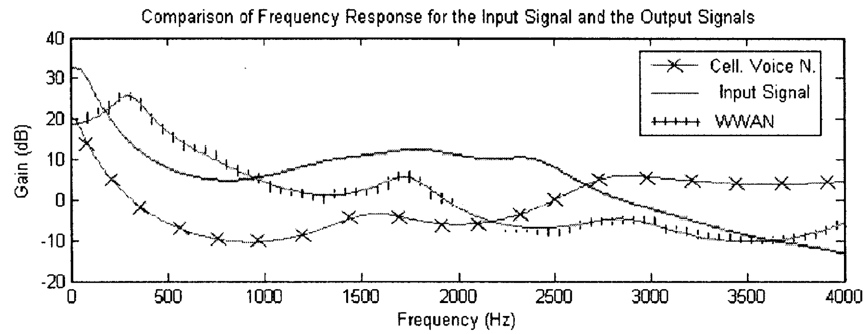

Figure 5-1: Comparison of Frequency Response for the Input Signal, the output of a Cellular Voice Network and the output of a WWAN

Voice Network's output and WWAN output). This is more due to the fact that `Speex` is an efficient codec than to any internal network characteristic.

From the comparisons it is visible that the output from the WWAN reproduces the original signal more faithfully compared to the output from the cellular voice network. Therefore, we can claim that WWAN offers a higher audio quality than a cellular voice network for the same throughput and sampling rate.

Reliability

Our goal in case of reliability was to have a WWAN voice communications system at least as good as any cellular voice network. For most of the experiments both WWAN and cellular voice networks were 100% reliable. The only experiments where we faced a decrease in reliability were the ones involving a moving source. While transmitting data over a single network, the `CDMA2000 1xRTT` network had an outage at a rate as high as 3 dropped calls every 15 minutes. We did not have any sort of complete network outage while transmitting data over `CDMA2000 1xEV-DO`. However, as described in Section 2.3, WWAN have less than 1% packet drop rate. Therefore, the notion of reliability translates into latency for a WWAN. As seen in Figure 4-5, there is an extremely high latency associated with the results. This latency cripples the system making it unfeasible for a realtime voice communications system.

On the other hand, WWAN fared very well in all experiments where the cellular voice network did not have any dropped calls. Even though the latency for WWAN was higher than our accepted threshold ($250ms$), it was the latency inherent in the underlying network, and not caused by reliability problems. We can test this by comparing the results to the result of a single stationary sender (Figure 4-3).

Latency

Latency proved to be the greatest hurdle in our attempt to create realtime voice communication over WWAN. For all of our experiments, the average latency for the WWAN was over $250ms$, our threshold. Even for a stationary source the end-to-end latency was at least $300ms$ (Figure 4-3). Latency got worse for a moving source, getting as high as $610ms$ (Figure 4-5). This was because extra time was spent negotiating transfer between cell towers, which in turn meant lost packets that had to be retransmitted by the link layer. Mobility also introduced a high jitter which meant that the buffer was rarely full. We could have smooth out this jitter by increasing the buffer size / initial buffering time, but we decided not to do so for the sake of consistency.

We were able to decrease the amount of latency for a mobile source by using two WWAN channels. Their overlap meant that while one channel was busy connecting to a new cell tower or faced a very weak signal, we were able to send data over the other channel. this decreased the latency in the network to less than $500ms$.

In comparison, the cellular voice network had a very small latency associated with it. For a stationary source, maximum latency amounted to less than $200ms$ (Figure 4-2). With a mobile source we would get short periods of high latency ($300ms$) and large periods of medium latency ($150ms$) (Figure 4-5). The high latency was caused by the cell tower hand off and could not be avoided. However, by making use of two cellular voice networks, we were able to transmit the data through one network while the other was busy with hand off. This smoothed out the variations in the latency, leaving us with a maximum latency of about $200ms$.

Unfortunately voice communication over WWAN was not able to meet the latency requirement in our experiments. This latency could not be avoided because it was a part of the underlying network. We tried to minimize any extra latency we added to the system ($50ms$ for initial buffering and $50ms$ for encoding and decoding audio frames). Even in the best case scenario (stationary source) the maximum latency was more than our threshold. Making use of multiple WWANs is not helpful because of the realtime nature of voice communication.

