

University of Alberta

MINT

(Master's in Internetworking)

Department of Computer Science

Faculty of Science

“IPTV Quality of Experience”

(Project Report – MINT 709)

Submitted by:

Ritika Bhansali

Supervisor:

Dr. Mike MacGregor

March 2010

ACKNOWLEDGEMENTS

I owe deepest gratitude to my supervisor Dr. Mike MacGregor for his immense support, guidance and patience throughout this project. His advice and encouragement made this project successful.

I would like to extend my gratitude to Prof. Bischof for providing his time and feedback for this research study.

I also want to thank my colleague Johan Johan for briefing his project to me. I have extended his project and his help in initial set-up saved me lot of time and energy.

I am indebted to many of my friends who offered their time and help in this research study. I am also grateful to Indian Student Association (INDSA) at University of Alberta for providing volunteers for this study.

Last but not least I am thankful to my Husband, Mranal Jain, who kept me motivated and supported me in all possible way to finish my degree successfully.

I would like to dedicate this degree to my son, Aarush Jain.

Table of contents

Acknowledgements	2
Table of Contents	3
Introduction	4
Initial Phase	5
Implementation Phase	7
Results and Discussion	12
Conclusions and Suggestions	19
References	20
Appendix A	21

1. Introduction

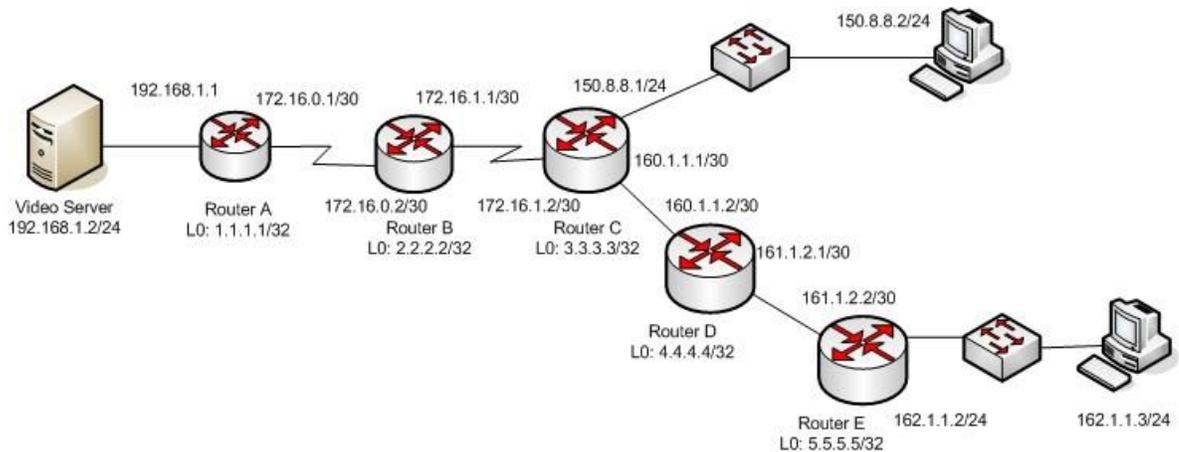
IPTV (Internet Protocol Television) is a system where a digital television service is delivered using Internet Protocol over a network infrastructure. In this service, initially the TV signals are encoded in IP format. Further, these encoded data packets are delivered to appropriate destinations using internet network and finally at the client locations, using suitable decoder, these data packets are converted back to the TV signals. IPTV offers various advantages over traditional TV i.e. the quality of digital video and audio is much better compared with the traditional analogue TV. Secondly, as IPTV uses standard networking protocols, it promises lower costs for operators and lower prices for users. Finally, interactive features can be added to the IPTV i.e. Video-on-Demand, incoming call display and other interactive features. Most of these features are already available in the IPTV packages offered by various IPTV service providers.

As IPTV is based on internet protocols, the quality of IPTV is dependent upon the packet loss and delays associated with the network. This study deals with the assessment of acceptable errors in the IPTV signal transmission. The organization of this report is as follows: the next section provides details about initial phase and previous studies related to this project. The following section outlines the implementation aspects of this project. In results section, user-data is presented in form of tables and graphs followed by the conclusions.

2. Initial Phase

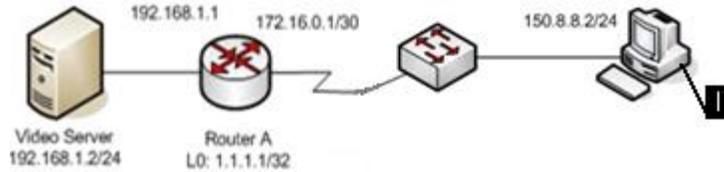
In the first part of this project, a laboratory scale IPTV set-up (figure 1) is arranged such that different types of errors i.e. packet loss, corruption, delay and duplication can be introduced through the network. It is worth mentioning that this part has been done earlier by my colleague Mr. Johan (as part of his MINT project). As the current study is a follow-up study, I replicated his work and implemented the multicast network in MINT lab with few changes. The details about the previous study can be found in the following project report, “Impact of Network Errors on User Perception of Streaming Video” by Mr. Johan. The network topology shown in figure 1 utilizes various Cisco Routers (series 2600, 2800 & 3600), Switches (Catalyst 3750 & 3500) and Video server with VLC Player.

