Tuning Video Redundancy

A Major Qualifying Project Report

submitted to the Faculty

of the

WORCESTER POLYTECHNIC INSTITUTE

in partial fulfillment of the requirements for the

Degree of Bachelor of Science

by

Lisa Lei Zhang

Brandon Ngo

Date: April 24, 2000

Approved:

Professor Mark Claypool, Major Advisor

Abstract

This project analyzes the effects of various redundancy techniques on network congestion and on user perceptual quality. A network simulator was used to simulate TCP and UDP data packets across a network. Various parameters were adjusted including traffic mix, bandwidth, router queue length, and redundancy amount Pre-built movies with various loss and redundancy were used for the perceptual quality user studies. We found no statistical correlation between redundancy and perceptual quality.

Acknowledgements

We would like to thank our Major Qualifying Project advisor, Professor Mark Claypool, for getting us involved with this very interesting and exciting field of Multimedia. We are very grateful for his support and advice that he has given to us on this paper. With his guidance and wealth of knowledge in this area, we are able to overcome many technical hurdles that would have taken us much longer to resolve ourselves.

Also, we would like to show appreciation to Yanlin Liu and Jae Chung, who have supported us in their specialized area of multimedia research. Many thanks goes out to the people who took time out of their busy schedule to help us with our user study and research.

TABLE OF CONTENTS

List of Figures

List of Tables

Chapter 1: Introduction

As technology further evolves, multimedia applications are becoming extensively used in both business and the home. Currently, multimedia applications allow researchers to attend project meetings, seminars and conferences from their desktops; enable students across the world to participate in submarine excursions from their classrooms; and facilitate distance learning by allowing students to remotely participate in lectures [HSK98].

Development in audio transmission came before video packet delivery over the network due to research by telephone companies. In the past, people were able to make further developments in audio transmission than video communications because video requires more system support. Thus the quality of video multimedia needs more development to reach the specification that is accepted by different user applications.

The potential uses for multimedia applications across the Internet are unlimited. It is not hard to imagine Internet related multimedia applications playing a far greater role in everyday life in the very near future. One day, “Videophones” may replace the ordinary telephone. People from remote or isolated areas may be able to attend school or college over the Internet. Movies, shows, news, sporting events, concerts, etc. may all become as easily accessible over the Internet as they are in their current medium.

Transmitting speech across long-haul packet networks date back to the ARPANET and SATNET, which helped launch packet-based multimedia conferencing research. Currently, videoconferencing on the Internet is still in the nascent stages of development and considerable exploration and research remains to be done.

Conducting research in this area to further the development of this growing technology will prove both beneficial to the student and to the field of computer science.

1.1 Multimedia on the Internet

Research in video transmission in the Internet has not been as extensively examined as audio. However, many of the issues facing audio transmissions can be applied to video. Research into audio transmission over the Internet has unveiled various problems including packet loss, scheduling in a multitasking OS, and acoustic problems [HSK98]. Perpetuating the problem of packet loss is insufficient network capacity as web traffic explodes on the Internet.

Because the Internet runs on IP, a best effort service, it is very difficult to develop multimedia applications that are time sensitive. End-to-end delay as well as jitter pose significant problems to quality and need to be addressed. Presently, streaming audio/video having delays of five-to-ten seconds is feasible on the Internet. However, when network traffic increases during peak hours, performance degrades significantly. This traffic spike causes network congestion and packet loss. This packet loss will produce video streams of unacceptable quality to the receiver. Due to congestion on the network and video packet loss, methods of minimizing loss and improving video quality had to be developed.

Current approaches to improve multimedia quality include client-side repair and server-side repair. Receiver-side repair includes insertion, interpolation, and regeneration [PHH98]. It works by having the receiver manipulate data in order to conceal the loss before showing it to the user.

Sender-side repair includes retransmission, interleaving, and forward error correction. This type of repair can be either active or passive. In active repair, the sender waits for acknowledgements from the receiver. Upon timing out, the sender will resend the packets, which it assumes to be lost. Passive sender-side repair techniques include forward error correction and interleaving. Forward error correction sends repair data to the client in order to compensate for data lost. However, this introduction of redundancy increases the network load, possibly resulting in performance degradation. Interleaving works by reshuffling the order of packets. The idea is that multimedia quality will not be affected as much during bursty loss since the data has been sufficiently distributed [PHH98].

Network Simulator (NS) was used to test the effects of redundancy and Group of Pictures on network congestion and data loss. MPEGs are generally divided up into separate frames called "Group of Pictures." GOP is essentially the manner in which all MPEG's are encoded and decoded. Redundancy is a technique that ameliorates the effects of packet loss by attaching a lower quality frame of the previous frame onto each frame that is sent across the network. In the case where a packet is dropped or lost, the proceeding packet will contain a copy of the loss frame. The copy will be of a lower quality to reduce the amount of data sent across the network. Because this lower quality frame can replace the lost frame, the perception of video transmitted over a network is not as degraded. Frames are typically lost during times of heavy network congestion or usage.

Congestion on the network is another variable that was varied to determine the effect on the efficiency and effectiveness of the scheme under various loads. Congestion on a computer network is analogous to traffic congestion in the sense that when too much data is sent across a network of limited bandwidth, movement slows down significantly. Unlike traffic congestion, congestion on a network sometimes results in data being dropped from the network altogether. Congestion often occurs at routers that may be classified as "bottlenecks." This occurs when the router becomes inundated with incoming data and cannot handle the sheer amount of data given to it. When data is dropped during times of heavy congestion, perceptual quality of transmitted video can be affected significantly.

Perceptual quality is essentially how a user views the quality of something he/she is viewing. In the case of videos, the more smooth, clear and crisp the movie is, the higher the perceptual quality of the movie will likely be. In the user study, perceptual quality of movies was measured quantitatively by having users view and rate 27 different movies. The effectiveness of different redundancy schemes was determined by having users rate movies of various redundancy schemes. Users rated the movies on a scale of 1 to 100 based on whether they felt the movie was of high or poor quality.

1.2 Currently Used Protocols

The Internet relies on the TCP/IP protocol, which is part of the transport layer of the 7-layer OSI model. TPC/IP provides data transmission verification between client and server. With TCP, a connection is established between client and server before data is sent. Once the transmission is complete, the connection is terminated. The reliability of TCP stems from its use of acknowledgements and retransmission. TCP in turn, relies on the services of the IP protocol.

IP is part of the network layer and is responsible for moving data packets from node to node by decoding addresses and routing data to their destination. IP can be used to allow computers to communicate across a room or across the world [tcp]. IP is a "best-effort" service, which means that it attempts to move datagrams from sender to receiver as fast as possible. However, end-to-end delay and jitter cannot be controlled. TCP/IP is composed of four layers:

- Application: includes all the higher level protocols such as TELNET, FTP, SMTP, DNS, HTTP, etc.
- Transport: TCP, a connection oriented protocol resides within this layer. It is responsible for verifying the correct delivery of data from client to server by invoking retransmission upon detection of data loss.
- Internet/Network: responsible for delivering IP packets to where they are supposed to go (i.e., packet routing).
- Host-to-network: the host connects to a network using some protocol so it can send IP packets over it.

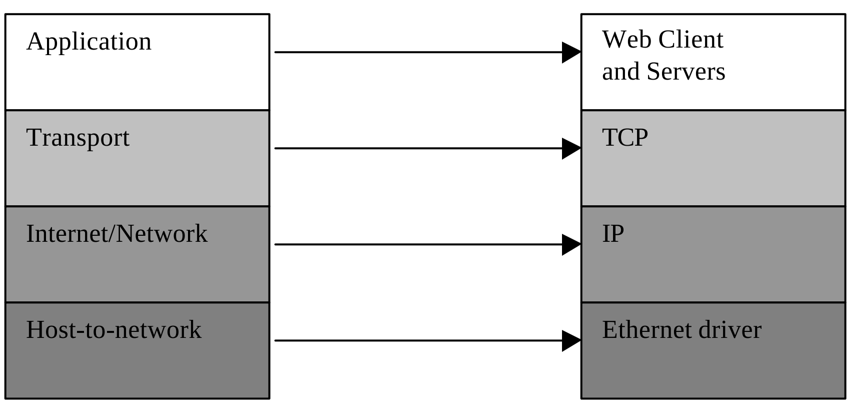


Figure 1.2: Part of the OSI model

Residing in the transport layer, UDP (User Datagram Protocol) provides an unreliable and connectionless protocol for applications that do not require TCP's sequencing and/or flow control. According to RFC 768, a UDP segment is structured as defined in Figure 1.2.2. The source port indicates the sending process and represents the location to which any replies need to be sent. The destination port allows the correct application to receive the data that is transmitted. Length represents the datagram's header and data combined size in octets. The UDP checksum ensures that the transmitted data is uncorrupted by recording the one's complement of the sum of all the 16 bit words in the datagram [POS80].

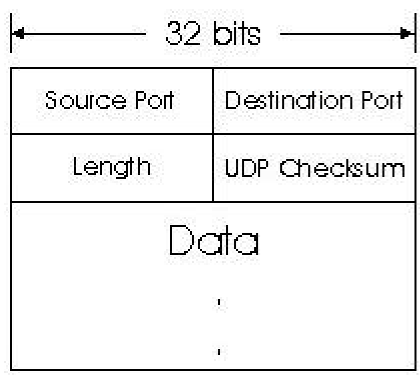


Figure 1.2.2: UDP segment structure

UDP is most commonly used where speed is more important than reliability [TAN96]. Internet phone, real time video conferencing, streaming video/audio, NFS, SNMP, and DNS are examples of applications that would all be better implemented with UDP [KR00]. Transmitting video or audio across the Internet can have dire consequences if used in conjunction with TCP. TCP's built in congestion control would slow down the transmission of data in times of heavy traffic resulting in poor quality video or audio quality.

UDP has certain advantages over TCP that make it a better alternative in various situations [KR00]. UPD has:

- *No connection establishment* - Unlike TCP, UDP requires no preliminary "handshaking" before data is exchanged. This significantly reduces waiting time since no time is needed to establish a connection.
- *No connection state* - Since reliability is not an issue, UDP does not maintain any connection state nor any of the parameters associated with the state. These parameters include receive and send buffers, congestion control parameters, and sequence and acknowledgement number parameters.
- *Small segment header overhead* - Overhead for a UDP segment is only 8 bytes as opposed to the 20 bytes for a TCP segment.
- Unregulated send rate - Unlike TCP, data transfer rate using UDP is only constrained by factors such as the application's ability to generate data to be sent and bandwidth. When network congestion rises, data transmission does not slow down; rather, a minimum send rate is maintained.

The lack of congestion control associate with UDP is a double-edged sword. Although the result is faster data transmission, a network that is being inundated with data from multiple UDP transmissions may result in queues at routers filling up and losing data. One possible solution to this problem, which has been the subject of much research, is adaptive congestion control [KR00].

Chapter 2: MPEG

This project will deal with MPEG's to a large extent. MPEG (Motion Picture Expert Group) is the name given to a family of International Standards used for coding audio-visual information in a digital compressed format [mpo]. The MPEG family of standards includes MPEG-1, MPEG-2 and MPEG-4.

The goal of MPEG-1 was to produce video with quality equivalent to a VHS videotape recorder using a bit rate of 1.2 megabits per second. Its purpose was to serve as a format for digitally stored media. MPEG-2 is a slightly more advanced format providing a resolution of 720x80 and 1280x720 at 60 fps having CD-quality audio. Able to handle data rates below 10 Mbit/second, MPEG-2 is the format typically used on DVD's and digital television [web99]. MPEG-4 is based on the Quicktime file format and is serves as a standard for multimedia applications. MPEG-4 addresses various key issues such as ease of accessibility in heterogeneous and error prone network environments and compression efficiency [SIK97].

MPEG records only key frames and predicts what the missing frames look like by comparing differences between the key frames. MPEG works differently than other video compression formats currently on the market. In addition to compressing individual frames, MPEG also compresses between individual frames of a video sequence.

2.1 Frames Types

MPEG streams are composed of three major frame types.

- I-Frames (Intracoded)

- P-Frames (Predictive)
- B-Frames (Bidirectional)

I-frames are self-contained still pictures that must appear regularly in the stream (e.g., every half second) and are needed to decode P and B frames. P-frames contain block-by-block differences with previous frames [GKL+98]. In other words, P-frames require information from previous I-frames and/or all information from previous P-frames. B-frames contain differences with the previous and the next frames. Therefore, they require information from both the previous and following I and/or P-frames (See Figure 2.1). The compression rate, which determines video quality is highest for B-frames while lowest for I-frames.

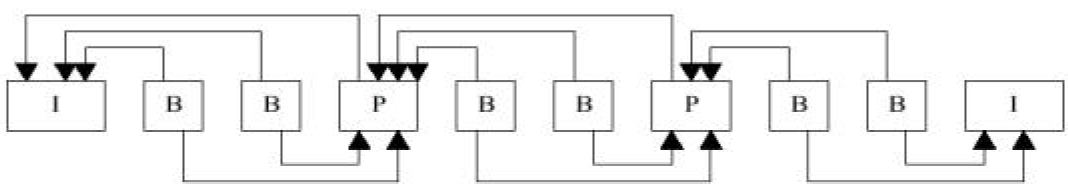


Figure 2.1: Relationship between I, P and B frames

2.2 Group Of Pictures (GOP)

An MPEG encoder stores only the complete picture of the baseline frame (i.e., I-frame) and partial pictures of any subsequent frames. It does so by breaking the video sequence up into GOP (Group Of Pictures). Each GOP generally contains 15 frames and has an I-frame at the beginning. Therefore, I-frames are composed of the first frame in a video sequence and numerous other “baseline” frames within the video stream. Frames following the I-frames are analyzed and only the differences between it and the I-frame are compressed. This increases the compression performance. The Group of Picture pattern that was used in building the movies for the user test was IBBPBBPBB [LC99].

Chapter 3: Related Work

3.1 Multimedia Quality

Quality is a central issue to multimedia applications. Factors that could impact quality include latency, jitter and data loss. Quality can be measured through objective means such as jitter or data loss. It may also be measured through subjective means such as performing user studies.

Claypool and Riedl have listed three basic measures that determine acceptable video quality - latency, jitter, and data loss [CR99]. Latency is the time it takes for data to be successfully transmitted from the source to the destination, and it may cause unacceptable delays between the time of the actual event and the reception of the data. Jitter is the variance in latency. Jitter causes video streams to have unevenness between frames, and can result in an unnatural flow of graphics. Data loss can either be voluntary from bandwidth limitations, or involuntary due to the problems in the transmission medium, but the end result is the same. Smooth video presentation and information of critical importance is lost and is unacceptable. These three criteria are objective in nature. They are obtained from system analyses and do not require user opinion, which may vary.

Watson and Sasse agree that video quality can be measured objectively but instead, they chose a subjective approach [WS95]. Their approach, Mean Opinion Scores (MOS), is conventionally used in speech assessments. This is a subjective rating system based on user opinion. However, they question the applicability of the use of this rating system for video due to the low transmission rate of video data over the Internet, and that the perception of video quality is often psychological. Perceptual quality of the video may be improved when audio is used as a complement to the multimedia presentation, although the physical pictures of the video were not altered. They stress that user opinionated evaluation must be carefully considered due to the subjective nature of this type of rating.

Similar to other research papers that try to measure multimedia quality, this paper will use both an objective and subjective way to quantify its effects on the Internet and users. Hopefully, our redundancy with group of pictures (GOP) and spacing techniques will be able to improve both perceptual quality and reduce network load. Systems data was collected and analyzed. From examples set forth by previous works [LC99], we are able to build a user interface that will help us collect user preferences of different video quality.

3.2 Repair Techniques

There are two types of repair techniques that were used for audio, which were modified to be used with video. They are sender-based and receiver-based repairing techniques [PHH98][LC99]. Their intentions are to improve perceptual multimedia quality. We used a sender-based technique.

3.2.1 Sender – Based Repairs

Sender-based repairs are repairs driven by the sender.

