University at Buffalo
The State University of New York

Performance Evaluation of video streaming in multi hop wireless mesh network

By : Santosh Kumar Dubey
Person # 5009 7059

May/16/2014

Table of Contents

Performance Evaluation of video streaming in multi hop wireless mesh network

1. Abstract

- The problem of transmitting real time video stream over ad-hoc wireless network possess a special challenge due to the demanding requirement of the data and the special network model.
- The requirement to satisfy for real time video streaming is large throughput of data and minimal and stable end-to-end delay, this characteristic of the data is required since transmission of video requires large bandwidth even with modern encoders due to the growing demand for high quality and the fact that complex encoding/decoding can't be done in real-time.
- The transmission of real time video, for usage such as video conference or live events broadcast for example, adds the requirement of minimal end-to-end delay which is required so that the transmission over the network would be negligible in order to create the effect of real-time on both ends.
- The medium of an ad-hoc wireless network is mostly challenging due to the fact that though these networks are more common today they still have relatively low bandwidth and low transmission distance compared to wide-area networks or cellular base-stations, this requires special handling for multi-hop transmission and flow-control.

2. Introduction

a) Wireless network model

We consider a problem of multiple nodes on a surface carrying each single transceiver able to either listen in one of several known frequencies (channels) or transmit in one channel to neighboring nodes in a known protocol.

In the above network we consider a set of demands for video-streams which is composed by several triplets which are the sending node, destination node and the requested rate for each stream, each of the nodes has a location *(x,y)* which may change over time in the surface.

A direct communication between two nodes in this network is possible if their distance is smaller than the transmission range for these nodes. An indirect (multi-hop) communication between two nodes is possible if there exists a path of nodes with direct communication between the source and destination.

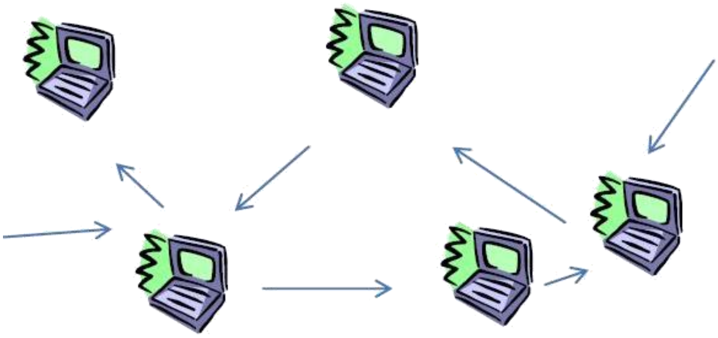


Fig : wireless network model

Here we are transmitting a video over wireless network from one node (source) to destination. In a wireless network there are many nodes connected with each other via router. Here we implemented this structure for streaming a video to one node to another node.

b) Impacts and usages of video streaming

In just the last few years, video application usage has increased significantly and is comprising a greater portion of Internet traffic every day. Rich-media-hungry smartphones and tablets have flooded the marketplace and more applications that drive video usage continue to arrive on the market. Today's wireless networks must be designed from the start with the capacity to handle bandwidth-eating video traffic.

c) Make video seamlessly stream over wireless

- Providing high-quality video over wireless poses challenges above and beyond sheer bandwidth requirements. For starters, video traffic has very low tolerance for packet loss in the transport network from video server to video client. High or variable latency can also cause issues for streaming video applications. Wireless networks must take these factors into consideration during the design phase.

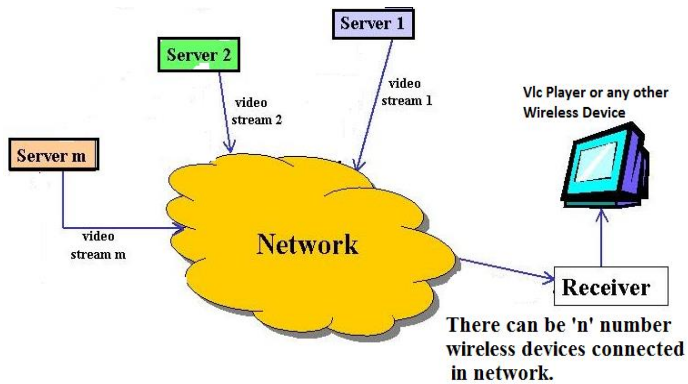
- Video over wireless becomes even more challenging in high-density, high-usage scenarios such as classrooms or training rooms where dozens of users may be simultaneously accessing a single video source. Worst-case scenarios must be considered when designing wireless networks that will be used for such applications.