Figure 1. Network topology for multicast network



As the present study requires one user involvement at a time, a simplified set-up is adapted from figure 1. The network topology for the used set-up is as shown in figure 2 which involves a Router (2600 Series), Switch (catalyst 3750) and slider.

Figure 2. Network topology design for single user.



After creating this set-up, video server is employed to broadcast video using VLC Player. VLC player is a free cross-platform multimedia player released under the GNU General Public License. VLC is used in this project to stream video packets to a multicast network.

The following VLC's commands are used to stream a movie to a multicast network:

```
vlc /home/movies/gafla.mpg --ttl 16 --sout  
'#duplicate{dst=std{access=rtp,mux=ts,dst=239.255.0.1:1234,sap,name="testing"  
,dst=display}'
```

/home/movies/gafla.avi is sample movie stored on Server. The other movie used was *Paycheck.avi*

access=rtp means that RTP is used to stream packets.

mux=ts means that streams' s encapsulation method, which is MPEG2-TS muxer.

dst is the destination.

"testing" is the name displayed at user side for streamed video.

3. Implementation Phase

In this phase, user/ client responses are obtained for different types of network errors. Various errors are introduced at video network and subsequently presented to the user for quality evaluation. The IPTV video quality is quantified using a slider with a range between 0 (lowest quality) and 1000 (highest quality). The user response in form of slider readings are directly recorded on video server as text files.

There are various types of network errors like packet loss, packet corruption, packet duplication, packet re-ordering and packet delay that could affect the IPTV quality. Among these errors, packet loss and corruption have significant impact on quality of video as compared to others. Therefore, effect of these errors on IPTV quality is being evaluated by the user. Both of these errors are discussed below:

Packet loss

Packet loss occurs when one or more packets of data traveling across a computer network fail to reach their destination. Packet loss can be caused by a number of factors, including signal degradation over the network medium, oversaturated network links, corrupted packets rejected in-transit, faulty networking hardware, faulty network drivers or normal routing routines. When caused by network problems, lost or dropped packets can result in highly noticeable performance issues or jitter with streaming technologies like IPTV.

Command: *tc qdisc add dev eth0 root netem loss 0.03%*. This command will introduce 0.03% loss in network.

Packet corruption

Packet corruption occurs when the contents of a packet are damaged during transmission of data. The cause of packet corruption and its effects on video quality are similar as of packet loss as mentioned in the previous paragraph.

Command: *tc qdisc add dev eth0 root netem corrupt 0.25%*. This command will corrupt 0.25% data in network.

Error sequencing and introduction

Initially, various set of errors were introduced ranging from .01% to 2.5% error. The first error was noticed at .03% and any error more than 2% appears same. Using random generator ten values were selected in the range of .03 % to 2%. Those values were than arranged in different sequences. A set of error values is shown below for a user in table 1. Using same values a different sequence (by randomly arranging the errors) is generated for another user. The user is asked to evaluate ten such sequences introduced for loss and corruption alternatively.

After numerous experiments and testing, Dr. Mike, Dr. Bischoff and I decided to conduct user study by demonstrating the worst (2% error) and best (0% error) videos in beginning. Once user gets accustomed to both qualities then errors are introduced in above sequence. A set of worst and best errors are added in beginning of each segment to afresh both the qualities of error. Each error lasts for 20 seconds.

Using slider, the visual quality of video was evaluated by the users. Slider was marked “high” for best quality and “low” for worst quality. Users were asked to continuously move the slider depending on the quality of video. The values given by user was recorded on client computer using software program slider.c . The Slider is a USB device which was installed on client computer using program Phidget.c.

Table 1. The randomly arranged error sequences are shown for single user.