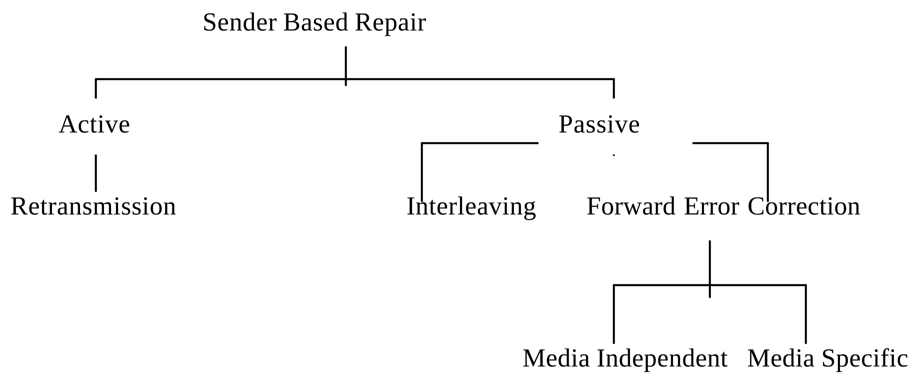


Figure 3.2.1 Taxonomy of Sender-Based Repair Techniques [PHH98]

Sender side repairs can be broken down into two categories - active retransmission and passive channel coding (refer to Figure 3.2.1). Passive channel coding is done on a "best-effort basis." The sender sends the repair data and does not require an acknowledgement that the receiver has received the information correctly. With active retransmission the sender can participate in repairs after knowing the receiver had problems getting the correct data. Sender-based repairs can be further subdivided into Retransmission, Interleaving and Forward error correction.

A. Forward Error Correction (FEC)

Forward error correction techniques require the addition of repair data or redundancy to a stream. This additional information allows lost data to be repaired on the client side. Under FEC are two classes of repair data - media independent and the media specific.

a. Media Independent FEC

Media independent FEC does not require knowing what is in the contents of the stream. Block or algebraic codes are transmitted to help repair what was lost. There are k data packets in a codeword and n-k extra check packets are transmitted for n packets that needs to be sent over the Internet.

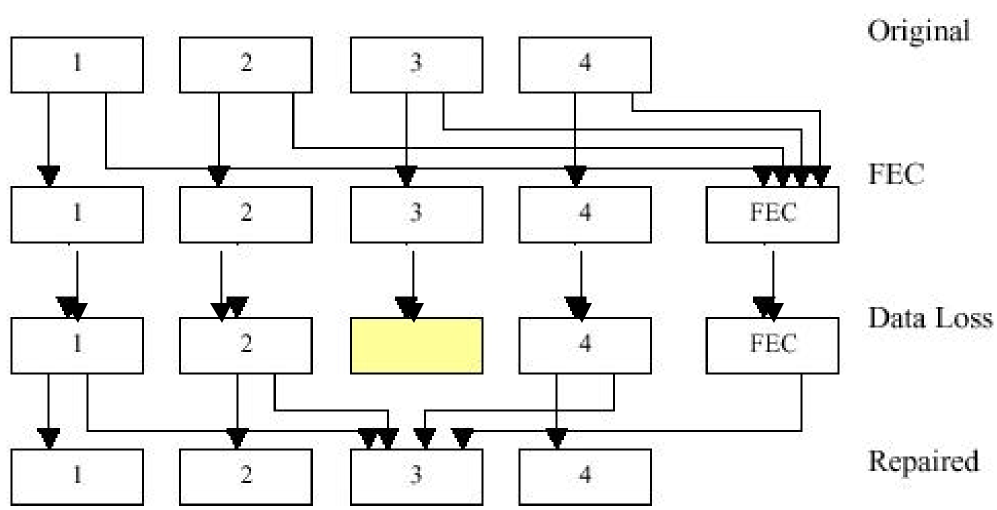


Figure 3.2.1.2: Repair Using Parity FEC

Parity coding and Reed-Solomon coding are two block coding design (refer to Figure3.2.1.2) [PHH98]. Error correction data bits generated from either parity coding or Reed-Solomon's repair technique is used to help check for entire packet lost, rather than single bit errors. The two block coding schemes are used across the corresponding bits in blocks of packets.

In parity coding, one parity packet is sent after every n-1 data packets. These parity packets are generated using the exclusive-or (XOR) operation across groups of packets. Only one recovery can be made out of n packets by using this design.

The properties of polynomials are used in Reed-Solomon codes. The codewords are coefficients to a polynomial that is transmitted. The codeword is found by taking all the non-zero values of x over the number base. Although this is straightforward to encode, decoding may take longer and be more costly.

There are benefits to these FEC media independent repair designs, such as independence of packet contents and replacement of lost packets. Also computation that is required by these two schemes are small and simple to implement. However, they produce additional delay, increase bandwidth and require a more complex decoder.

b. Media Specific FEC

Media Specific FEC safeguards against packet loss by transmitting each frame multiple times. When a packet is lost, one of its duplicates is able to replace it (refer to Figure 3.2.1.3). The first transmitted copy of audio or video information is the primary encoding because it has the best quality. Copies of this package is the secondary encoding because the sender is able to decide if the quality or bandwidth of this packet should be the same or lower than the primary encoding packet. There is some complexity when it comes to deciding on choices for encoding, such as between bandwidth requirements and the computational complexity of encoding. We used media specific FEC.

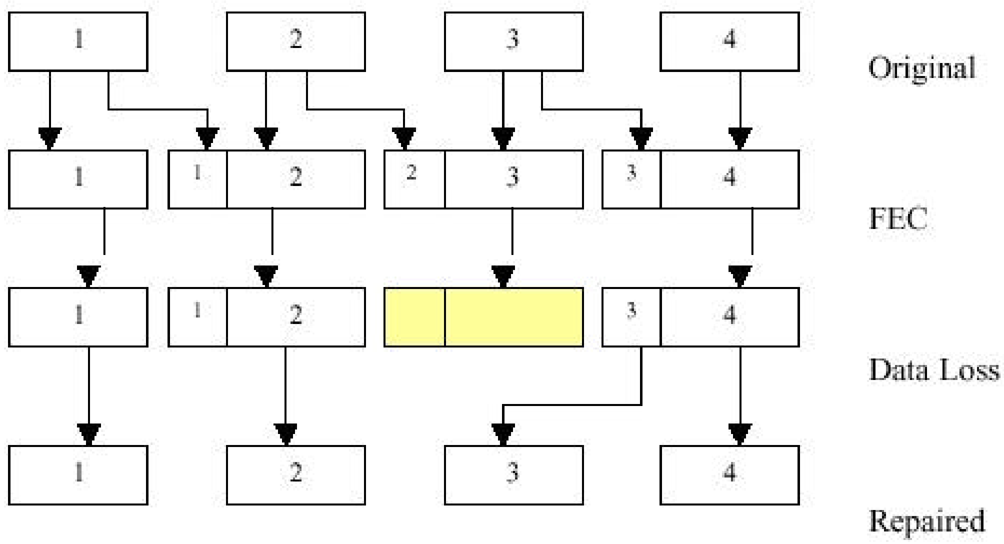


Figure 3.2.1.3: Repair Using Media Specific FEC [LC99]

B. Interleaving

Interleaving uses the method of spreading out each data packet into different units. On the sender side the original packets are split up and reshuffled. Each original packet is then reassembled on the receiver side (refer to Figure 3.2.1.4). This method works well because when one unit is lost, the effect on the overall picture or sound presentation will not have a drastic effect on users' satisfaction. Interleaving data packets may be a good way to do repair on audio and graphic frames without introducing more overhead on the network. Because of latency, this method is not effective when interactive multimedia is needed. It takes a longer duration of time for the receiver to put together the original packet because it is waiting to receive all the pieces to it before reassembling the whole presentation together.

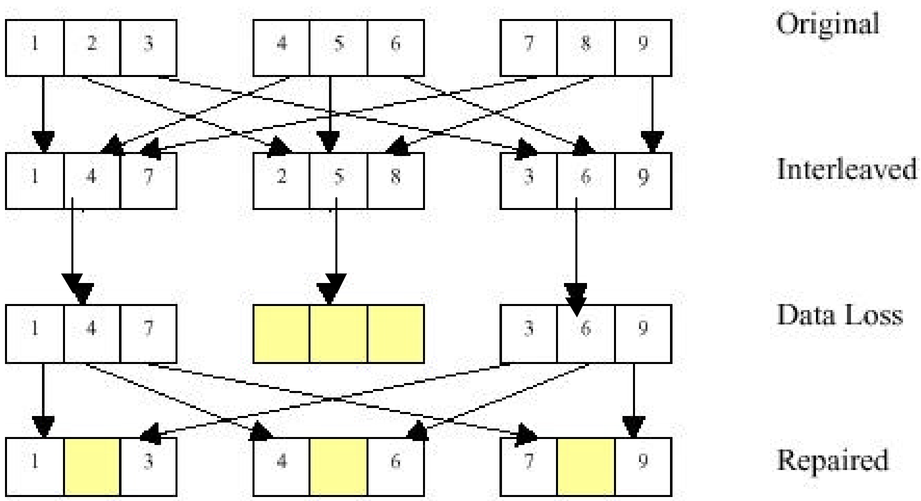


Figure 3.2.1.4: Interleaving units across multiple packets.

C. Active Retransmission

In active retransmission the sender transmits packets by using a reliable multicast or unicast scheme and waits for an acknowledgment from the receiver. The sender will resend the packet after it times out waiting for the acknowledgement from the receiver. Similar to interleaving schemes, a retransmission scheme would not be a good choice when interactive multimedia is required due to the delay time of re-sending packets. This method guarantees the deliverance of information.

3.2.2 Receiver – Based Repairs

Receiver - based repairs are done by the receiver side. It does not require the help of the sender (Figure 3.2.2). Another name for receiver-based repairs is Error Concealment. The receiver uses techniques to conceal data loss before showing the transferred information to the users. Error concealment techniques fall under either one of these three categories: insertion based repairs, interpolation based repairs and regeneration based repair.

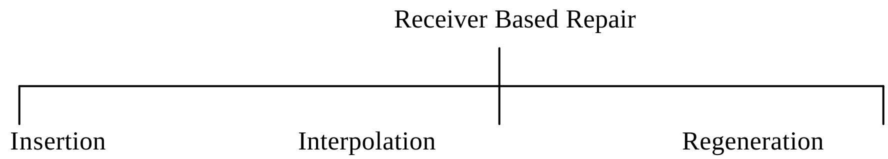


Figure 3.2.2 Taxonomy of Receiver-Based Repair Techniques [PHH98]

In this research project, Media Specific Forward Error correction was used. There are several ways in piggybacking frames in groups of pictures (GOP), which will be discussed more thoroughly in section 5.1. This type of repair technique was selected because the sender is given the control of delivering the repair packets. The quality of an MPEG stream needs to be maintained throughout the process of sending and receiving the data packets.

Chapter 4: Measuring Network Congestion

4.1 Measuring Overhead From Redundancy

``When too many packets are present in (a part of) the subnet, performance degrades. This situation is called congestion.'' - Tanenbaum

When routers are congested, packets may be lost. That is, when packets are arriving at routers so fast that the routers cannot handle them all, the packets are simply dropped. This is the primary reason for packet loss on the Internet. Since packet loss is what degrades video quality of multimedia applications across the Internet, this is one of the primary issues that needs to be looked into and addressed.

Conducting accurate traffic analysis on the Internet is very difficult at best given the random arrival rate and variable loading of datagrams. In addition, many other network load factors during any given time further complicates matters. Therefore, according to Tim Bass, the best estimates of network congestion are “ascertained from hundreds (perhaps thousands) of samples averaged over long test intervals” [BAS98].

Existing tools used to measure RTT on the Internet and hence network congestion include the PING and Traceroute utilities. The Ping utility essentially sends an ICMP “echo request” as a data packet to a remote host. Once it completes, it provides brief round trip time and packet loss data. Traceroute allows users to check on the route to a machine to check for problems on the route. It allows users to view the route an IP packet takes to get to its destination and it determines the amount of time the packet takes to travel the route [ns].

For the purposes of this project, network congestion was measured by looking at the frame size, frame delay time, throughput and measurement of how many packets are dropped at various routers within NS. Furthermore, the units of time datagrams take to reach their destination was also used as an indication of network congestion.

Chapter 5: Network Simulator (NS)

This project deals with setting up and running a simulated network and varying parameters to analyze its effects using NS. Developed at UCLA Berkeley, NS is an event driven network simulator, which simulates a variety of IP networks and implements many network protocols such as TCP and UDP [ns]. Parameters that can be varied include the size of traffic, the frequency at which packets are being delivered and the number of connections. Furthermore, NS is able to be used to set up a topology with various specifications. We can define the number and placement of routers, bandwidth of each path, queue size of a router and router queue management.

5.1 Procedure

From the network parameters listed above, this project focuses on traffic size, frequency at which packets are being delivered and the number of connections. With respect to topology, we researched the placements of routers and the bandwidth of our simulated network. To accomplish this, we wrote an OTcl script, which served to initiate an event scheduler, set up the network topology, and started/stopped transmitting packets. Once the simulation had completed, data was gathered from a text based output file generated by NS; this must be specified in the input Otcl script [ns]. The basic interface of NS appears as in Figure 5.

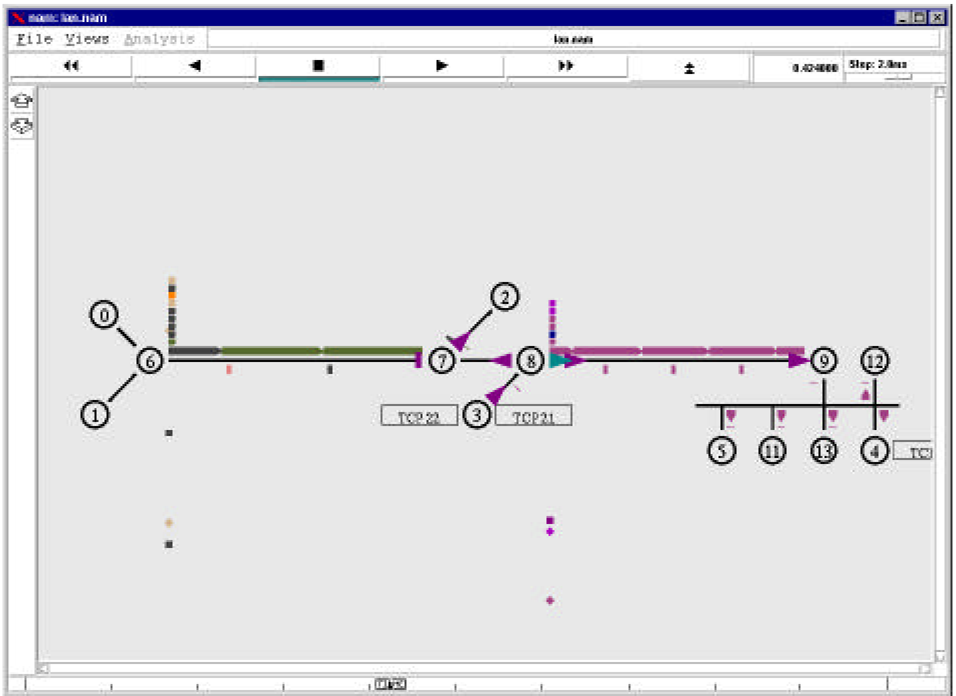


Figure 5: NS Interface

5.1.1 Topology

We initially analyzed video transfers through a simple topology shown here in figure 5.1.1.