- In order to provide a high-quality video streaming experience for users over a wide range of applications, wireless networks must provide the following essential elements:

  1. Sufficient wireless signal
  2. Sufficient wireless bandwidth
  3. Quality of service (QoS)
  4. Multicast optimization

3. System Architecture

This is a simple architecture of a video streaming over wireless network. Here I used this system architecture to streaming purpose. I transmit a video over wireless network using multi hop technique. This hoping occurs respect to bandwidth. In this project I used to work with transport layer as well as application layer.

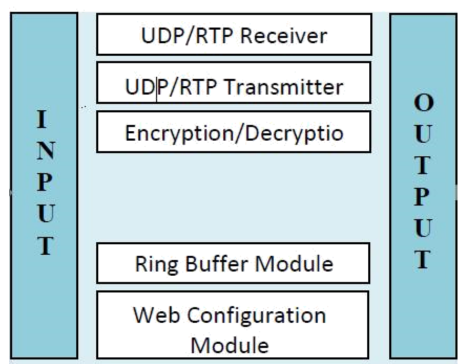


a) Video network protocols

There are many limitation of the TCP & UDP in video transmission So to overcome with this problem, a new transport level protocol, called Real Time Transport Protocol (RTP), was specified within the Internet Engineering Task Force (IETF).

b) Real Time Transport Protocol

RTP is a real-time end-to-end transport protocol. However, considering RTP as a transport protocol may be misleading because it is mostly used upon UDP, which is also considered as a transport protocol. RTP is best viewed as a framework that applications can use to implement a new single protocol. RTP doesn't guarantee timely delivery of packets, nor does it keep the packets in sequence. RTP gives the responsibility for recovering lost segments and
Re-Sequencing of the packets for the application layer. There are a couple of benefits using RTP.
Which are required by most multimedia applications.
Here I have used user datagram protocol (UDP) which is very important in terms of my project according to the paper which I have chosen for this project.

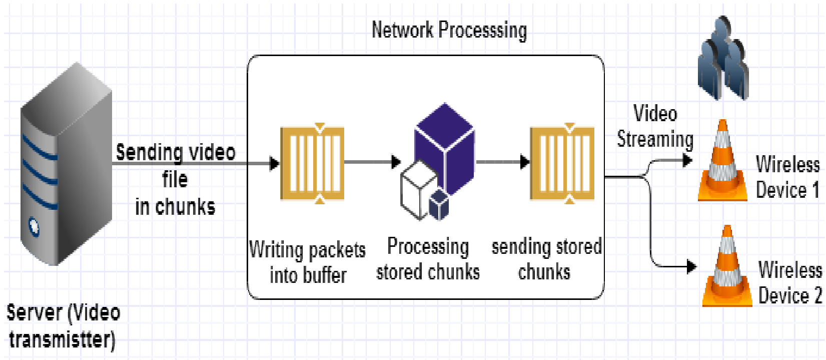


c) Datagram (SOCK_DGRAM)

In Internet Protocol terminology, the basic unit of data transfer is a **datagram**. The socket sends datagrams as independent packets with no guarantee of delivery. You can experience or receive lose or duplicate data. Datagrams can arrive out of order. The size of the datagram is limited to the data size that you can send in a single transaction. You can issue a **connect()** function on this type of socket. However, on the **connect()** function, you must specify the destination address that the program sends to and receives from.

4. Project Implementation and Approach

First I Implemented stream based network to figure out streaming concept. In this I implemented binary bytes streaming over wireless mesh network in multi hop fashion. Once successfully implemented this feature I came to know how wireless network behaves. After that I provided video over wireless network and successfully received on system. Here is result of video streaming over wireless network.