No.	% of error									
1.	1.1244	2	0.9056	0.4678	0.6867	0.03	1.7811	1.5622	1.3433	0.2489
2.	1.1244	1.5622	1.7811	1.3433	2	0.4678	0.9056	0.6867	0.2489	0.03
3.	0.2489	2	1.3433	1.7811	0.03	1.1244	0.9056	1.5622	0.4678	0.6867
4.	2	0.9056	1.1244	0.4678	0.2489	0.6867	1.7811	0.03	1.3433	1.5622
5.	0.6867	0.9056	1.1244	0.03	0.2489	2	1.7811	1.3433	1.5622	0.4678
6.	2	1.1244	0.6867	0.2489	1.3433	0.03	1.5622	0.9056	0.4678	1.7811
7.	1.1244	0.6867	0.9056	0.4678	1.3433	1.5622	0.2489	2	1.7811	0.03
8.	0.9056	0.2489	0.6867	1.5622	2	1.7811	0.03	0.4678	1.3433	1.1244
9.	1.5622	1.1244	0.03	1.3433	1.7811	0.6867	0.2489	2	0.4678	0.9056
10.	1.5622	0.6867	1.7811	0.03	2	0.4678	1.3433	0.9056	0.2489	1.1244

Error Initialization on video server

To introduce error, ten text files are created and saved locally on server. Using command window these errors files are called and enforced on network. One of the file is demonstrated below. The command starts with 20, which means 20 sec interval followed by error command.

```
20
tc qdisc add dev eth0 root netem corrupt 2.0000%
20
tc qdisc change dev eth0 root netem corrupt 0.0%
20
tc qdisc change dev eth0 root netem corrupt 1.1244%
20
tc qdisc change dev eth0 root netem corrupt 0.9056%
20
tc qdisc change dev eth0 root netem corrupt 0.2489%
20
tc qdisc change dev eth0 root netem corrupt 1.3433%
20
tc qdisc change dev eth0 root netem corrupt 1.5622%
20
tc qdisc change dev eth0 root netem corrupt 1.7811%
20
tc qdisc change dev eth0 root netem corrupt 2.0000%
20
tc qdisc change dev eth0 root netem corrupt 0.0300%
20
tc qdisc change dev eth0 root netem corrupt 0.4678%
20
tc qdisc change dev eth0 root netem corrupt 0.6867%
20
tc qdisc del dev eth0 root
```

After running this file on server, a text file is generated which contains time and error information for above introduced file and looks like

```

Fri Jul 24 17:58:29 2009 tc qdisc add dev eth0 root netem corrupt 2%
Fri Jul 24 17:58:49 2009 tc qdisc change dev eth0 root netem corrupt 0.00%
Fri Jul 24 17:59:09 2009 tc qdisc change dev eth0 root netem corrupt 1.12%
Fri Jul 24 17:59:29 2009 tc qdisc change dev eth0 root netem corrupt 0.91%
Fri Jul 24 17:59:49 2009 tc qdisc change dev eth0 root netem corrupt 0.25%
Fri Jul 24 18:00:09 2009 tc qdisc change dev eth0 root netem corrupt 1.34%
Fri Jul 24 18:00:29 2009 tc qdisc change dev eth0 root netem corrupt 1.56%
Fri Jul 24 18:00:49 2009 tc qdisc change dev eth0 root netem corrupt 1.78%
Fri Jul 24 18:01:09 2009 tc qdisc change dev eth0 root netem corrupt 2.00%
Fri Jul 24 18:01:29 2009 tc qdisc change dev eth0 root netem corrupt 0.03%
Fri Jul 24 18:01:49 2009 tc qdisc change dev eth0 root netem corrupt 0.47%
Fri Jul 24 18:02:09 2009 tc qdisc change dev eth0 root netem corrupt 0.69%
Fri Jul 24 18:02:30 2009 tc qdisc del dev eth0 root

```

This information is combined with data collected from user on client computer to determine error and its corresponding slider values for given time. Every user is asked to experience one hour of IPTV quality subjected to above mentioned error sequences. Prior to the data collection, clocks on video server and client computer were synchronized using following command:

```
sudo ntpdate -u time.srv.ualberta.ca
```

4. Results and Discussion

The analysis is carried out for data obtained from 18 users is analyzed. Several tables and figures are plotted to demonstrate user interpretation of error. A part of user reading (raw data) is shown below in table 2. The day, date and time are shown in column A, B, C, D & E, while column G & H corresponds to the Slider reading input by the user. The values in Column I Shows the start time for the different magnitude of errors. The % error value corresponding to time value in column D is shown in column J. The complete data set, even for a single user, is too large to be presented in this report. The full data sets for all users are enclosed with this report in form of CD (Compact Disc).

Table 2. A sample of user reading.

A	B	C	D	E	F	G	H	I	J
Sat	Aug	8	17:24:35	2009	Sensor	1	186	17:24:35	0.91%
Sat	Aug	8	17:24:35	2009	Sensor	1	197	17:24:55	0.91%
Sat	Aug	8	17:24:35	2009	Sensor	1	208	17:25:15	0.91%
Sat	Aug	8	17:24:35	2009	Sensor	1	220	17:25:35	0.91%
Sat	Aug	8	17:24:35	2009	Sensor	1	231	17:25:55	0.91%
Sat	Aug	8	17:24:35	2009	Sensor	1	244	17:26:15	0.91%
Sat	Aug	8	17:24:35	2009	Sensor	1	255	17:26:35	0.91%
Sat	Aug	8	17:24:35	2009	Sensor	1	267	17:26:55	0.91%
Sat	Aug	8	17:24:37	2009	Sensor	1	257	17:27:15	0.91%
Sat	Aug	8	17:24:37	2009	Sensor	1	246	17:27:35	0.91%
Sat	Aug	8	17:24:37	2009	Sensor	1	236		0.91%
Sat	Aug	8	17:24:40	2009	Sensor	1	260		0.91%
Sat	Aug	8	17:24:40	2009	Sensor	1	280		0.91%
Sat	Aug	8	17:24:40	2009	Sensor	1	305		0.91%