Chapter 6

Conclusions

In this chapter we conclude the thesis with a summary of its goals and contributions followed by proposed improvements and directions for future work.

6.1 Goals and Contributions

Our research leverages the widespread cellular wireless wide area data networks (WWAN) to implement a realtime voice communications and tests this implementation against cellular voice networks. Our primary goal was to develop a voice communications system comparable to the cellular voice networks in term of audio quality, reliability and latency.

Our experiments proved that audio of higher quality can be transmitted through WWAN in comparison to cellular voice networks for the same throughput and sampling rate. We used the `Speex` codec for compression purposes, and it yielded better results than the compression codecs hard coded inside cellular voice networks.

Our system was more reliable because of the lack of dropped calls. In case of a mobile mission critical application, WWAN provide better support for getting the data across to the receiver than cellular voice networks. However, the gain in reliability translated to an increase in latency which was detrimental to our system.

Unfortunately, latency was a big problem for the system. In all experiments latency fir the system was not only higher than that of cellular voice networks, but also greater than the acceptable threshold of 250ms. Motion increased the latency by almost twice, though it was reduced by making use of two overlapping WWAN. Latency remains our biggest concern for the system.

This thesis proved that it is feasible and desirable to create a voice communications system over wireless wide area networks, even if there are some latency related issues. This is because it provides us with a higher quality and a more reliable method of communicating, qualities that lack in a cellular voice network. It also showed that even though WWAN channels have high and variable round trip times, considerable burstiness, and much instability when moving, we can still implement stable applications on top of it by using a robust buffering scheme and taking advantage of overlapping networks.

6.2 Future Work

Our implementation of a one-way voice system was more of a proof-of-concept than a deployable, real-world system. Much additional work is possible on developing such a system. The following are few of the obvious possibilities:

Bi-Directional Audio System

We implemented a uni-directional voice system because of a lack of time for the added complexity. It will be a good experiment to study two way communication and to extend our study of latency to response time. WWAN are asymmetrical networks in term of bandwidth and latency, and a study of this asymmetry would be helpful

We can also extend the system to transfer more than just voice audio. There can be many uses for such a system, e.g., transferring medical data (EKG) or transmitting surround sound.

Optimized Codec

`Speex` added an extra $50ms$ to the latency of the system. It is a codec optimized for a high bandwidth network and, thus, is not tweaked for faster encoding/decoding.

A custom codec that not only is faster but is also capable of taking advantage of overlapping networks is very desirable.

On-the-fly Stream Merge

For our experiments involving more than one channels, we saved the received data to hard drive and merged it later to create a better stream. This approach is orthogonal to the idea of a realtime system. A better way of merging the output streams on-the-fly is necessary for a voice communications system.

Bibliography

[1] Advantages of CDMA2000.
http://www.cdg.org/technology/3g/advantages_cdma2000.asp.

[2] AMBE/IMBE Software.
http://www.dvsinc.com/products/software.htm.

[3] ASTRA project.
http://nms.lcs.mit.edu/.

[4] BARWAN: the Bay-Area Wireless Network.
http://daedalus.cs.berkeley.edu/.

[5] internet Low Bit Rate Codec.
http://www.ilbcfreeware.org/.

[6] Massachusetts General Hospital Stroke Service.
http://www.stopstroke.org.

[7] MIT class: Discrete Time Signal Processing.
http://web.mit.edu/6.344/www/.

[8] Modem spy telephone recorder.
http://www.modemspy.com/.

[9] Motorola phone tools.
http://direct.motorola.com/ens/MPT/MPThome.asp.

[10] MPLAYER: the linux video player.
http://www.mplayerhq.hu/.

[11] Realnetworks codec comparison.
http://exoduslabs.net/results/realnetworks/realnetworks.html.

[12] Speex: A Free Codec For Free Speech.
http://www.speex.org/.

[13] T-mobile.
http://www.t-mobile.com/.

[14] Trx phone recorder.
http://www.nch.com.au/trx/index.html.