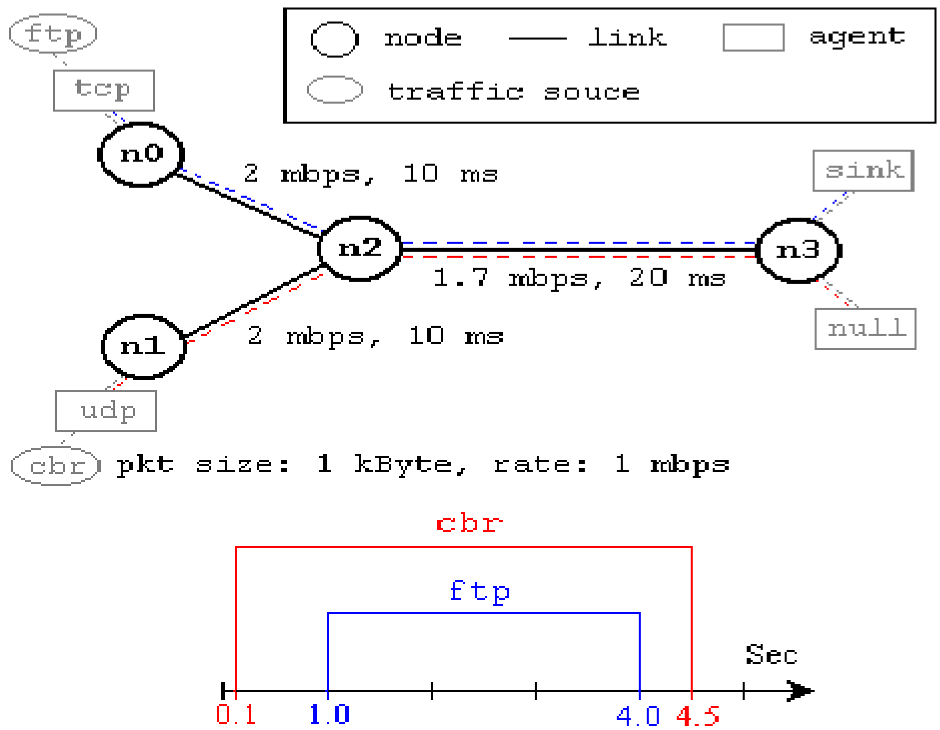


Figure 5.1.1: A Simple Topology [ns]

We looked at several types of topologies before choosing one topology as our model to analyze. Parameters that were varied were:

- network load
- router type and location
- data stream type (e.g., UDP, TCP)
- data packet size
- data transfer speed

Using the simple topology shown in Figure 5.1.1, bandwidth and queue size at router n2 was varied. After reviewing the resulting data, an expanded topology (i.e., Topology 2) was created (see Figure 5.1.1.2). The main difference between the simple topology and topology 2 is the addition of one more TCP stream. For topology 2, we varied the start and end time of the three senders. Different queue sizes and bandwidths were also modified and explored.

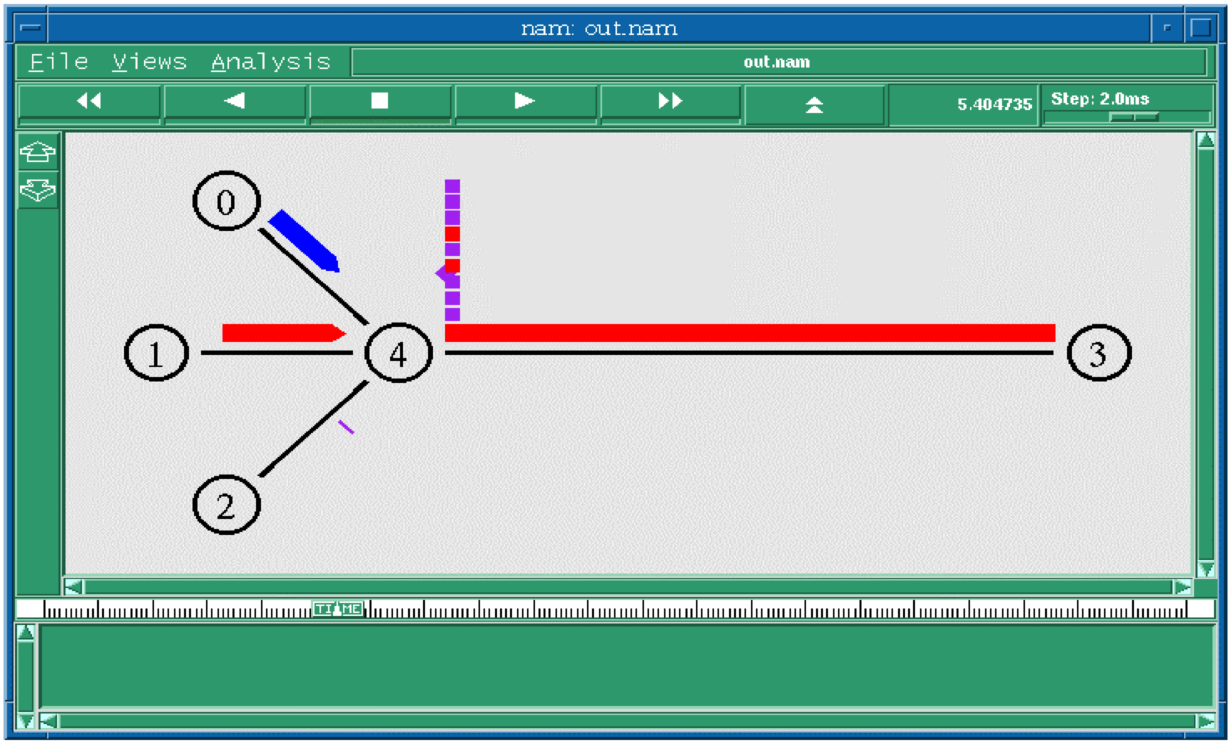


Figure 5.1.1.2 : Topology 2

In Figure 5.1.1.2, nodes 0 and 2 are TCP connections whereas node 1 is a UDP connection. We were able to send a multimedia stream over a UDP connection from node 1 to node 4 using a protocol written by Jae Chung [CC200]. Essentially, the protocol extended NS's capabilities by allowing it to read in an input file (e.g., mpeg_trace.input)and use varying packet sizes. Node 3 is a receiver that returns acknowledgment messages back to nodes 0 and 2. The size of the RED queue at router 4 was initially set to 5 and then 10. However, with the small queue size parameter, the queue performed similarly to a Droptail queue.

After simulating other topologies, Topology 3 was selected (see Figure 5.1.1.3) as the final model to analyze. The parameters used on Topology 3 are shown in Table 5.1.

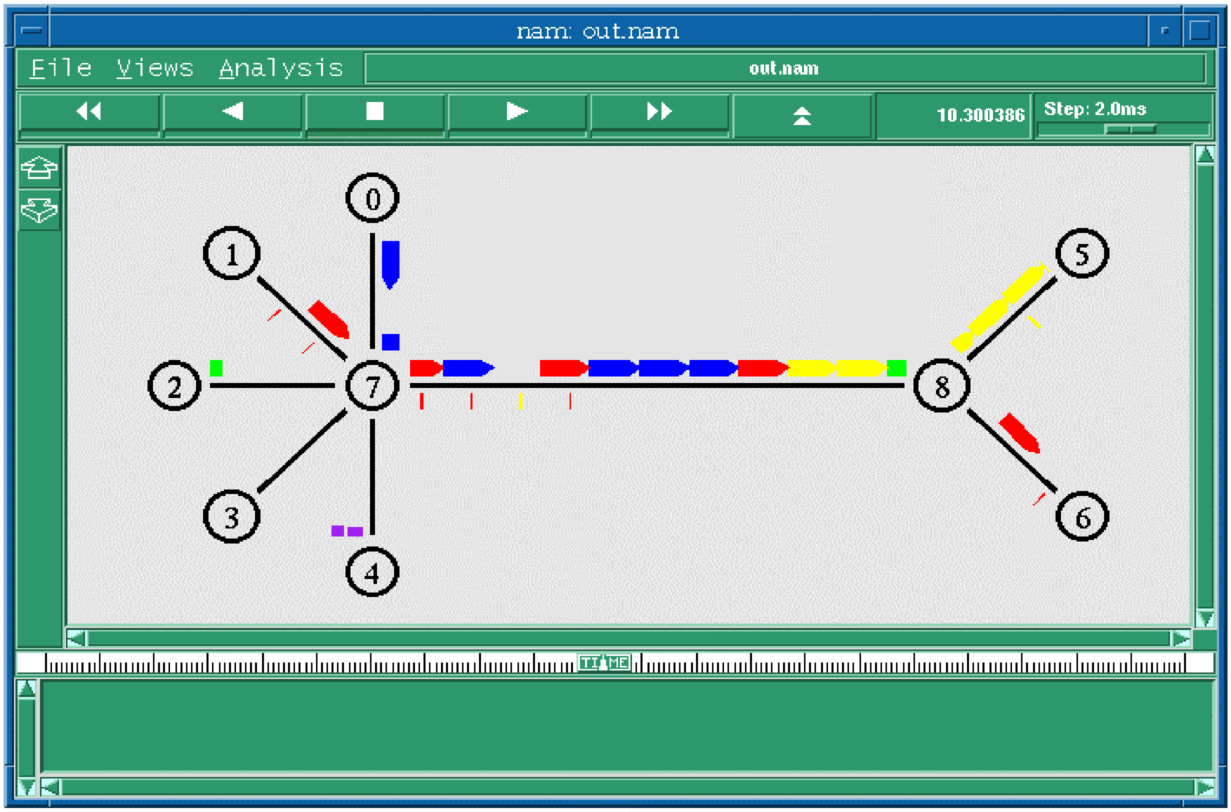


Figure 5.1.1.3: Topology 3

The current state of the Internet is such that multimedia packets are delivered using UDP in a TCP dominated environment. Table 5.1 shows the TCP dominant version of Topology 3, which consists of four TCP senders and one UDP sender.

Node	Type of Node	Bandwith 1	Bandwith 2	Queue Size 1	Queue Size 2	Start Time	End Time
				TCP Dominant			
0	TCP	8 Mb	16 Mb	5	10	0 sec	20 sec
1	TCP	8 Mb	16 Mb	5	10	2 sec	20 sec
2	UDP	8 Mb	16 Mb	5	10	2 sec	20 sec
3	TCP	8 Mb	16 Mb	5	10	4 sec	20 sec
4	TCP	8 Mb	16 Mb	5	10	3 sec	20 sec
						Receives From Nodes:	
5	Receiver 1	8 Mb	16 Mb	5	10	0, 2, 3	
6	Receiver 2	8 Mb	16 Mb	5	10	1, 4	
7	Router 1	8 Mb	16 Mb	5	10		
8	Router 2	8 Mb	16 Mb	5	10		

Table 5.1: Topology 3 : TCP Dominant

Table 5.2 shows a different scenario for the same topology as depicted in Table 5.1. Table 5.2 explains Topology 3's parameters on a UDP dominant network. By looking at both these scenarios, a comparison of network congestion and performance may be made between TCP dominant and UDP dominant networks.

Node	Type of Node	Bandwith 1	Bandwith 2	Queue Size 1	Queue Size 2	Start Time	End Time
0	UDP	8 Mb	16 Mb	5	10	0 sec	20 sec
1	UDP	8 Mb	16 Mb	5	10	2 sec	20 sec
2	TCP	8 Mb	16 Mb	5	10	2 sec	20 sec
3	UDP	8 Mb	16 Mb	5	10	4 sec	20 sec
4	UDP	8 Mb	16 Mb	5	10	3 sec	20 sec
						Receives From Nodes:	
5	Receiver 1	8 Mb	16 Mb	5	10	0, 2, 3	
6	Receiver 2	8 Mb	16 Mb	5	10	1, 4	
7	Router 1	8 Mb	16 Mb	5	10		
8	Router 2	8 Mb	16 Mb	5	10		

UDP Dominant

Table 5.2: Topology 3 : UDP Dominant

5.1.2 Parameters Used in NS

A multimedia stream sent across a network was simulated using Jae Chung's extended protocol of NS [CC00]. Initially, bandwidth was 8 Mb, queue size was 5, and frames per second was 30. With these three parameters and a TCP dominant network, the network load was 2.96 Mb per second. The equation in obtaining the value for the network load is as follows:

30 fps * (Weighted average of multimedia frame sizes + 4 * TCP senders) * 8 bits= 2.96 Mb per second.

1 UDP, multimedia stream:

11000, 8000 and 2000 are the sizes of I frames, P frames and B frames respectively. There are 9 frames in each Group of Pictures (GOP), IBBPBBPBB. Thus the weighted average of the multimedia frame sizes are [1 (11000) + 2(8000) + 6(2000)]/9.

30 fps * 4333 1/3 bytes * 8 bits = 1.04 Mb per second

4 TCP streams:

The packet size of data sent across the TCP stream was set to 2000 bytes.

30 fps * 4 * 2000 bytes * 8 bits = 1.92 Mb per second

The calculated TCP packet has maximum load of 1.96 Mb per second, however this value will depend on the congestion of the network.

With a UDP dominant network, the network load is 4.9 Mb per second. Later bandwidth was increased to 16 Mb and queue size was increased to 10. Five different mpeg_trace.input files were generated to reflect the different types of redundancy schemes to be analyzed. The five redundancy arrangements were I frame, P frame, B frame, All frames, and no frames. The group of pictures (GOP) pattern used was: IBBPBBPBB.

5.1.2.1 I Frame Redundancy

The packet size of the primary I frame was 11000 bytes. A lower quality I frame is piggybacked on the proceeding B frame in the GOP used. The lower quality I frame added 1568 bytes or 14.25% of the original higher quality I frame to the proceeding B frame. This is depicted in Figure 5.1.2.1. Percentages used to calculate the lower quality frames were obtained from Yanlin Liu's thesis [LC99].

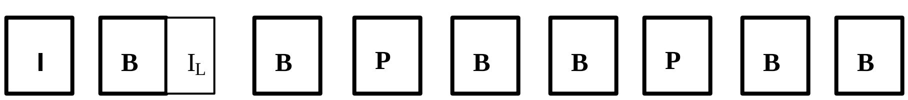


Figure 5.1.2.1: I Frame Redundancy

5.1.2.2 P Frame Redundancy

The primary P frames were each 8000 bytes. Using P frame redundancy, a low quality P frame was piggybacked onto the proceeding B frame. The lower quality P frame was 1440 bytes or 1.8% of the primary P frame. This is shown in Figure 5.1.2.2.

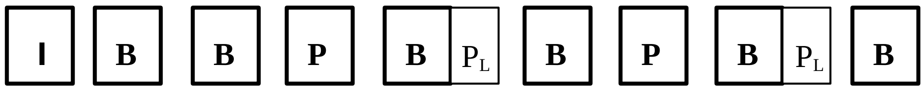


Figure 5.1.2.2: P Frame Redundancy

5.1.2.3 B Frame Redundancy

Primary B frames were 2000 bytes. Similarly, we piggybacked a lower quality B frame onto the proceeding B or P frame in the GOP. This is shown in Figure 5.1.2.3. The lower quality B frame is 256 bytes or 12.8% of its primary B frame counterpart.

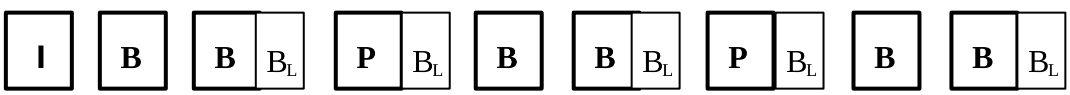


Figure 5.1.2.3: B Frame Redundancy

5.1.2.4 All Frame Redundancy

The redundancy scheme that piggybacks a low quality frame for every frame is depicted in Figure 5.1.2.4. A low quality version of I, P and B frames was piggybacked to the following frames.

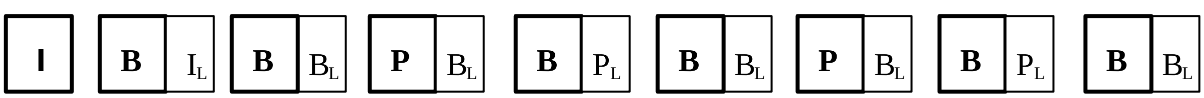


Figure 5.1.2.4: All Frame Redundancy

5.1.2.5 No Redundancy Frame

The no redundancy frame scheme does not piggyback any low quality frames. In hopes to make improvements to network congestion, the original scheme with no redundancy was compared to the other four redundancy schemes in the analysis section of this report.

For each of the five redundancy schemes, a GOP pattern was repeated (with the redundancy frames factored) to generate a sequence of 1000 frames. Each of the five patterns were written to five separate mpeg_trace.input files and used as input into NS.