However there can be packet loss since this depends on the bit rate of the video and the size of the video, so to overcome this problem we can accordingly adjust the **"threads" sleep time.**

For the video show below the threads sleep time is adjusted "14900" microseconds but due to network problem and wireless congestion there can be the problem of packet loss.

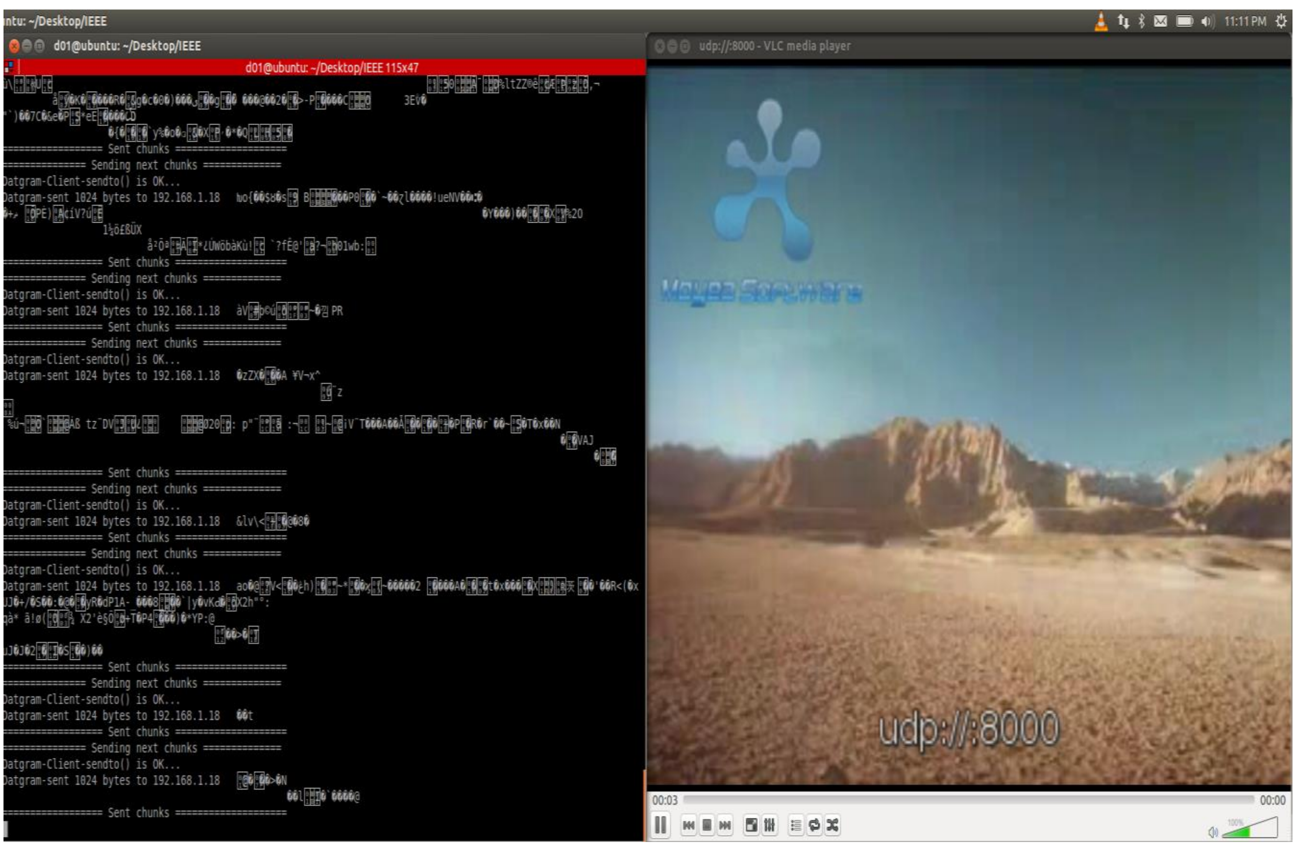


Fig: Implemented result.