[15] Verizon wireless.
http://www.verizonwireless.com/b2c/index.html.

[16] Yet another telephony engine.
http://yate.null.ro/pmwiki/index.php.

[17] Hari Adiseshu, Guru M. Parulkar, and George Varghese. "A Reliable and Scalable Striping Protocol". In *SIGCOMM*, pages 131–141, 1996. http://citeseer.nj.nec.com/adiseshu96reliable.html.

[18] D. Andersen, D. Bansal, D. Curtis, S. Seshan, and H. Balakrishnan. "System support for bandwidth management and content adaptation in Internet applications". In *Proceedings of 4th Symposium on Operating Systems Design and Implementation, USENIX*, pages 213–226, October 2000.

[19] J. Apostolopoulos and S. Wee. "Unbalanced Multiple Description Video Communication using Path Diversity".
citeseer.ist.psu.edu/apostolopoulos01unbalanced.html.

[20] Cristina Aurrecoechea, Andrew T. Campbell, and Linda Hauw. A survey of qoS architectures. *Multimedia Syst*, 6(3):138–151, 1998.

[21] Hari Balakrishnan, Venkata N. Padmanabhan, Srinivasan Seshan, and Randy H. Katz. "A comparison of mechanisms for improving TCP performance over wireless links". *IEEE/ACM Transactions on Networking*, 5(6):756–769, 1997.

[22] Deepak Bansal and Hari Balakrishnan. Binomial Congestion Control Algorithms. In *IEEE Infocom 2001*, Anchorage, AK, April 2001.

[23] Ali Begen, Yucel Altunbasak, and Ozlem Ergun. "Multi-path Selection for Multiple Description Encoded Video Streaming". In *IEEE ICC*, 2003.

[24] Lawrence S. Brakmo, Sean W. O'Malley, and Larry L. Peterson. TCP vegas: New techniques for congestion detection and avoidance. In *SIGCOMM*, pages 24–35, 1994.

[25] Joel Cartwright. "GPRS Link Characterisation". http://www.cl.cam.ac.uk/users/rc277/linkchar.html.

[26] R. Chakravorty, J. Cartwright, and I. Pratt. "Practical Experience With TCP over GPRS". In *IEEE GLOBECOM*, 2002.

[27] Dan Chalmers and Morris Sloman. A survey of quality of service in mobile computing environments. *IEEE Communications Surveys and Tutorials*, 2(2), 1999.

[28] Benjamin A. Chambers. "The Grid Roofnet: a Rooftop Ad Hoc Wireless Network". http://citeseer.nj.nec.com/chambers02grid.html.

[29] Ashvin Chedda. A spectral efficiency comparison of umts-fdd and 1xrtt cdma2000 for voice services, 2002.

[30] D. Clark and D. Tennenhouse. "Architectural Consideration for a New Generation of Protocols". In *ACM SIGCOMM*, 1990.

[31] Constantinos Dovrolis, Parameswaran Ramanathan, and David Moore. "Packet-dispersion techniques and a capacity-estimation methodology". *IEEE/ACM Trans. Netw.*, 12(6):963–977, 2004.

[32] J. Duncanson. "Inverse Multiplexing". *IEEE Communications Magazine*, pages 34–41, April 1994.

[33] Dawson R. Engler, M. Frans Kaashoek, and James O'Toole Jr. "Exokernel: an operating system architecture for application level resource management". 1995.

[34] N. Feamster and H. Balakrishnan. "Packet Loss Recovery for Streaming Video". In *the 12th International Packet Video Workshop*, April 2002.

[35] Nicholas G. Feamster. "Adaptive Delivery of Real-Time Streaming Video". citeseer.ist.psu.edu/443706.html.

[36] Nick Feamster. Adaptive Delivery of Real-Time Streaming Video. MEng Thesis, Massachusetts Institute of Technology, May 2001.

[37] D. Ferrari. Client requirements for real-time communication services. *RFC*, 1193, November 1990.

[38] P. Fredette. "The Past, Present, and Future of Inverse Multiplexing". *IEEE Communications Magazine*, pages 42–46, April 1994.

[39] Vijay K. Garg. "Wireless Network Evolution: 2G to 3G". Prentice Hall Communications.