We begin our analysis with a single user data. After importing all data in excel, ten sub-sheets are created for loss and corruption for every user. In figure 3 and 4, raw

user data (obtained from slider) is plotted against time for packet corruption and packet loss. The vertical lines in figure 3 and 4 correspond to the start time for the shown % of error. The delay in user response, immediately after the change in error magnitude is apparent from figure 3 and 4. It is important to note that such delays in user-response cannot be avoided and would contribute towards the errors in analysis.

Figure 3. User’s slider response is plotted against time for a single packet corruption error sequence. The vertical lines correspond to the start time for the shown % of packet corruption.

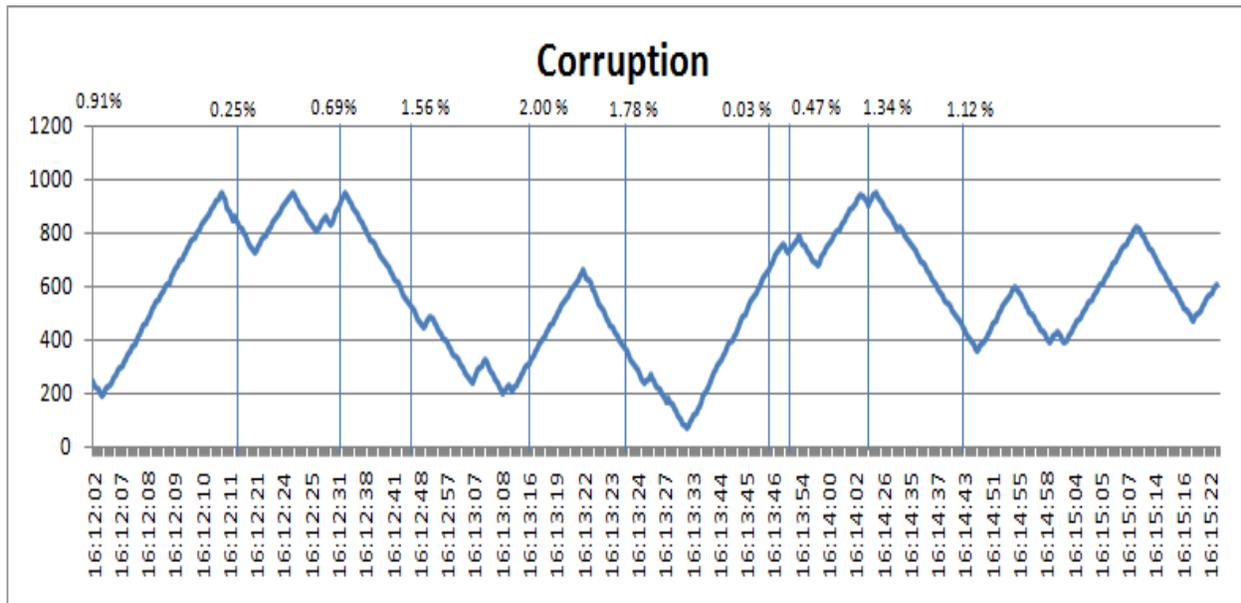
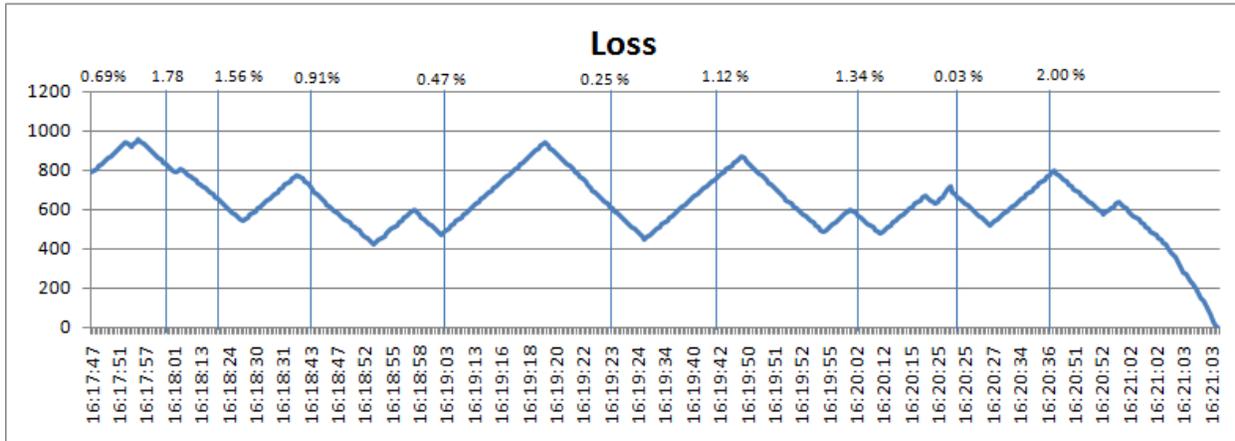