I adopted MPEG-2 technique for video transmission purpose. According to me or my project prospective MPEG-2 is a one type of container which contains my video and I used TCP/RTP transport protocol for video streaming. Brief about MPEG-2 is described below.

a) MPEG2-TS

- MPEG transport stream (MPEG-TS, MTS or TS) is a standard format for transmission and storage of audio, video.
- It is used in broadcast systems such as DVB, ATSC and IPTV.
- Transport stream specifies a container format encapsulating packetized elementary streams, with error correction and stream synchronization features for maintaining transmission integrity when the signal is degraded.
- MPEG-2 Transport Streams are composed of 188 byte TS Packets, each with a 4 byte header.

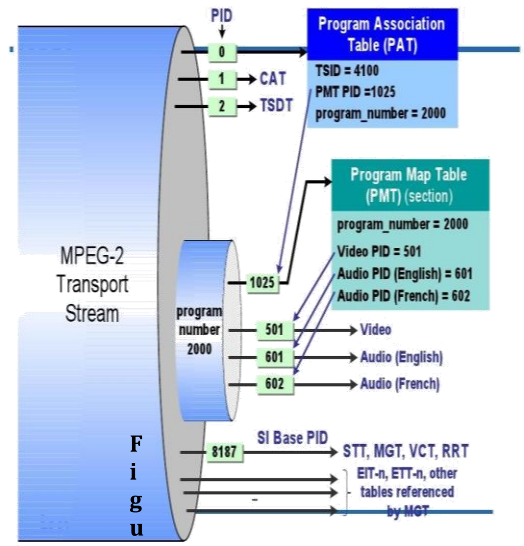


Figure MPEG2-TS Structure

As shown in figure MPEG2-TS stream contains Elementary stream and also have single either Audio or Video file or both and also have subtitles.

Why use MPEG-2 Transport Stream encapsulated using the RTP protocol?

If we use the MPEG-2 transport stream then we believe the addition of the RTP header provides significant benefit over simple subdivision into UDP datagrams. This is because the RTP header makes the stream more network friendly and allows it to be more easily modified over the transmission path.

1. RTP allows sequence reordering of packets.
2. RTP is friendlier to a firewall
3. Error Recovery

b) Hardware and software requirements/used :

i) VLC Player : to receive video from server as a wireless device ii) GCC compiler : to compile C Program.

5. Analysis and Observation

Robustness:

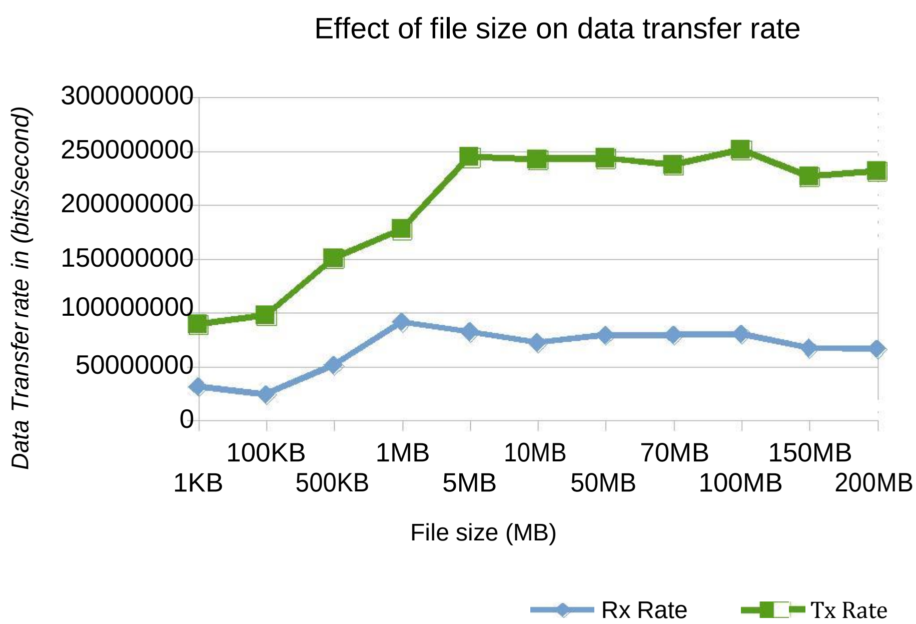
When we want to establish a wireless environment on network simulator for video streaming purpose we have to make sure about packet size as well as network complexity. When we designed a code relatively and implement successfully than we can say that data of this size can be transmitted over wireless network. But when you extend the size of video or increase the number of packets you have to change your code. Here I implemented robust flow for this project. In this project we can transmit different size of the videos easily. So we can say that this is best solution for video streaming.