Each of the five redundancy schemes was ran with NS to produce five separate simulation files. Next, a Perl script was used to parse each of the files to calculate the number of drops and amount of throughput for each sender. The results returned by the Perl script was compared to the results from a program called `grepfile` (written by Jae Chung) that had been already tested to ensure accuracy. The Perl script's correctness was further tested by comparing its results to the observations made during the NS simulation. The following is the pseudo code of the Perl Script in gathering the packet drops and throughput of each sender.

Program Name: *Automate.pl*

Description: *This program parses the data file generated by NS and calculates the number of dropped and sent packets between various nodes. The results are printed out to screen.*

How To Run Program: *At the prompt, type:*

> perl automate.pl <parameter file> <Final statistics filename>

Pseudocode:

1. The program begins by opening a parameter file supplied by the user containing the following information:

- name of the NS input file being read (1st line)
- name of the output file to which the intermediate data is written and stored (2nd line)
- number of paths that require analysis (3rd line)
- The start and end node numbers (4th through Nth line)

2. Open the input in read mode and output files in write mode and assign them file handles.
3. Read through each line of the input file (generated by NS[1]) and search for frames sent between a specified time period (i.e., between 5 and 15 seconds).
4. For every line from the input file that meets the criteria specified above, copy the line to the intermediate output file.
5. Open the intermediate data file for reading.
6. Check the each line of the intermediate data file for combination type:
    - Received or dropped
    - UDP or TCP

Four variables - $sum_of_dropped_udp, $sum_of_received_udp, $sum_of_dropped_tcp, and $sum_of_received_tcp are used to track this data.

[1] Note that the input file generated by NS (i.e., 'output.tr') is in the format shown below:

```
r  5.0002   7 8 tcp 2000 ------- 2 1.0 6.0 350 2643
+ 5.0002   8 6 tcp 2000 ------- 2 1.0 6.0 350 2643
-  5.0002   8 6 tcp 2000 ------- 2 1.0 6.0 350 2643
```

7. Open the file to which the final statistics are written. This file will contain the statistics in the proceeding format:

   TCP Connections

   Start Node: 0, End Node: 7, R: 973
   Start Node: 0, End Node: 7, D: 3

   UDP Connections

   Start Node: 0, End Node: 7, R:
   Start Node: 0, End Node: 7, D:

The above example indicates that from node 0 to node 7, nine-hundred and seventy-three data packets were received and three data packets were dropped using TCP. However, the number of UDP data packets received or dropped is zero. Excel was then used to graph the data that we collected from running the NS simulation.

5.2 Analysis of NS

The data generated by NS was filtered in such a manner as to reduce the measurement of activity associated with the start and end of sending data. NS was run for twenty seconds, but only the data generated during the five to fifteen second range was recorded and used. We wanted to eliminate any sporadic conditions caused by startup or ending conditions in the network simulation. This was accomplished using the `automate.pl`, the PERL script mentioned previously. Once the data files containing the measurements are generated by the PERL script, the information was entered into a Microsoft Excel file. The data contained within the data files were then used to create the various graphs that are described in the next section.

---

r 5.00045 7 1 ack 40 ------- 2 6.0 1.0 346 264

5.2.1 TCP Dominant : TCP Transmission

The measures in the Figure 5.2.1 describe the percentage of packets dropped based on altering two parameters – bandwidth and queue size. Results show that packet drops were most frequent when using a bandwidth of 8 and queue size of 5. Second in terms of packet drops were obtained by using a bandwidth of 8 and queue size of 10. Next came bandwidth 16 and queue size 5. The least number of packet drops was achieved by using a bandwidth of 16 and queue size of 10. There was no clear correlation between the type of redundancy used and the resulting number of packet drops.

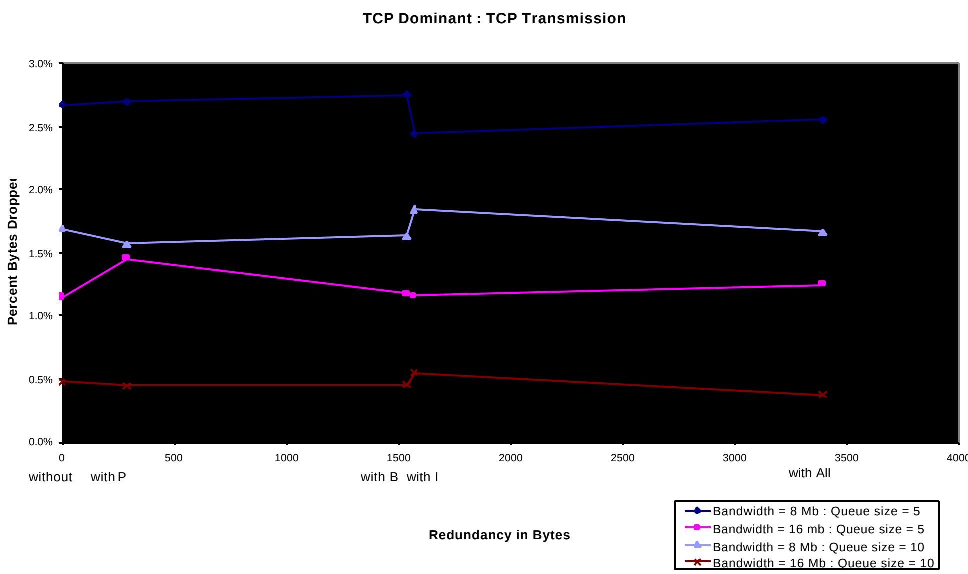


Figure 5.2.1: TCP Dominant : TCP Transmission

5.2.2 TCP Dominant : UDP Transmission

This graph (Figure 5.2.2) indicates that the least number of packet drops occurred when using a bandwidth of 16 and queue size of 5. Behavior of the other cases used displayed no pattern. Using different types of redundancy resulted in different percentage packet loss for the various cases.

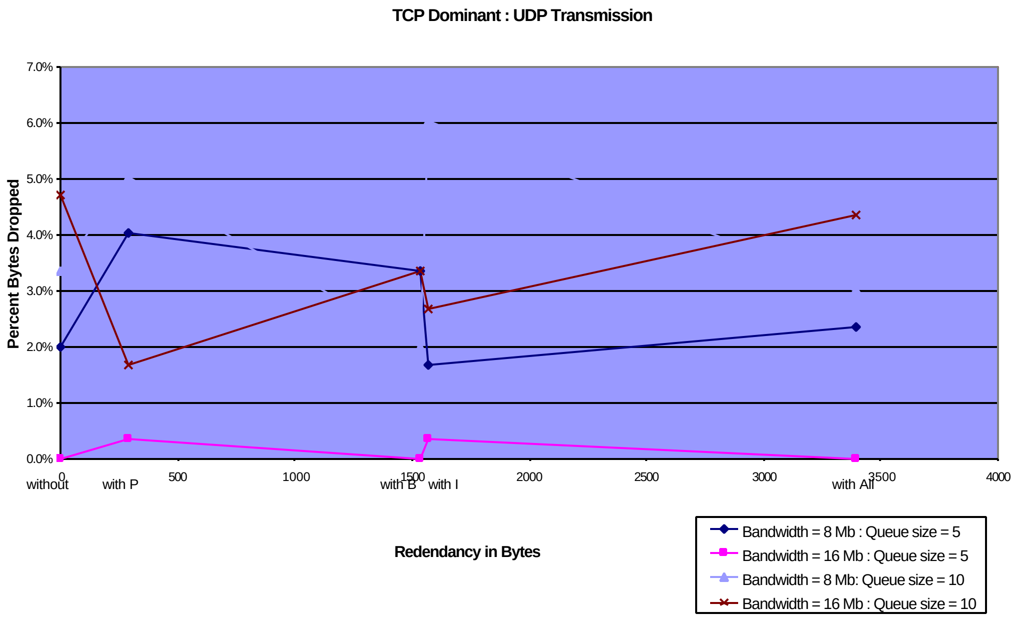


Figure 5.2.2: TCP Dominant : UDP Transmission

5.2.3 UDP Dominant : TCP Transmission

Shown in Figure 5.2.3, the most packet drops occurred when using a bandwidth of 8Mb and queue size of 5. Using a bandwidth of 16Mb and queue size of 5 resulted in the second most frequent number of packet drops. Next came bandwidth 8Mb and queue size 10. The least number of packet drops was achieved when using a bandwidth of 16 and queue size of 10. A pattern can be noticed when using bandwidth 8 and queue size 5. As redundancy in bytes is increased, percentage of bytes dropped also increases. There is no discernable pattern with the other three cases.

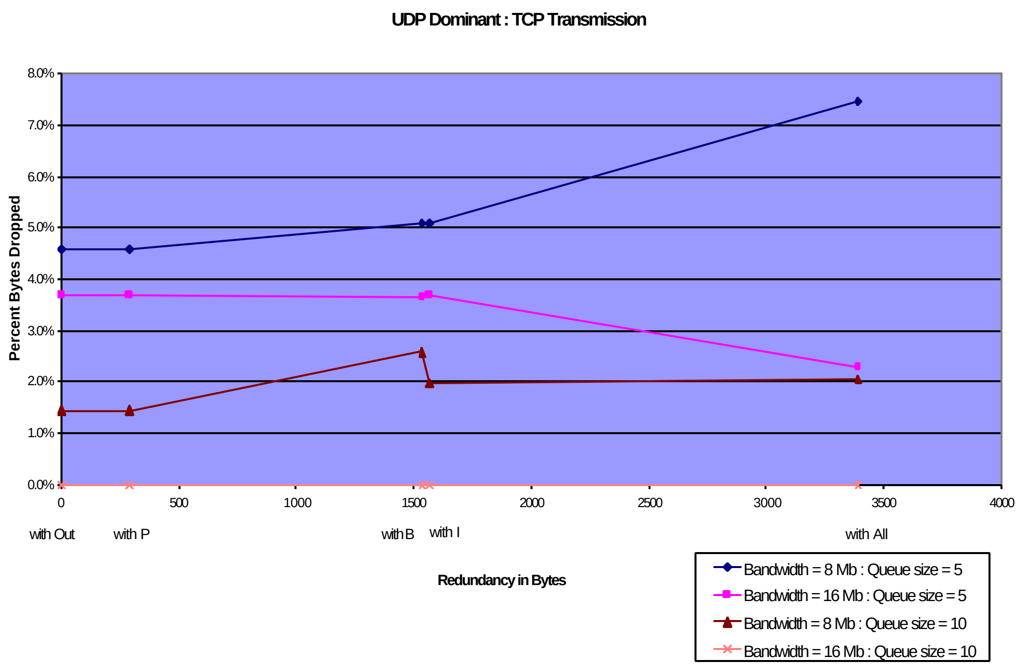


Figure 5.2.3: UDP Dominant : TCP Transmission

5.2.4 UDP Dominant : UDP Transmission

In two particular cases, the most number of packet drops were obtained using redundancy of B packets while the least number of drops occurred when using redundancy of all packets (refer to Figure 5.2.4). The percentage drops resulting from using redundancy of B packets resulted in a much higher loss rate than that of using no redundancy at all. The findings suggest that the best performance should result when using redundancy of all packets. This observation occurred under the conditions that:

a. bandwidth = 8Mb; queue size = 5
b. bandwidth = 8Mb; queue size = 10

In both these cases, percentage packet drops were identical for using redundancy of P frames and using no redundancy at all. For the case where bandwidth was 16Mb and queue size was 10, percentage drop was zero for all amounts of redundancy.

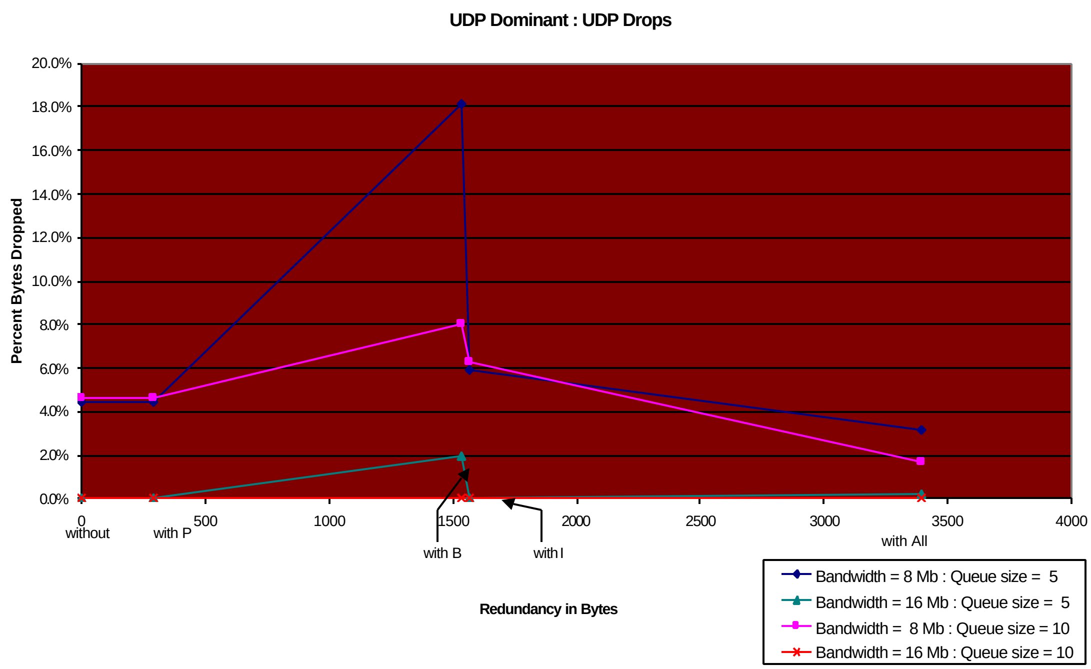


Figure 5.2.4: UDP Dominant : UDP Transmission

5.2.5 Average Percent Drops

The data in Figure 5.2.5, suggests that in a UDP dominated environment, a scheme using all packet redundancy would achieve the least number of drops. In a TCP dominated environment, B packet redundancy should be used. Packet drops for a TCP transmission in a TCP dominated environment did not vary significantly; packet drop levels remained relatively steady. Packet drops for a TCP transmission in a UDP dominated environment fluctuated slightly. In this specific case, packet loss was highest when using all packet redundancy followed by B packet redundancy.

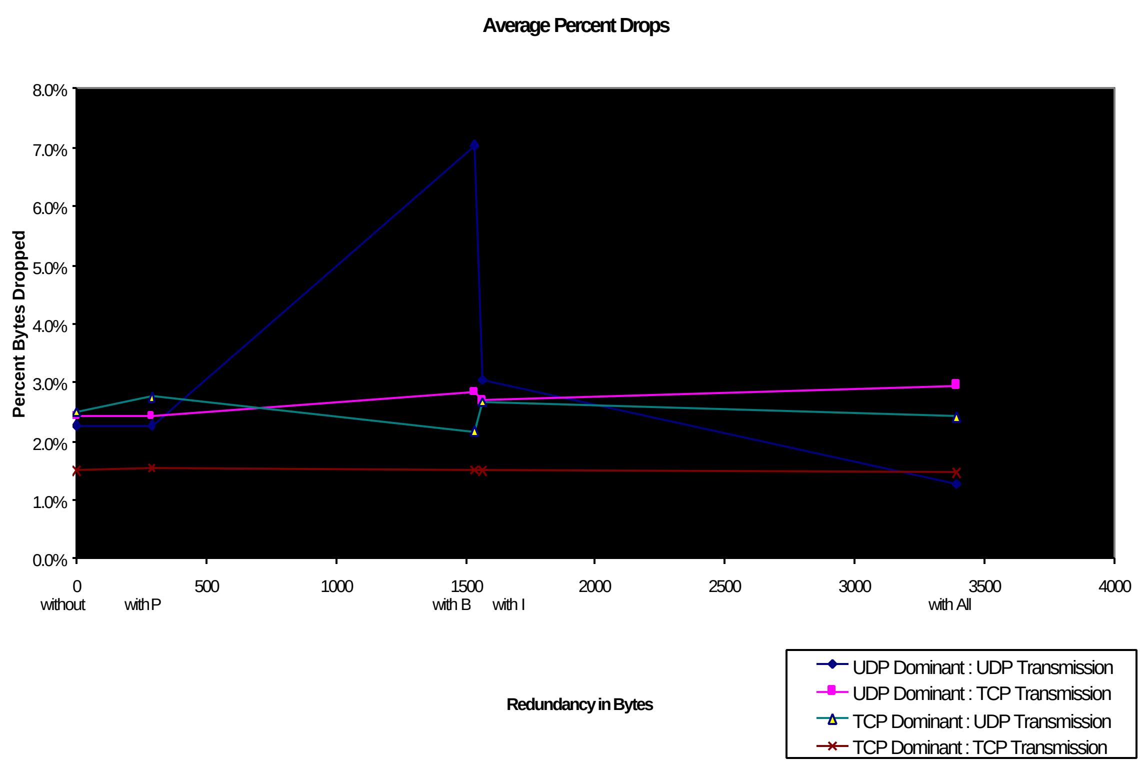


Figure 5.2.5: Average Percent Drops

Chapter 6: User Study

Another important facet to this paper is the user quality study. By using the redundancy techniques described in the NS section of this paper, we built a user interface to test user perception of the video quality. From previous examples of user testing interfaces [LC99][TC99], we used Microsoft Media Player 6.0 to play our video streams. Visual Basic was used as the front end for users to view video clips and rate video quality.