Figure 4. User’s slider response is plotted against time for a single packet loss error sequence. The vertical lines correspond to the start time for the shown % of packet corruption.



Next, the mean values and standard deviation are calculated for all error sequences for single user (for both packet loss and packet corruption). Both mean slider values and standard deviation are shown for single user in table 3 (for packet corruption error) and table 4 (for packet loss error).

Table 3. Mean and Standard deviation values for single user (10 sequences) corresponding to packet corruption error.

Error Percentage	Mean	SD
0.03%	599.45	188.96
0.25%	667.28	230.11
0.47%	635.01	140.23
0.69%	448.69	182.62
0.91%	592.38	184.47
1.12%	551.14	111.95
1.34%	486.97	189.52
1.56%	370.31	145.92
1.78%	447.73	188.86
2.00%	380.25	192.76

Table 4. Mean and Standard deviation values for single user (10 sequences) corresponding to packet loss error.

Error Percentage	Mean	SD
0.03%	724.13	109.26
0.25%	708.15	147.01
0.47%	701.27	136.27
0.69%	584.61	176.19
0.91%	509.53	178.25
1.12%	439.54	204.97
1.34%	474.17	142.27
1.56%	316.56	188.12
1.78%	449.83	224.35
2.00%	306.54	186.55

As shown in above tables the lesser the error, the higher the averages are. Since error introduction is not in sequence, the value provided by user for error depends on what percent of error was preceded. If low error (say 0.03 %) was introduced after high error (say 2.0 %), then there will be some after effects on the low error reading and vice versa. The trend between user response and error magnitudes is more apparent with the scatter plots shown in figure 5 and figure 6. An approximate linear decline in the slider values are observed for increase in error magnitude (figure 5).

Figure 5. Scatter plot showing mean response (for ten sequences) for packet corruption error and packet loss error.

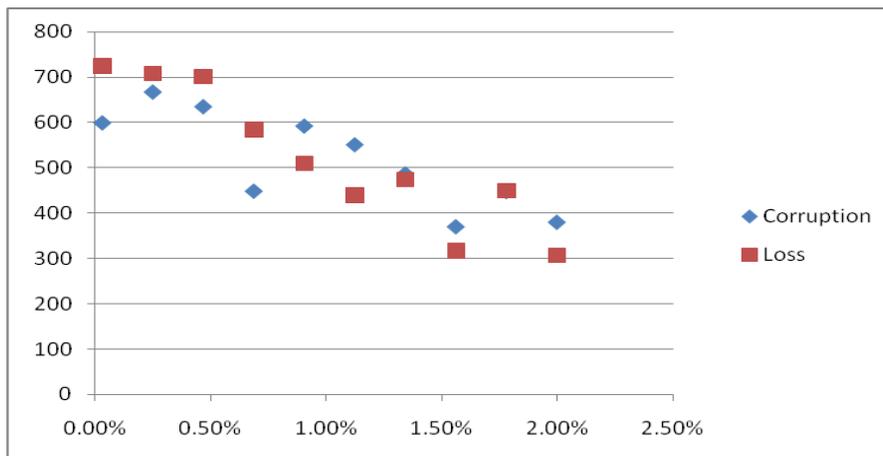
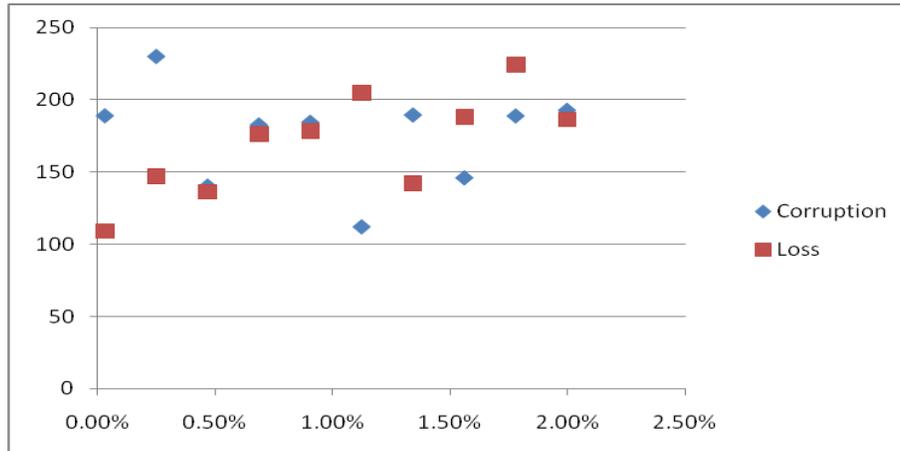


Figure 6. Scatter plot showing user standard deviation (for ten sequences) for packet corruption error and packet loss error.