Evaluation of the performance:

To check the performance of the video streaming over muti-hope/Single-Hop wireless, I have performed several experiment using different video file sizes and calculated their throughputs (in bit/second) using networking tools like iperf. This evaluation of the performance have been done on local LAN using 3-4 systems as well as the system on which the program as the server (transmitter) is running. While evaluating this project we can across many factors which needed to be taken into consideration and which defines what motivation is required in order to get video transmission without packet loss. The experiment clearly shows the reason about the degradation or improvement of the video transmission in the network.

Experiment 1: Data rate vs Video File Size

Observation:



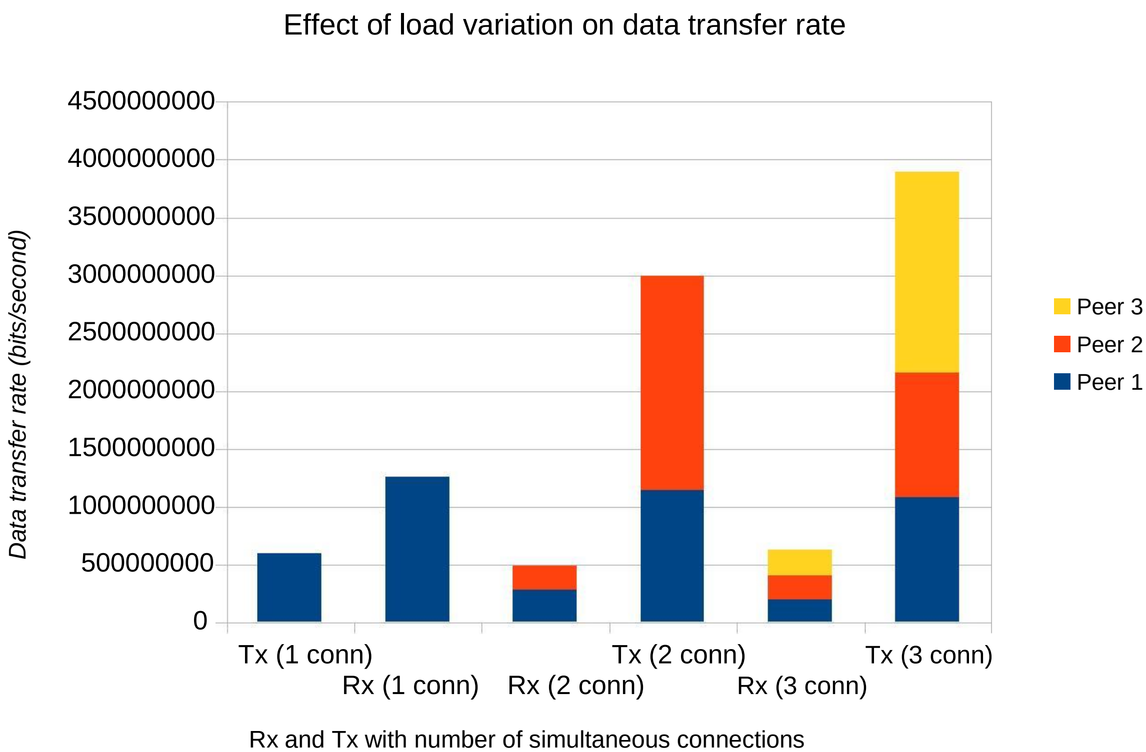
I, observe that as the size of the file increases, the rate of transfer increases and then the transfer rate more or less stabilizes, with very little fluctuations.