We measured perceptual quality ratings that users assigned to movies of varying redundancy schemes and amounts of loss. Obtaining and analyzing these parameters provides useful information in guiding further research in the multimedia community. Each frame also has many parameters that can be varied such as frame rate, frame size, color (i.e., color or black and white), content of the frame, spacing of frames (i.e., when and where placement of I frames), GOP and Quality of the frames (i.e., high or low). Out of these variables we analyzed the amount of data loss and the quality of the frames.

6.1 Parameters of Video Clips used in the User Study

Different parameters were used to build the movies in order to test the user's reaction to the various redundancy techniques being analyzed in this paper. Quality of the primary frames and secondary frames were selected with the size of different types of quality levels in mind. That is, a quality level of 5 was chosen over a quality level of 1 because the size was significantly smaller without a drastic difference in perceptual quality. From previous research, video clips of quality 1 are exponentially larger than video clips of quality 5 [LC99]. For our user study, the primary video frames have a quality of 5 and secondary video frames have a quality of 25. Since there was previous work that was accomplished with primary video clips that have a quality of 1 [LC99], we wanted to do a perceptual quality comparison on primary video frames with quality of 1 and quality of 5.

This chapter analyzes the perceptual quality rated by users for the different percentage of packet drops in a movie clip. The percentages that were examined are 0, 3, 5, 10 and 15. Usually Internet packet loss percentage is less than 5 percent. However, we wanted to look at two extreme loss cases, including loss percentages of 10 and 15 in this study. In this user experiment, movie clips had the five different amounts of drops.

Since the study is on the user's perceptual quality ratings of the five redundancy schemes, we did not want consecutive packets being dropped; doing so would produce a confounded factor in our analysis. Movies with different redundancies of I frames, P frames, B frames and all frames were built in combination with different loss percentages. The redundancy scheme is described in section 5.1.2 of this paper.

6.2 Building the Movie Clips

Using an MPEG player on a UNIX platform, original MPEG files were broken up into separate ppm (Portable Pixmap File format) files. These movies were of various types including:

- Sports
- Sitcom
- Cartoons
- News Broadcast

- Home Shopping

Two sets of ppm files were created and stored into two separate directories. The first set consisted of high quality frames (i.e., quality level 5) while the second set consisted of low quality frames (i.e., quality level 25). The underlying idea was to simulate a situation where dropped packets were being replaced with low quality redundant frames to counteract the effects of loss. In other cases, a frame would be repeated numerous times.

In order to accomplish this task a Perl script (see pseudocode below) was used to either replace various high quality frames with low quality frames or repeat certain high quality frames. The low quality frames represented the redundant frames that replaced the lost high quality frames. When a redundant frame is not available to replace lost high quality frames, the last successfully received high quality frame is repeated. The following is pseudocode describing the `Replacer.pl` program that performed the function described above:

Program Name: Replacer.pl

Description: This program allows a user to build movies using various quality levels and redundancy schemes. The program uses two sets of movie frames to simulate the implementation of Forward Error Correction. The user is prompted to enter the level of loss and the loss is simulated through the program dropping random high quality frames. Dropped high quality frames are either replaced by a lower quality version of itself or the last successfully received frame. The course of action taken is dependent upon the frame type in question and the type of redundancy used.

How To Run Program: *At the prompt, type:*

\> *perl Replacer.pl*

Pseudocode:

1. Prompt user to enter:
   - Number of movie frames (i.e., PPM files)
   - Percentage rate loss
   - MPEG filename
2. Calculate the number of frames that need to be dropped (i.e., N drops)
3. Initialize low quality array and chosen array. Low quality array keeps track of frames that have degraded quality because of lost frames. Chosen array keeps track of which frames have been randomly dropped.
4. Loop N times and continue to generate a random frame to drop until the frame chosen will not result in consecutive loss. Also ensure that the frame chosen has not already been dropped; if so, generate new random frame number to drop.
5. Record the frame number dropped and determine the frame type of the dropped frame (i.e., I, B, P1 or P2). P1 is the first P-frame in the GOP sequence and P2 is the second P-frame in the sequence.
6. If Frame type == I Then

   If using *I Redundancy* or *All Redundancy* Then

      Make the I-frame and its dependant frames low[2] quality

      Increment the number of replaced frames

   End If

[2] PPM stands for Portable Pixmap File Format

If using *B Redundancy* or *P Redundancy* or *No Redundancy* Then

Repeat the last frame received multiple times

Increment the number of repeated frames

End If

End If

If Frame type == P1 then

If using *P Redundancy* or *All Redundancy* Then

Make the P1-frame and its dependent frames low quality

Increment the number of replaced frames

End if

If using *I Redundancy*, *B Redundancy*, or *No Redundancy* Then

Repeat the last frame received multiple times

Increment the number of repeated frames

End If

End If

If Frame type == P2 then

If using *P Redundancy* or *All Redundancy* Then

Make the P1-frame and its dependent frames low quality

Increment the number of replaced frames

End if

If using *I Redundancy*, *B Redundancy*, or *No Redundancy* Then

Repeat the last frame received multiple times

Increment the number of repeated frames

End If

End If

If Frame type == B then

If using *B Redundancy* or *All Redundancy* Then

Make the B frame low quality

Increment the number of replaced frames

End if

If using *I Redundancy*, *B Redundancy*, or *No Redundancy* Then

Repeat the B frame with the previous frame

Increment the number of repeated frames

End If

End If

Calculate the percentage of frames replaced and repeated

Print out statistics:

- Total repeated frames
- Total replaced frames
- Percentage repeated frames
- Percentage replaced frames

Once these operations have been performed on the set of high quality frames, an MPEG encoder is used to reassemble the new MPEG file from the set of modified ppm files. The encoder reads in a parameter file as input in the process of rebuilding the MPEG files. Within each parameter file are various specifications including:

- Frame pattern; in our tests, it was IBBPBBPBB
- I, P, and B quality levels (in our test, the quality level ranged from 1-30)

6.3 User Interface Design

The user interface was designed on Dual Pentium Pro computers with Windows NT 4.0 and Microsoft Office 2000 installed on them. They have two 200 MHz CPUs. Microsoft Visual Basic 6.0 was used to design the user display and it is connected with Microsoft Access to store the user personal information and movie ratings. Media Player 6.0 was used to display the MPEG movie clips. Twenty-seven unique MPEG clips were selected for the study, each movie clip lasting 16 seconds. Movies with combinations of the following two sets of parameters were chosen:

- Loss Percentage of 0, 3, 5, 10 and 15
- Redundancy Scheme with I frame, P frame, B frame, all frames and no redundancy

The movie sequence and its parameters are shown in Table 6.4, which is in the User Data Analysis section of this paper. The experiment was carried out on 10 Dual Pentium Pro computers with Windows NT platform and Office 2000 installed on them.

The first screen of the user interface is shown in Figure 6.3. This screen requests the user's personal information including their name, e-mail address, age, major and computer familiarity (refer to the Appendix for the user demographic data information). The requested information is then stored in a Microsoft Access database. Users were given the option of inputting their real user information or a fictitious user name, e-mail address and age in order to keep their privacy safe. Access was designed to assign unique keys to each individual user to keep user information distinguishable.

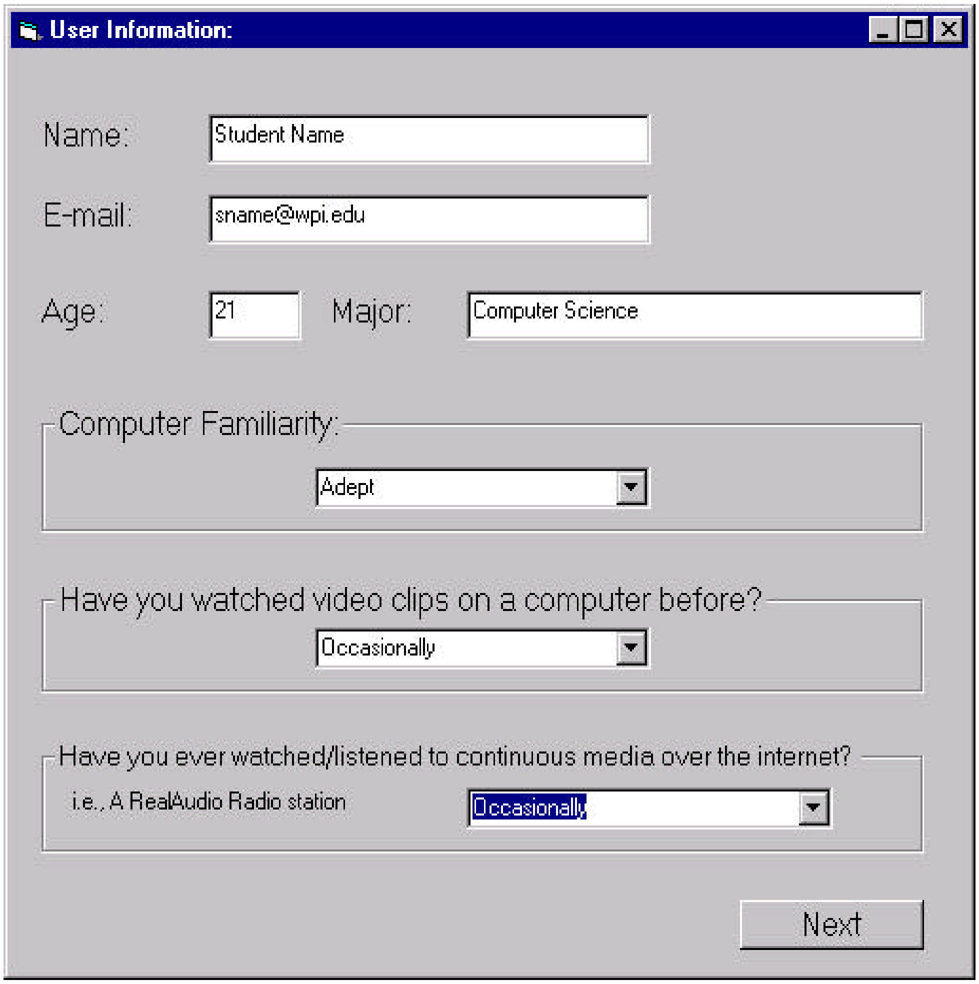


Figure 6.3: First User Interface Screen

There are three pull down selection boxes for the questions on how familiar the user is to computers, viewing video clips on the computer and viewing or listening to continuous media over the Internet. For the question on user's computer familiarity, he or she has three choices to choose from. They can select on Novice, Adept or Wizard. For the questions, "Have you ever user watched video clips on a computer before?" and

"Have you ever watched/listened to continuous media over the Internet?" he or she has these following three choices: Never, Occasionally and Frequently. Limiting the answers to these three choices to the question allows the data to be more standardized.

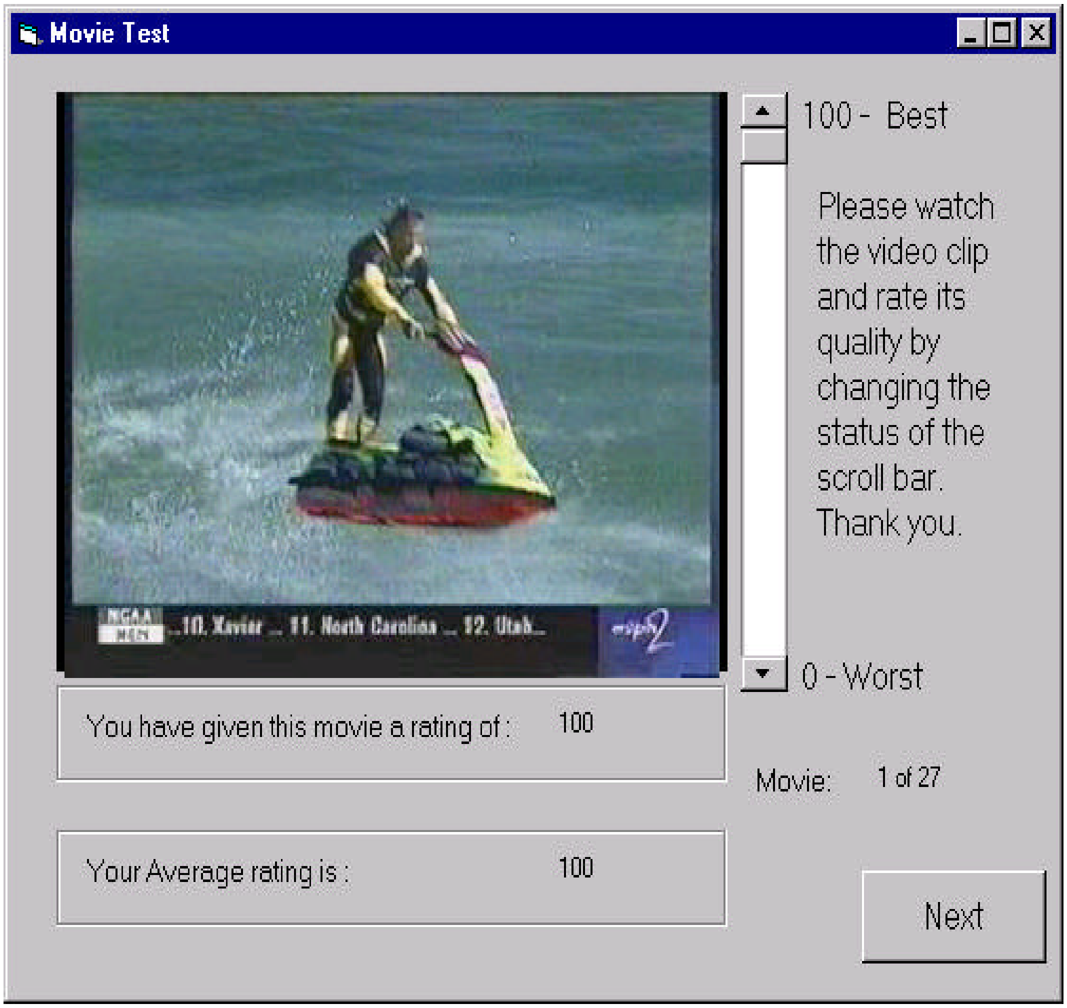


Figure 6.3.2: Second User Interface Screen

The second Visual Basic screen is used to display the 27 MPEG video clips. As shown in Figure 6.3.2, Media Player is embedded inside the VB interface with a scroll bar for users to rate the movies to the right of the movie window. At the top of the scroll bar is a rating of 100, designating the highest quality rating that a video clip can possibly receive. At the bottom of the scroll bar is 0, the lowest quality rating that a video can possibly receive. For every video clip that is shown to the user, the position of the scroll bar is set to 100 before the user moves the position of the scroll bar. After the user finishes watching the MPEG video clip and gives it a rating, the user will click on the 'Next' button and a new video will be displayed in the movie window.