The typical value for standard deviation is 15-20% i.e. equal to 150-200 in terms of slider value as shown in figure 6. This can be attributed to delay in user response time for randomly generated error sequence.

Figure 7. Scatter plot for mean response plot for packet corruption error for all users.

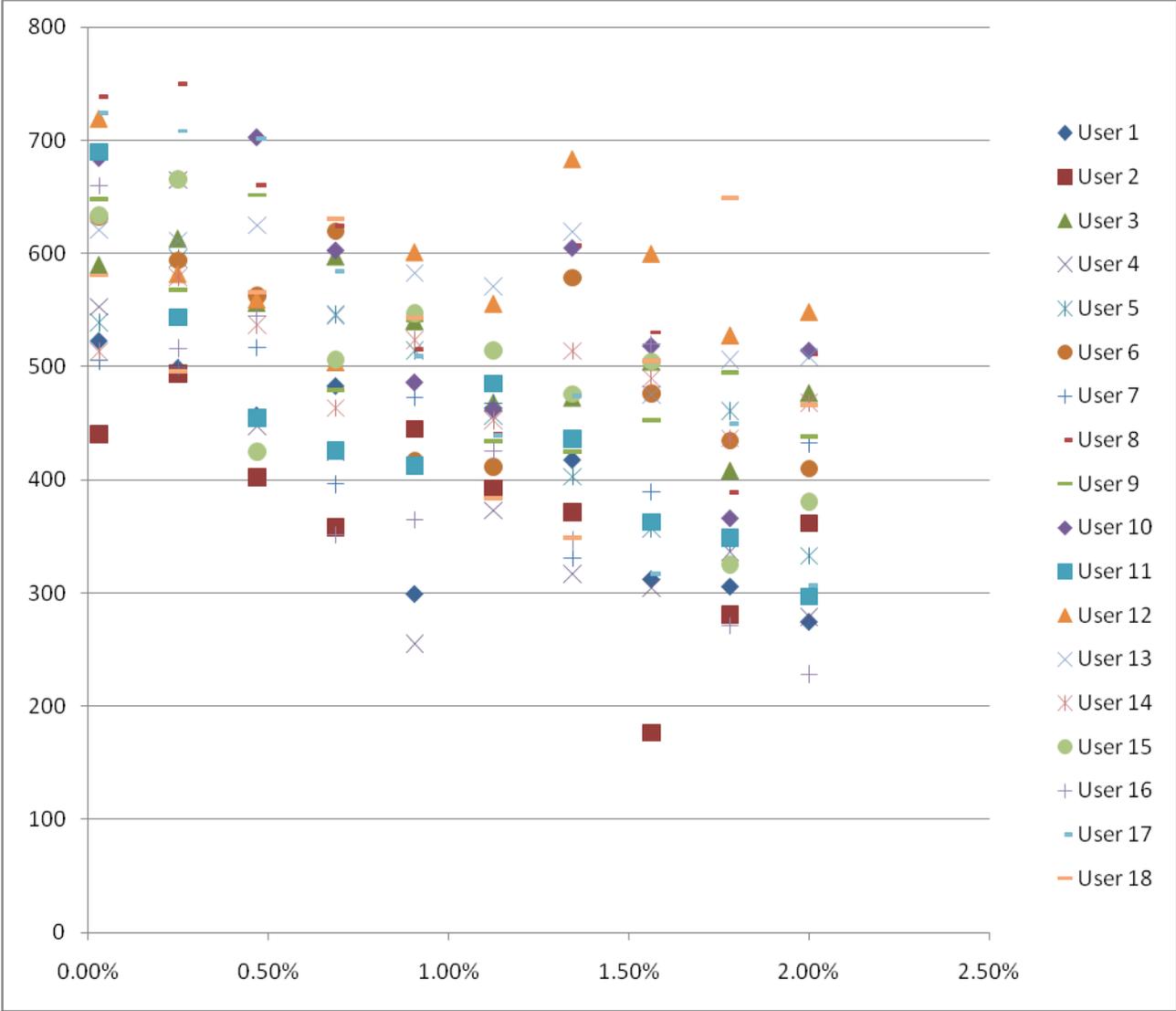
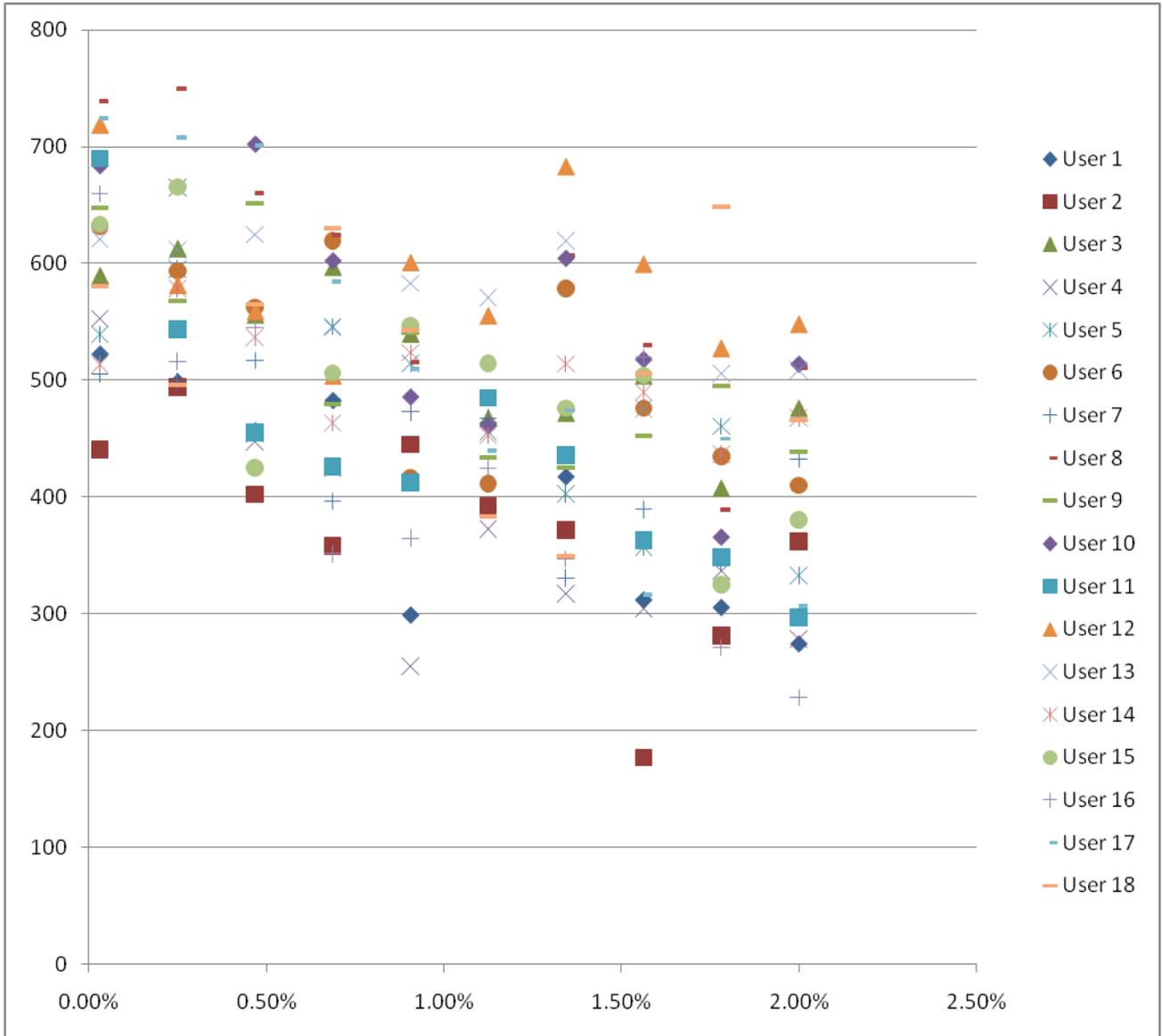