Analysis:

- The observation can be largely attributed to disk I/O performance.
- Since the packet size is constant, the effect of network latency or bandwidth is minimal on the data transfer rate.
- The overall rate can be largely affected by the disk performance.
- Most, if not all, disks uses a small cache called disk buffer, to speed up the disk I/O performance.
- When block of data is written to the disk, it is actually stored in the disk buffer and the disk signals that the write is complete immediately after receiving the data. The data is actually written to the disk later.
- This reduces the I/O latency in the process doing the write.
- Similarly the disk may do some read-ahead and saves the data on the disk buffer, as a result the process requesting the data may receive it without much delay.
- However, when the amount of data to be written/read increases, the disk buffer may become full. When the amount of data is more than the disk buffer size, the process no longer gets benefited from the buffer, and the disk performance becomes the bottle neck.
- As a result of the above, I see that the transfer rate keeps increasing. At a certain point, when I hit the disk buffer limit, the transfer rate stops increasing as disk performance becomes a bottleneck.

Experiment 2: Data Rate vs Load

Observation:



1. This test was performed by clients (1, 2, 3) receiving files simultaneously from server in each iteration respectively.
2. I, observe that for the client receiving the files simultaneously, the data transfer rate on individual connection decreases as the load increase, but the cumulative transfer rate remains more or less the same.
3. For the servers sending the files, the transfer rate remains almost same.

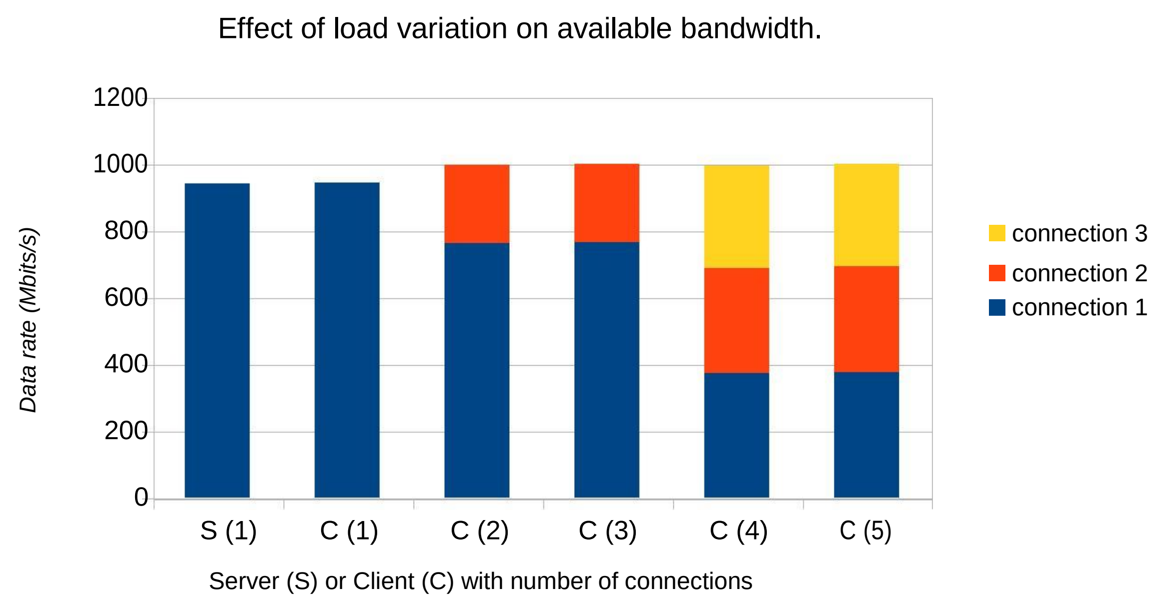
Analysis:

1 This results are expected.
2 In the client receiving the files, both the network performance and the disk I/O is divided among the connections. The resources being finite, the overall performance remains constant but gets divided among the various connections as the load increases.
3 For the server sending the files, irrespective of how many connections the receiving client is part of, tries to send as much as possible and as fast as possible, hence the overall transfer rate remains constant.
4 I, would have expected the rate to go down a little because the receiving client must be

busy serving other connections and may delay the ACK. But probably because the performance bottleneck here is the disk I/O and not the network bandwidth, hence the network delay cause by the receiver would not impact the transmitter's performance.