Figure 6.3.3: User Interface displaying the 11th Movie Clip

At the bottom of the second Visual Basic user interface, the user's rating is displayed. There is also a display of the average user rating for the previously viewed MPEG video clips. This display gives the user information on how he or she had rated the previous video clips for comparison, as shown in Figure 6.3.3. Toward the lower right hand corner of the user interface is a display of the video clip sequence number that the user is currently viewing (out of the 27 total video clips). This display gives the user an idea on how many more video clips remain. When the user finishes rating the last video clip, the ‘Next’ button becomes an ‘Exit’ button for the user to exit the user interface.

All the user data and ratings are saved to a Microsoft Access database, which is located on the C drive of each of the 10 computers that were used. Ten separate databases were used to solve the concurrency problem where 10 users may be trying to access the same database at the same time. Later, the 10 databases were merged into one database for retrieving data and analysis.

6.4 User Data Analysis

The data gathered from the 46 users were both graphed and analyzed. The demographics of these 46 users' experience with computers, multimedia on computers and multimedia experience on the Internet are shown in Figures 6.4, 6.4.0.2 and 6.4.0.3. From these graphs, the majority of users selected ‘Adept’ for their experience with personal computers and ‘Occasionally’ for their multimedia experience both on PCs and over the Internet. Fewer users felt they were either novices or wizards in their experience with personal computers. Fewer users also felt they “Never” or “Frequently” used multimedia on PCs and over the Internet. The range of ages of these users is from 17 to 40 years of age. Some of the majors that were in the user profile are Computer Science, Electric Engineering, Mathematics, Biology, and undecided.

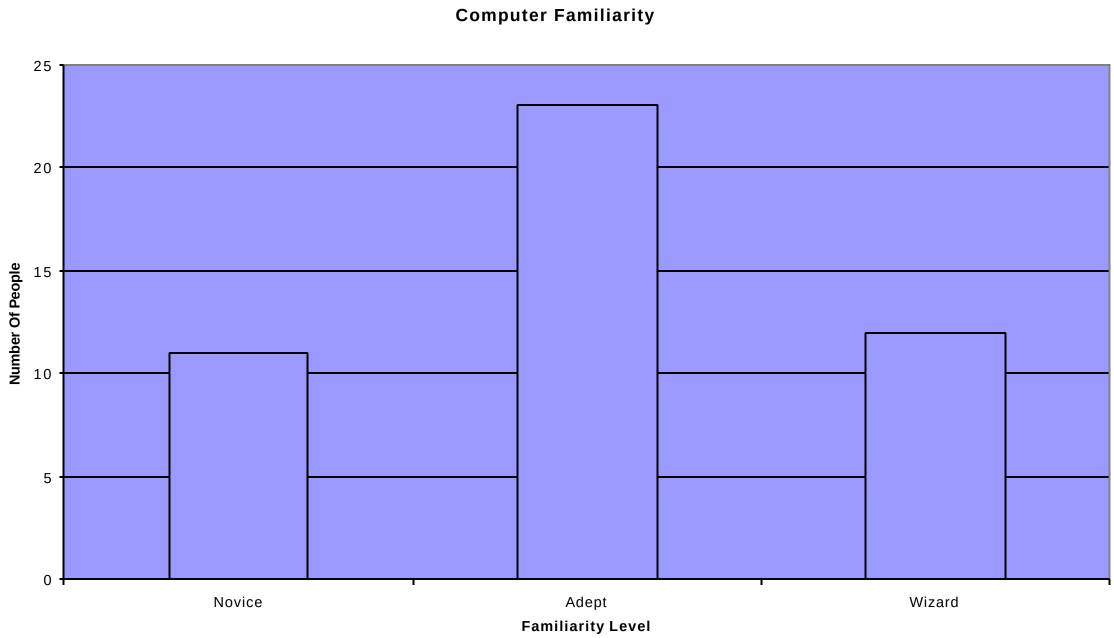


Figure 6.4: Computer Familiarity

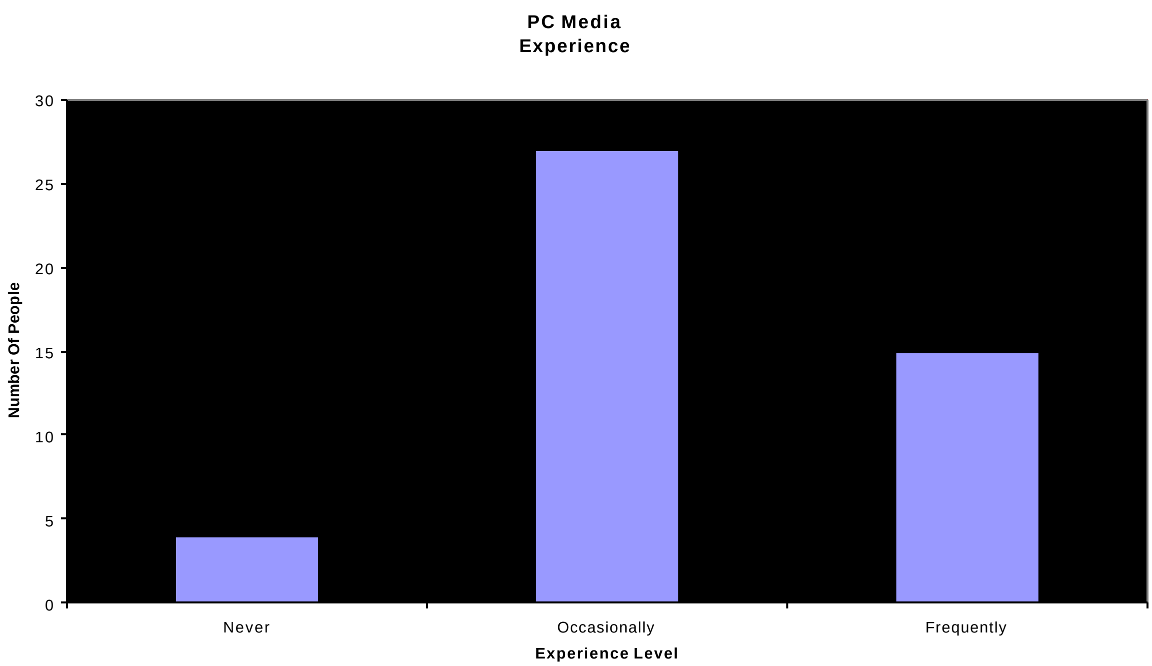


Figure 6.4.0.2: Computer Multimedia Experience

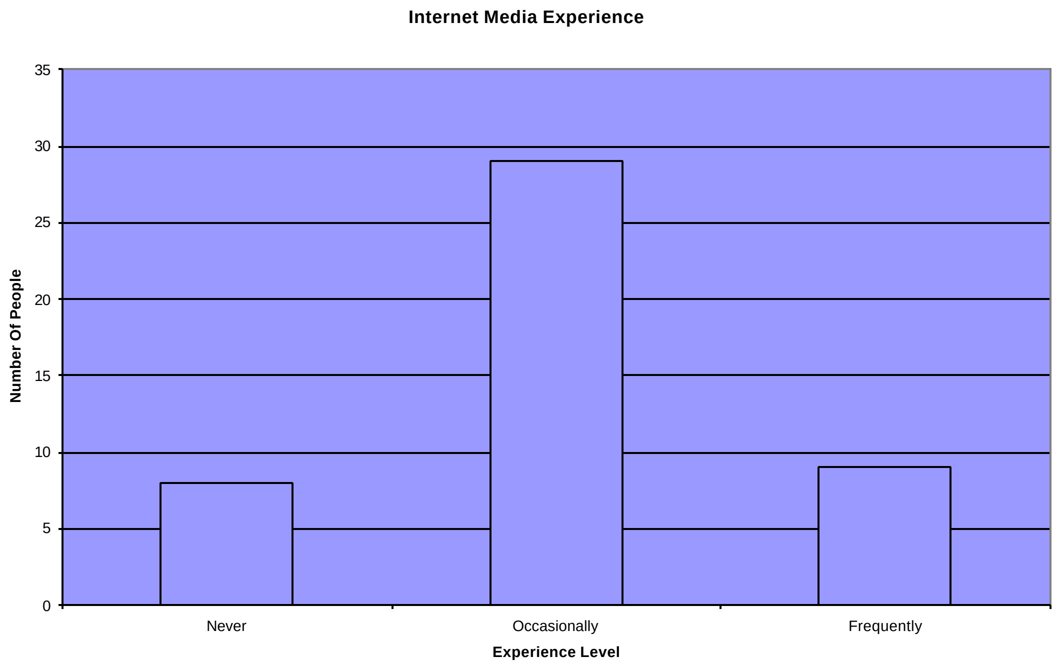


Figure 6.4.0.3: Internet Multimedia Experience

The order of the randomized movies is shown in Table 6.4.1.1 with different MPEG clip parameters. The average perceptual quality ratings of the users were derived from the Access database. The value 100 is taken out before the averages of the movies were calculated. These ratings of the different video clip are listed on the last column of following table. Table 6.4.1.2 displays the same list of parameters and information according to the different percentage loss and redundancy scheme that was applied to each video clip.

Movie ID	Redundancy / Quality	Percent Loss	Type of Movie	Average - no 100
1	Quality 5	0	Sports1	82.00
2	WithOut	5	Cartoon1	63.50
3	WithAll	15	Sports2	55.83
4	WithB	10	SitCom1	66.67
5	WithP	3	News1	70.07
6	WithI	15	SitCom2	51.39
7	Quality 5	0	News2	79.59
8	Quality 1	0	Cartoon2	79.69
9	WithAll	5	News3	70.72
10	WithP	10	Sports3	65.84
11	WithB	3	Cartoon3	78.50
12	WithI	10	News4	72.33
13	Quality 25	0	Shopping1	53.65
14	WithOut	3	SitCom3	65.45
15	WithI	5	Sports4	75.27
16	WithP	15	Cartoon4	62.11
17	Quality1	0	News5	89.21
18	WithAll	3	News6	72.72
19	WithOut	15	Shopping2	66.73
20	WithB	15	News7	70.26
21	Quality 5	0	Cartoon5	77.76
22	WithI	3	News8	70.93
23	WithB	5	Sports5	82.73
24	Quality 25	0	News9	60.67
25	WithAll	10	Cartoon6	63.11
26	WithP	5	SitCom4	62.11
27	WithOut	10	News10	68.80

Table 6.4.1.1: Movie Parameters and Perceptual Quality Ratings

Percent Loss	Movie ID	**WithOut**	Average - no 100	Movie ID	**With P**	Average - no 100
0%						
3%	14	SitCom	65.45	5	News	70.07
5%	2	Cartoon	63.50	26	SitCom	62.11
10%	27	News	68.80	10	Sports	65.84
15%	19	Shopping	66.73	16	Cartoon	62.11

Percent Loss	Movie ID	**With B**	Average - no 100	Movie ID	**With I**	Average - no 100	Movie ID	**With All**	Average - no 100
0%									
3%	11	Cartoon	78.50	22	News	71.57	18	News	72.72
5%	23	Sports	82.73	15	Sports	76.35	9	News	70.72
10%	4	SitCom	66.67	12	News	72.93	25	Cartoon	63.91
15%	20	News	70.26	6	Sitcom	51.39	3	Sports	55.83

Table 6.4.1.2: Redundancy Type and Perceptual Quality Rating

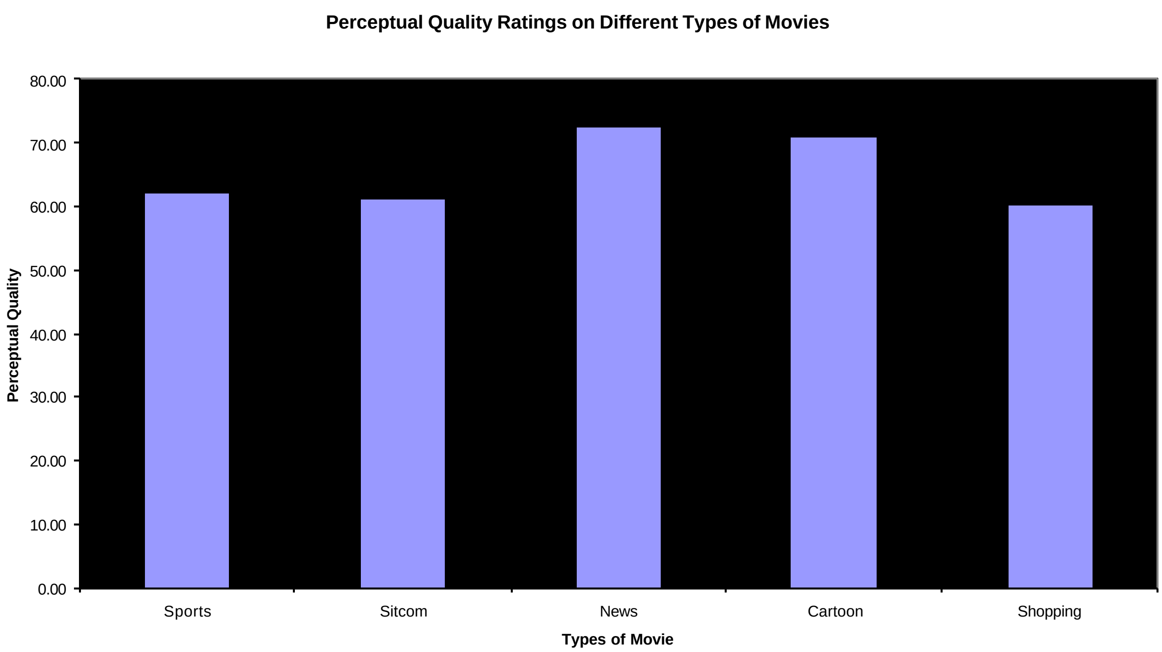


Figure 6.4.0.4: Perceptual Quality Ratings on Different Types of Movies

Figure 6.4.0.4 displays the different perceptual quality ratings of different types of movies. The movie categories include: Sports, SitCom, News, Cartoon and Shopping. Sport clips included water skiing, soccer games and hockey games. Sitcom video clips included “Married with Children” and “Third Rock from the Sun.” News clips featured segments of CNN and other news stations. Cartoon video clips included scenes from the cartoon series, “The Simpsons.” Video clips under the shopping category were taken from segments of the “Home Shopping Network.” The contents of different video clips do make a difference on perceptual quality ratings. How contents of video clips affect perceptual quality can be explored further in future work.

6.4.1 Stationarity of User Data

A stationary process is a data-gathering mechanism for which the pattern of variation does not change as more data are taken [PNC99]. Stationarity of the experiment is very important because it needs to be established before any more analysis of the data can be performed. The line plot (Figure 6.4.1) of the movie sequence versus perceptual quality was graphed and it shows stationarity. There is no increase or decrease as time progressed meaning the movie sequence and data collected was random enough to be further analyzed. The R^2 value was calculated to be 0.0015, which means there is no correlation between the x values and the corresponding y values. No correlation in this graph gives us confidence that the order in which the movies are presented does not influence the user ratings.