Figure 8. Scatter plot for mean response plot for packet loss error for all users.



The mean user response (slider values) is shown as scatter plot for all users in figure 7 (for Packet Corruption error) and figure 8 (Packet Loss error). Again, a linear decrease in slider values is evident with increase in error magnitude. For a given % of error, the range of mean values is observed as similar error magnitude is perceived differently by each user.

5. Conclusions and Suggestions

The primary goal of this project report is to determine error ranges and user perception of those errors. After determining error range and gathering user data the following can be concluded

- Amongst two kinds of error introduced, loss and corruption, results of loss is more in accordance with error percentage than corruption. It could be because loss in packets has higher impact than corrupted packets and user perceived those errors better.
- Cache and lag in video have pivotal role in user reading. The cached video in VLC player causes lag in error introduction, which in result causes reading from one error to interfere with another.
- The graphs and user readings suggest that the error introduced below 1 % is acceptable any error more than that results in poor quality which is intolerable.

While analyzing user data, I have realized on various occasion especially when error is low, user have not changed slider position continuously. I would like to suggest that when user has not moved slider, the reading should still be recorded for that time based on position of slider. It will help in getting better values as the more the values are the better the average is.

6. References

1. Project Report “Impact of Network Errors on User Perception of Streaming Video” by Mr. Johan Johan
2. “IPTV” - <http://en.wikipedia.org/wiki/IPTV>
3. “Packet Loss” - http://en.wikipedia.org/wiki/Packet_loss
4. “IP Multicast” - <http://www.cisco.com/en/US/docs/internetworking/technology/handbook/IP-Multi.html>
5. “Netem” - <http://www.linuxfoundation.org/collaborate/workgroups/networking/netem>
6. “VLC Media Player” - <http://www.videolan.org/vlc/>

7. Appendix A

A>en

A#sh run

Building configuration...

Current configuration : 799 bytes

!

version 12.3

service timestamps debug datetime msec

service timestamps log datetime msec

no service password-encryption

!

hostname A

!

boot-start-marker

boot-end-marker

!

!

memory-size iomem 10

no aaa new-model

ip subnet-zero

!

!

no ip domain lookup

!

ip multicast-routing

ip cef

!

!

!

!

!

!

!

!

!

!

!

!

```
!  
!  
!  
!  
!  
interface FastEthernet0/0  
ip address 192.168.1.1 255.255.255.0  
ip pim sparse-mode  
duplex auto  
speed auto  
!  
interface Serial0/0  
no ip address  
shutdown  
fair-queue  
!  
interface FastEthernet0/1  
ip address 150.8.8.1 255.255.255.0  
ip pim sparse-mode  
duplex auto  
speed auto  
!  
interface Serial0/1  
no ip address  
shutdown  
!  
ip http server  
ip classless  
!  
!  
!  
!  
!  
voice-port 1/0/0  
!  
voice-port 1/0/1  
!  
!  
dial-peer cor custom
```

```
!  
!  
!  
!  
line con 0  
line aux 0  
line vty 0 4  
  login  
!  
!  
end
```

```
Switch>en  
Switch#sh run  
Building configuration...
```

```
Current configuration : 2007 bytes  
!  
! Last configuration change at 16:08:14 MST Wed Jul 1 2009  
! NVRAM config last updated at 16:08:16 MST Wed Jul 1 2009  
!  
version 12.2  
no service pad  
service timestamps debug uptime  
service timestamps log uptime  
no service password-encryption  
!  
hostname Switch  
!  
!  
no aaa new-model  
clock timezone MST -7  
clock summer-time MST recurring  
switch 1 provision ws-c3750g-24ps  
ip subnet-zero
```

```
ip routing
no ip domain-lookup
!
ip igmp snooping vlan 1 mrouter learn cgmp
ip igmp snooping vlan 1 mrouter interface Gi1/0/1
ip igmp snooping vlan 1 static 239.255.0.1 interface Gi1/0/3
ip igmp snooping vlan 1 static 239.255.0.1 interface Gi1/0/5
ip igmp snooping vlan 1 static 239.255.0.1 interface Gi1/0/7
ip igmp snooping vlan 1 static 239.255.0.1 interface Gi1/0/9
!
!
!
no file verify auto
spanning-tree mode pvst
spanning-tree extend system-id
!
vlan internal allocation policy ascending
!
!
interface GigabitEthernet1/0/1
!
interface GigabitEthernet1/0/2
!
interface GigabitEthernet1/0/3
!
interface GigabitEthernet1/0/4
!
interface GigabitEthernet1/0/5
!
interface GigabitEthernet1/0/6
!
interface GigabitEthernet1/0/7
!
interface GigabitEthernet1/0/8
!
interface GigabitEthernet1/0/9
!
interface GigabitEthernet1/0/10
!
```

```
interface GigabitEthernet1/0/11
!  
interface GigabitEthernet1/0/12
!  
interface GigabitEthernet1/0/13
!  
interface GigabitEthernet1/0/14
!  
interface GigabitEthernet1/0/15
!  
interface GigabitEthernet1/0/16
!  
interface GigabitEthernet1/0/17
!  
interface GigabitEthernet1/0/18
!  
interface GigabitEthernet1/0/19
!  
interface GigabitEthernet1/0/20
!  
interface GigabitEthernet1/0/21
!  
interface GigabitEthernet1/0/22
!  
interface GigabitEthernet1/0/23
!  
interface GigabitEthernet1/0/24
  no switchport
  no ip address
!  
interface GigabitEthernet1/0/25
!  
interface GigabitEthernet1/0/26
!  
interface GigabitEthernet1/0/27
!  
interface GigabitEthernet1/0/28
!  
interface Vlan1
```

```
no ip address
!  
ip classless  
ip http server  
!  
!  
!  
control-plane  
!  
!  
line con 0  
line vty 0 4  
no login  
line vty 5 15  
no login  
!  
end
```