[40] Ekram Hossain and Nadim Parvez. Enhancing tcp performance in wide-area cellular wireless networks: transport level approaches. pages 241–289, 2004.

[41] H. Hsieh, K. Kim, and R. Sivakumar. "a Transport Layer Approach for Achieving Aggregate Bandwidths on Multi-homed Hosts". In *ACM MOBICOM*, pages 83–94, 2002.

[42] H. Hsieh, K. Kim, Y. Zhu, and R. Sivakumar. A receiver-centric transport protocol for mobile hosts with heterogeneous wireless interfaces. In *Mobicom*, pages 1–15, 2003.

[43] M. Jain and C. Dovrolis. "End-to-end available bandwidth: measurement methodology, dynamics, and relation with TCP throughput", 2002.

[44] K. Jarvinen, J. Vainio, P. Kapanen, T. Honkanen, P. Haavisto, R. Salami, C. Laflamme, and J.-P. Adoul. GSM enhanced full rate speech codec. In *Proc. ICASSP '97*, pages 771–774, Munich, Germany, April 1997.

[45] Nancy Joyce. "Building Stata : The Design and Construction of Frank O. Gehry's Stata Center at MIT". The MIT Press.

[46] N. S. Jyant. High-quality coding of telephone speech and wideband audio, 1990.

[47] D. Katabi, M. Handley, and C. Rohrs. "Internet Congestion Control for Future High Bandwidth-Delay Product Environments", August 2002.

[48] L. Magalhaes and R. Kravets. MMTP: Multimedia multiplexing transport protocol, 2001.

[49] Luiz Magalhaes and Robin Kravets. "Transport Level Mechanisms for Bandwidth Aggregation on Mobile Hosts". In *ICNP*, 2001.

[50] J. L. Mitchell, W. B. Pennebaker, C. E. Fogg, and D. J. LeGall. "MPEG Video Compression Standard". 1996. Chapman and Hall, International Thomson Publishing.

[51] Colin Perkins, Orion Hodson, and Vicky Hardman. A survey of packet-loss recovery techniques for streaming audio, September 09 1998.

[52] Jeffrey M. Perloff. "Microeconomics 3rd edition" . Addison Wesley Publishing Company.

[53] Asfandyar Qureshi. Flexible application driven network striping over wireless wide area networks. Master's thesis, Massachusetts Institute of Technology, Dept. of Electrical Engineering and Computer Science, 2005.

[54] Pablo Rodriguez, Rajiv Chakravorty, Julian Chesterfield, Ian Pratt, and Suman Banerjee. MAR: A Commuter Router Infrastructure for the Mobile Internet. In *Mobisys*, 2004.

[55] E. Setton, Y. J. Liang, and B. Girod. Multiple description video streaming over multiple channels with active probing. In *IEEE International Conference on Multimedia and Expo*, 2003.

[56] P. Sinha, N. Venkitaraman, R. Sivakumar, and V. Bharghavan. "WTCP: A Reliable Transport Protocol for Wireless Widea-Area Networks". University of Illinois at Urbana-Champaign.

[57] Alex Snoeren. "Adaptive Inverse Multiplexing for Wide-Area Wireless Networks". In *IEEE conference on Global Communications*, pages 1665–1672, 1999. http://citeseer.nj.nec.com/snoeren99adaptive.html.

[58] Joshua A. Tauber. Issues in building mobile-aware applications with the Rover toolkit. Master's thesis, Massachusetts Institute of Technology, May 1996.

[59] Michael C. Toren. tcptraceroute: an implementation of traceroute using TCP SYN packets. http://michael.toren.net/code/tcptraceroute/.

[60] JM Valin. The speex codec manual. http://www.speex.org/docs/manual/speex-manual.pdf.

[61] Y. Xiao, D. Gagliano, M. P. LaMonte, P. Hu, W. Gaasch, R. Gunawadane, and C. Mackenzie. "Design and Evaluation of a Real-Time Mobile Telemedicine System for Ambulance Transport". *Journal of High Speed Networks*, pages 47–56, September 2000. http://hrfp.umm.edu/Papers/2000/mobiletelemed.pdf.