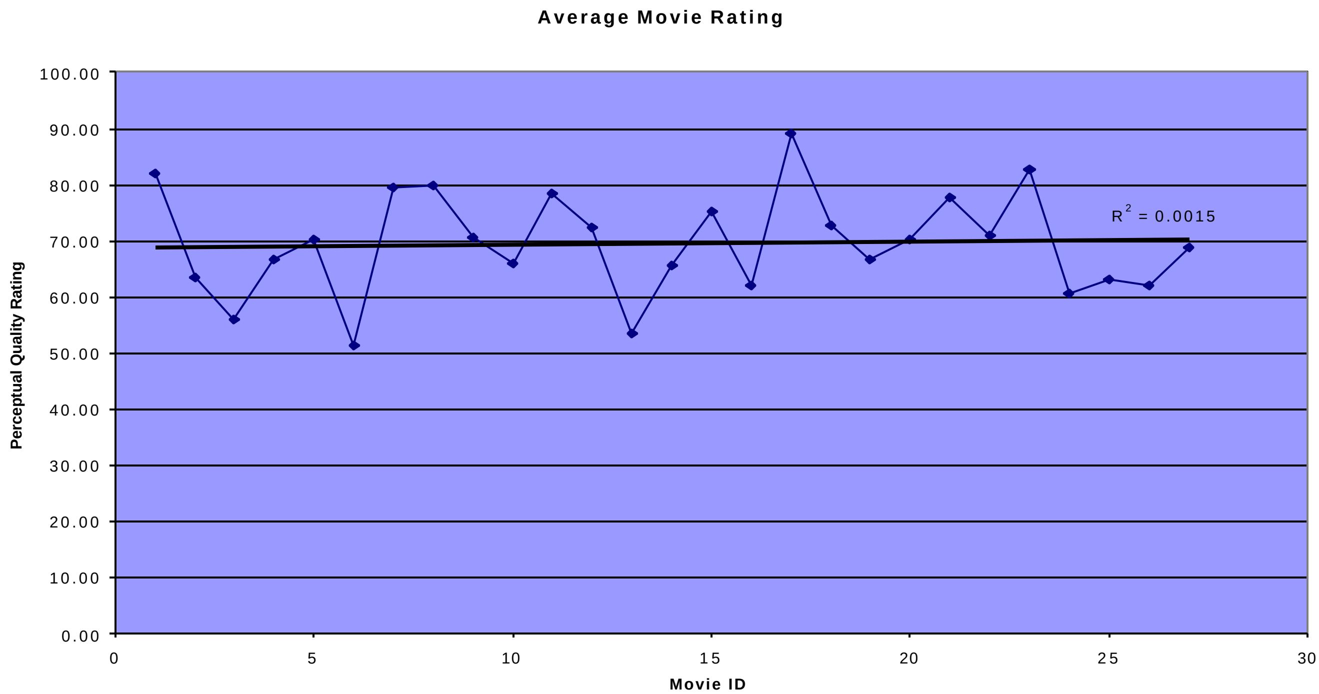


Figure 6.4.1: Graph showing Stationarity

6.4.2 Added Redundancy Bytes vs Perceptual Quality

According to Figure 6.4.2, B-frame redundancy appears to have performed the best in terms of preserving the perceptual quality of the MPEG's.

This result occurred when testing three, five, and fifteen percent data loss rates. As stated previously, this result may be due to the fact that B-frames appear so frequently in a group of pictures. Attaching a redundant frame of the frames that are most likely to be lost may result in the most optimal solution of repairing data lost. The only exception occurred when using a ten percent data lost rate; in that particular case, using I-frame redundancy performed the best.

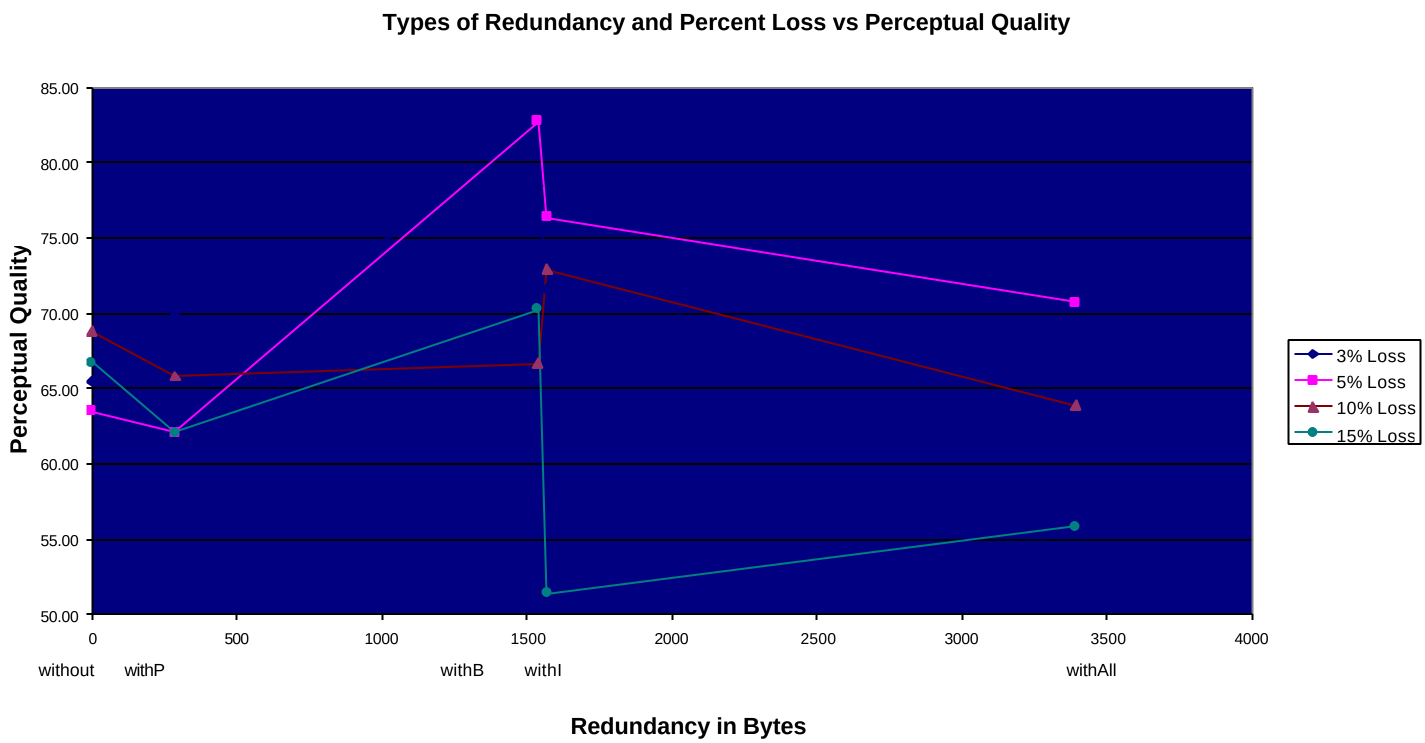


Figure 6.4.2: Added Redundancy in Bytes vs Perceptual Quality

6.4.3 Loss vs Perceptual Quality

As Figure 6.4.3 indicates, when data loss is high, perceptual quality degrades. The above graph includes the perceptual quality ratings based on a scale of one to one hundred that were attributed to various movies by users. In addition, the graph does not include user ratings that have scores of one hundred. Between one and three percent, there is a noticeably steep degradation in the MPEG quality. Quality degradation levels off between three and five percent loss, but continues to moderately degrade between a five and ten percent drop. For loss rates of ten percent or greater, degradation of quality becomes noticeably worst. The graph seems to indicate that users clearly see a difference between a perfect movie clip and a slightly flawed movie clip. When a movie clip has moderate loss (i.e., between three and ten percent loss), the quality level between movies having different lost rates becomes less noticeable.

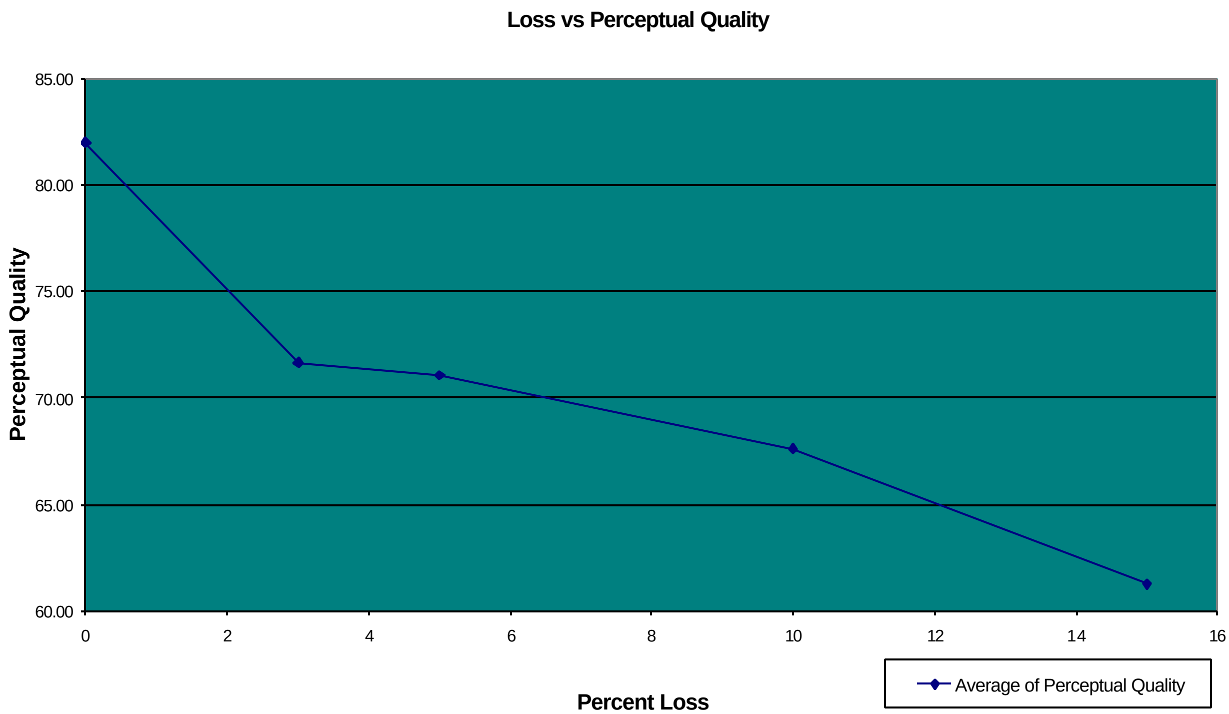


Figure 6.4.3: Actual Loss vs Perceptual Quality

6.4.4 Types of Redundancy, Percent Loss vs Perceptual Quality

Shown in Figure 6.4.4 is a slice of the user study result. The Y-axis is the perceptual quality ratings and the X-axis is the percentage loss. There is a noticeable downward trend for the lines in this graph, which displays that as the percentage loss of the video stream increases the perceptual quality ratings decrease. The typical congestion on a network will have a packet loss of less than 5 percent. In the portion of the graph spanning 0 to 5 percent loss, a positive correlation exists. The more redundant data added to repair lost data resulted in a higher the perceptual quality rating. The only exception was the B frame redundancy scheme.

This graph also shows data for network loss where there are packet drops larger than 5 percent. For the latter part of the graph, the lines show that the different redundancy schemes have flip-flopped from the beginning of the graph. This seems to show that with large amounts of drops in the data stream, users do not see the difference between one redundancy technique versus another. However, the B frame redundancy appears to have done better than the others for most amounts of loss. This may be due to the fact that a GOP is composed of 67 percent B frames. There is a greater probability for a B frame to be dropped than an I or a P frame. Piggybacking a lower quality B frame will help reduce the amount of B frame drops.

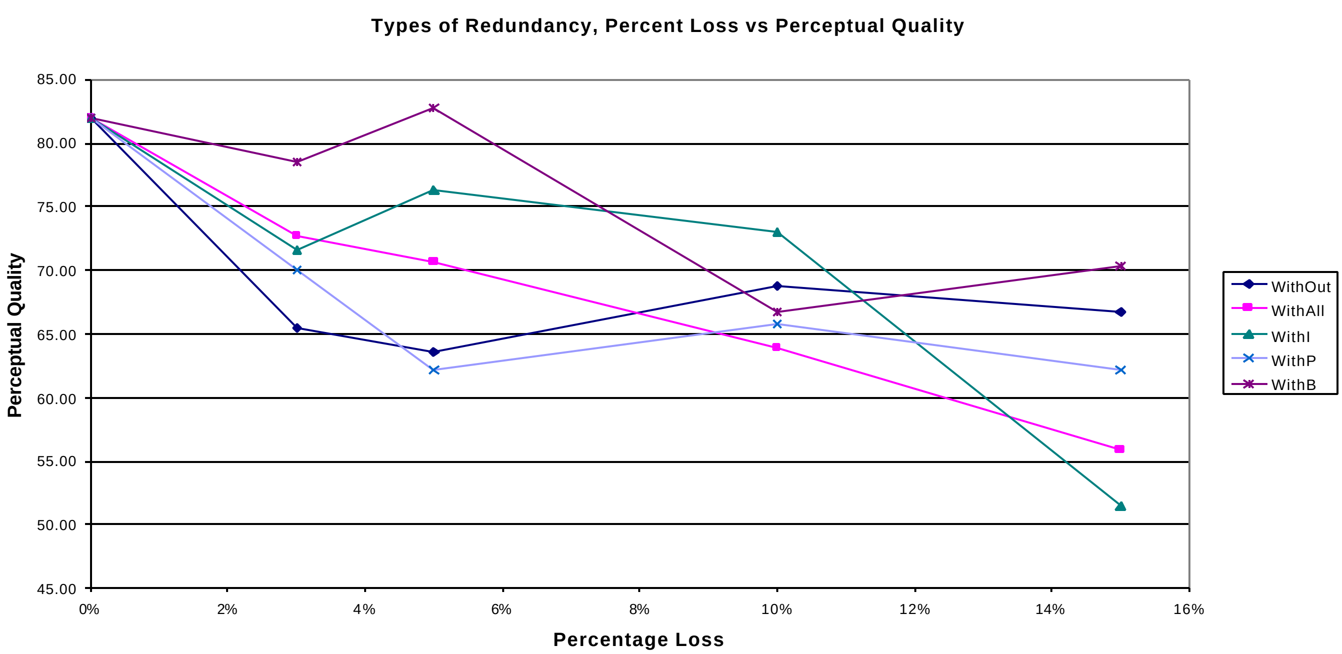
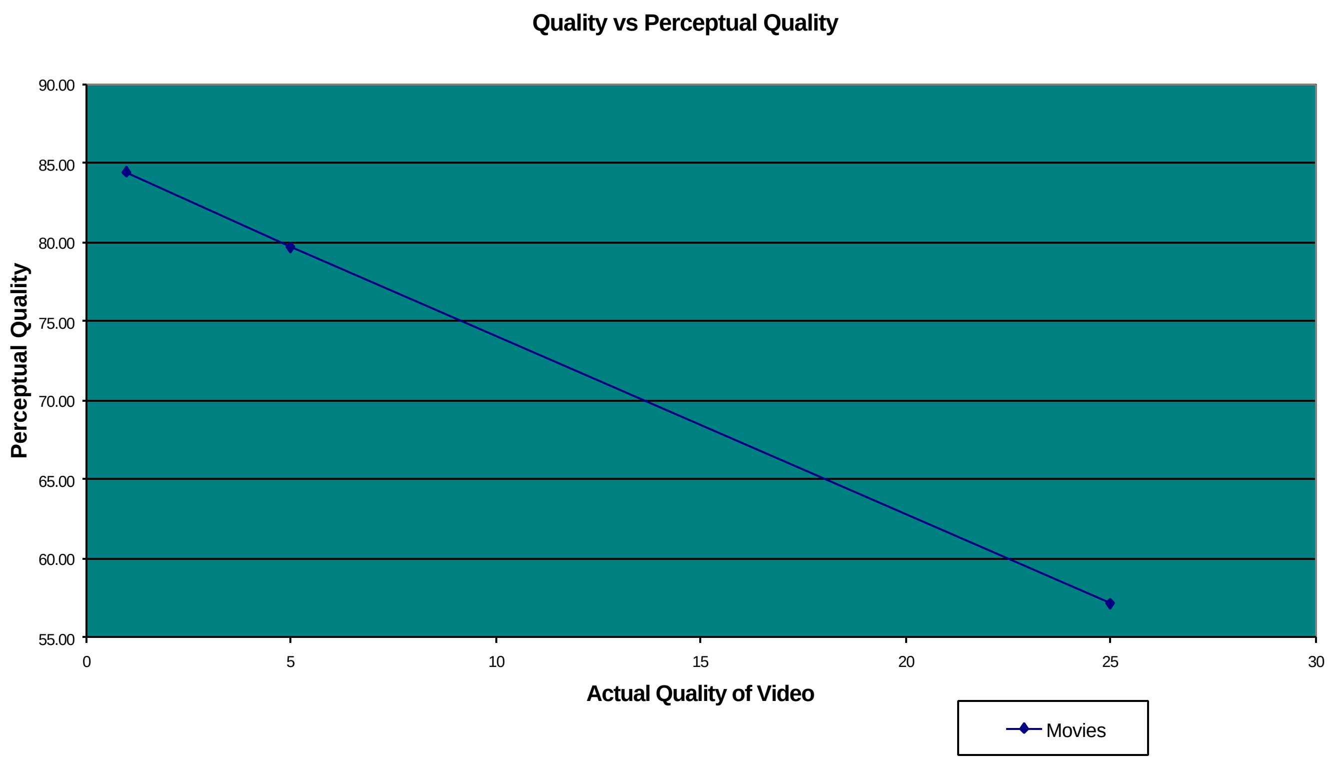


Figure 6.4.4: Percentage Loss vs Perceptual Quality

6.4.5 Levels of Quality vs Perceptual Quality

Since the primary frames of the video clips are a quality of 5 and the secondary video clips are a quality of 25, the users were given perfect video clips of these qualities to rate. Users also rated perfect video clips with an MPEG quality of 1 because a comparison can be made with qualities of 1 and 5. This information will be a contribution in determining which MPEG quality should be used as primary frames based on size and perceptual quality. Appendix A includes the movie sizes of the movies we built. Some movie clips with redundancy schemes built into them have larger file sizes because we wanted to imitate what the end users would see. Figure 6.4.6 shows Yanlin Liu's analysis of the size of MPEG movies having different qualities. There is an exponential increase in file size as the MPEG quality increases. However, the movies that we built were only a quarter the length of the movies on which she performed her analysis. Therefore, the number of bytes of each MPEG is a quarter the size of her MPEG video clips. These ratings were necessary to see how perceptual quality correlated to the actual quality of the MPEGs, shown in Figure 6.4.5.

As shown in Figure 6.4.5, there is a linear correlation on how users view perfect video clips with the video clip qualities of 1, 5 and 25. Multimedia researchers may find using primary frames of 5 instead of 1 will save a significant amount in bytes being transferred through the Internet [LC99]. In addition, there is a linear decrease in the perceptual quality viewed by the end users.



Figrue 6.4.5: Actual Quality of Video vs Perceptual Quality

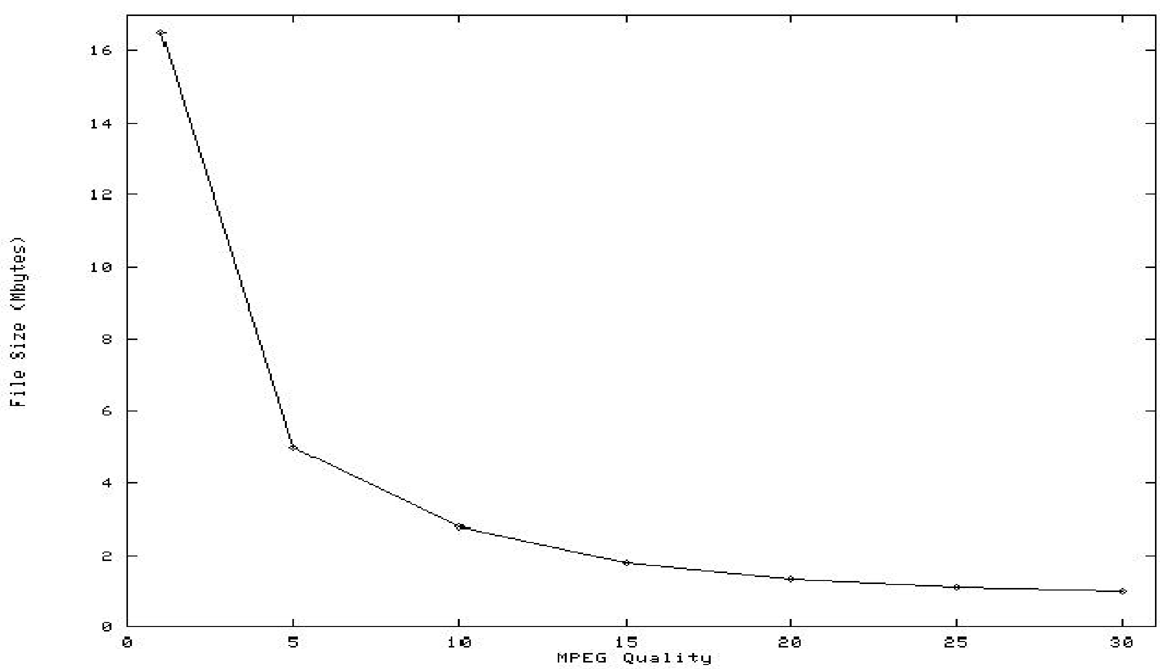


Figure 6.4.6: MPEG size vs actual MPEG quality [LC99]

Chapter 7: Conclusion

Research and study of repair techniques to alleviate the effects of data loss on a network has become a more pressing issue in recent years. Usage of multimedia applications involving the transmission of video and audio will only increase due to the ubiquity of the Internet in society. It is not feasible to use the traditional approaches (i.e., transmission, acknowledgements, etc.) to alleviate multimedia packet loss due to the time sensitivity of multimedia applications. Studying new or improved methods of alleviating data loss and the resulting effects of the loss on perceptual quality will become an important issue as usage of these applications continue to grow.

One of the areas that researchers have focused on is forward error correction, a form of sender-side repair. The advantage with this scheme is that it does not result in additional latency associated with using acknowledgements. The results from various experiments run on a simulated network on NS suggests that there is not a clear relationship between the redundancy scheme used and the amount of data that is lost. This may be caused by analyzing a small proportion of redundancy data added onto a large amount of the data stream.

Through a user study we have gathered that in typical network congestion, where the packet loss is less than five percent, there is a positive correlation between the amount of redundancy used and the perceptual quality. The more redundancy bytes are added to the network, the higher the perceptual quality rating it received. However, in a high network loss situation, perceptual quality did not have a clear correlation to the different redundancy schemes we examined. The perceptual quality tests suggests that B-frame redundancy results in the best performance in the three to five percent loss range. B frames comprised approximately sixty-seven percent of the frames in the group of pictures. Because of their sheer numbers, they have a greater chance of being lost than either the I or P frames. This may explain why incorporating redundant B frames into the data stream boosted perceptual quality.

An overview of our project and issues we dealt with in researching a repairing scheme for multimedia video deliverance:

- Research on what has been accomplished in the past
    - Importance of repairing multimedia streams
    - Different repairing schemes
        - On Video and Audio
    - Makeup of MPEG
    - Types of packets being delivered on the Internet
- Network Analysis
    - How various parameters affected network congestion
        - Different types of topologies
        - Queue type, size
        - Bandwidth
        - Different startup times of senders
        - Frames per second
        - Sender types (TCP and UDP) mix
    - Network Simulator (NS)
        - Setting up NS
        - Otcl files
        - Understanding the NS output files
    - Analysis
        - Writing Perl scripts to process the NS data
        - Graphing and analyzing the data
        - Measurement of congestion by evaluating packet loss and throughput
    - Our Contribution

        - Using Topology 3, we found different percent drops with certain redundancy schemes that we were analyzing.
        - Found that there is no correlation between more bytes added to the network increases percent drops

- Perceptual Quality Study
    - Building movies by using the repairing schemes we came up with (redundancy using I, P, B frames only, redundancy using all lower quality frames and no redundancy)
        - Automation of movie building using Perl script
        - Incorporated combination of MPEG quality (1, 5, 25) and amount loss (0, 3, 5, 10, 15) when building the movies
    - Building Visual Basic User Interface
        - Using user interface design techniques learned in HCI class to design a user friendly interface to display the movies
        - Build a Visual Basic interface with embedded Media Player
        - Microsoft Access database to store the movie ratings
    - Conduct User Study
        - Tried to eliminate any biases in the way users rated the movies by randomizing the sequence of movies that are presented to users
        - Made flyers to ask users to help us with this research
    - Graphed and Analyzed the User Study results
    - Contribution
        - Looked at the redundancy scheme of the movies and perceptual quality ratings
        - Looked at percentage loss in the video clips and perceptual quality
        - Analyzed the actual quality of MPEG video clips versus perceptual quality ratings
        - Did a comparison of the packet sizes of MPEG quality of 1, 5 and 25 versus the perceptual quality ratings these video clips received
        - Present ideas for future work

Chapter 8. Future Work

There remain many areas that may be further explored with respect to research in the field of multimedia applications and the Internet. The experiments that were performed in this project suggested that using B-frame redundancy achieves the best results in terms of improving perceptual quality for the user. Researchers may want to analyze the effect on perceptual quality that one type of packet loss has over another and verify this. Studying these problems from a higher level involves further analysis of the Group of Pictures (GOP).

There can exist many Group of Pictures used for any MPEG. The experiments performed in this project adhered to only one GOP. Since I, P and B frames all have various dependencies on one another, the placement and quantity of these frames within a GOP may have various effects on the final perceptual quality of the MPEG. Therefore, one other avenue that a researcher may look into involves examining how the specific GOP composition effects user perceptual quality and data loss over a network.

In the perceptual user study, we have noticed that the ratings may be affected by the content of the video clips. The contents that can be examined include speed of movement, color of clips and type of video clips. Examining how perceptual quality is affected by the contents of video clips can help multimedia researchers in deciding on what type of repairing techniques to use during varying usage of the multimedia streams.

References

[BAS98] Tim Bass, *Traffic Congestion Measurements in Transit IP Networks*, Science Applications International Corporation Center for Information Protection McLean, Virginia, http://www.silkroad.com/papers/html/cong/n1.html, 1998.

[CC00] Jae Chung, Mark Claypool, *Dynamic CBT – Router Queue Management for Improved Multimedia Performance on the Internet*, Department of Computer Science, Worcester Polytechnic Institute, Spring 2000.

[CC200] Jae Chung and Mark Claypool, *Better-Behaved, Better-Performing Multimedia Networking, SCS Euromedia Conference (COMTEC)*, Antwerp, Belgium, May 8-10, 2000.

[CR99] Mark Claypool, John Riedl, "End-to-End Quality in Multimedia Applications," *Handbook on Multimedia Computing*, CRC Press, Boc Raton, FL 1999.

[GKL+98] Steven Gringeri, Bhumip Khasnabish, Arianne Lewis, Khaled Shuaib, Roman Egorov, and Bert Basch, *Transmission of MPEG-2 Video Streams over ATM*, IEEE Multimedia, Jan-Mar 1998.

[HSK98] Vicky Hardman, Martina Angela Sasse, and Isidor Kouvelas, *Successful Multiparty Audio Communication Over the Internet*, Communications of the ACM, Vol. 41, No.5 pp. 74 - 80, May 1998.

[KR00] James F. Kurose and Keith W. Ross, *Computer Networking: A Top-Down Approach Featuring the Internet*, http://gaia.cs.umass.edu/kurose/transport/UDP.html, 2000.

[LC99] Yanlin Liu, Mark Claypool, *Video Redundancy - A Best-Effort Approach to Network Data Loss*, Department of Computer Science, Worcester Polytechnic Institute, May 1999.

[mpo] MPEG Organization, http://www.mpeg.org/MPEG/#mpeg

[ns] Jae Chung and Mark Claypool, *NS by Example*, http://saagar.wpi.edu/NS/

[PHH98] Colin Perkins, Orion Hodson, Vicky Hardman, *A Survey of Packet-Loss Recovery Techniques for Streaming Audio*, IEEE Network Magazine, September/October 1998.

[PNC99] Joseph D. Petruccelli. Balgobin Nandram and Minghui Chen, *Applied Statistics for Engineers and Scientists*, Prentice Hall, Upper Saddle River, New Jersey, 1999.

[POS80] J. Postel, *User Datagram Protocl*, RFC 768, August 1980, ftp://ftp.isi.edu/in-notes/rfc768.txt, 1980.

[SIK97] Dr. Thomas Sikora, MPEG Video Webpage, http://bs.hhi.de/mpeg-video, 1997.

[TAN96] Andrew S. Tanenbaum, *Computer Networks*, Prentice Hall PTR: Upper Saddle River, NJ, 1996.

[TC99] Jonathan Tanner, Mark Claypool, *Java Jitter - The Effects of Java on Jitter in a Video Stream*, Department of Computer Science, Worcester Polytechnic Institute, May 1999.

[tcp] Information on TCP, http://sregora.com/info/tcpnet.html

[web99] http://webopedia.internet.com/TERM/M/MPEG.html, 1999.

[WS95] Anna Watson, Martina Angela Sasse, *Evaluating Audio and Video Quality in Low-Cost Multimedia Conferencing Systems*, Department of Computer Science, University College London, December 1995.

Appendix A: Movie Sequence and Ratings

Movie ID	Redundancy / Quality	Percent Loss	Type of Movie	Average - no 100	Movie Size KB	File Name
1	Quality 5	0	Sports1	82.00	3,317	ski3
2	WithOut	5	Cartoon1	63.50	7,297	simp3
3	WithAll	15	Sports2	55.83	7,714	soccer1
4	WithB	10	SitCom1	66.67	4,861	married3
5	WithP	3	News1	70.07	4,997	cnn7
6	WithI	15	SitCom2	51.39	3,522	married1
7	Quality 5	0	News2	79.59	1,617	news2
8	Quality 1	0	Cartoon2	79.69	1,829	simp8
9	WithAll	5	News3	70.72	6,000	cnn6
10	WithP	10	Sports3	65.84	8,589	hockey1
11	WithB	3	Cartoon3	78.50	5,627	simp4
12	WithI	10	News4	72.33	5,081	cnn3
13	Quality 25	0	Shopping1	53.65	609	hs3
14	WithOut	3	SitCom3	65.45	6,045	third1
15	WithI	5	Sports4	75.27	5,562	Game1
16	WithP	15	Cartoon4	62.11	5,499	simp5
17	Quality1	0	News5	89.21	9,734	cnn5
18	WithAll	3	News6	72.72	4,375	news1
19	WithOut	15	Shopping2	66.73	6,524	hs1
20	WithB	15	News7	70.26	5,284	cnn8
21	Quality 5	0	Cartoon5	77.76	1,749	simp9
22	WithI	3	News8	70.93	5,029	Cnn1
23	WithB	5	Sports5	82.73	8,686	ski1
24	Quality 25	0	News9	60.67	506	cnn4
25	WithAll	10	Cartoon6	63.11	5,658	simp2
26	WithP	5	SitCom4	62.11	4,131	married2
27	WithOut	10	News10	68.80	5,502	cnn2

Appendix B: User Demographic Data

Age	Count
17	7
18	6
19	3
20	5
21	9
22	3
23	1
25	3
26	2
27	2
28	1
29	1
31	1
36	1
40	1

Major	Count
Chem Eng	1
Chemistry	1
CM	2
computer sciece	1
Computer Science	5
CS	9
CS/EE	1
cs/sd	1
EE	3
electrical engineering and computer science	1
IE	1
Management	1
MEA	1
Mechanical Engineeering	1
ND	1
None	2
Physics	1
Psycho	1
Undecided	9
Undeclared	1

Computer Familiarity	Count
Novice	11
Adept	23
Wizard	12

PC Media Experience	Count
Never	4
Occasionally	27
Frequently	15

Internet Media Experience	Count
Never	8
Occasionally	29
Frequently	9

Appendix C: User Study Flyer

Help us with our MQP!

Come View and Rate Some Videos....

Go To The Movie Lab

- Log onto the computer by typing ***mpeg*** for both the loginame and password.
- Run "Programs"->"User Study.exe"
- This experiment will take less than 10 minutes of your time.
- For more information please go to http://www.wpi.edu/~lzhang/MQP

Our MQP is to study how users view different types of Multimedia repair methodologies. We have created a Visual Basic user interface to display MPEG movies. These movies may be perfect or have been altered using different types of repair techniques. The user input will be collected and analyzed. The user information will point us to a better way of doing multimedia repairs.

Please be assured that your privacy will be respected. It is optional to enter in your real name and e-mail address in the VB user interface.

The Visual Basic user interface will be available in the Movie Lab (on the basement floor of Fuller, to your right when you enter the building through the glass doors). The VB user interface will be there from 3/27/2000 to 3/31/2000.

Please provide us with any feedback, such as comments or bugs, through email:

Brandon Ngo: bgno@wpi.edu or Lisa Lei Zhang: lzhang@wpi.edu

Thank you for your time and participation!