Experiment 2: Bandwidth calculation

Observation:



a) Although I would have expected that the results would be similar to that of the previous test, the results are slightly different. This can be depend on the network

b) On the server side I see that the results are almost identical to that of the previous test – the individual throughput of each connection decreases but the cumulative throughput remains almost same.

c) On the client side I see the same – the individual throughput of each client reduces as the load increases while the sum of the bandwidth of all the clients remains the same. This is very different from what I observed in the previous test.

d) Another observation is that the maximum available bandwidth calculated by this test is more than achieved by the previous test.

Analysis:

a) The major difference, and the cause of the different result, between this test and the previous one is that this test measures ONLY the network bandwidth.

b) At the server, the total available bandwidth is divided in processing the simultaneous connections. So I see that individual throughput of each connection reduces but the overall throughput remains almost same.

c) Because the server is busy in processing all the connected clients simultaneously.

d) At the clients end, even though they would try to receive as fast as possible, they couldn't because of the server is sending the data faster than the receiver can receive it and due to which some of the packets are been lost. So the server's processing power becomes the bottleneck.

e) I also observed that the maximum throughput achieved by this test is more than that of the previous test, this is mainly because the tool which is used that can only measures network bandwidth and disk I/O is not involved, this leaves the process more time to send and receive data over the network, hence more throughput.

Overall Performance :

The below graph shows the overall performance if the video streaming over network.

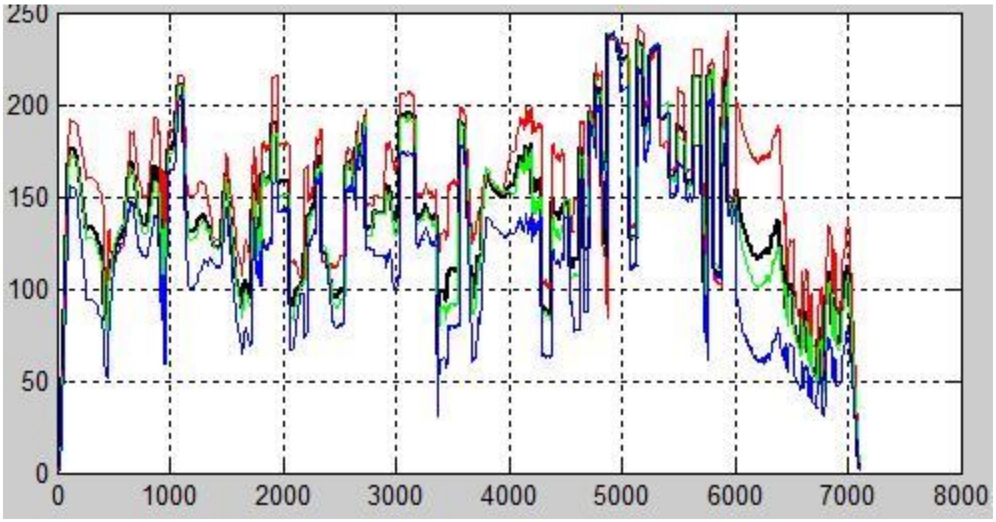


Fig: video streaming graph

6. Screenshots and Project configuration

Step : 1

The make have been made to compile the project successfully so that the server can be started.

Step 2 :

This screen is from VLC player which is our wireless device and it is retrieving all the data which server is sending but in order to get the data and successfully play into VLC we needto connect it with our server, in order to do that we need to

Go to the menu of the VLC -> Select "Open Network Stream" -> type "udp://@:8000"

8000 is the port number on which your server listening and sending the data you need to connect on this port, similarly we can connect any number of VLC with our server and those connected player can receive the packets from the server.

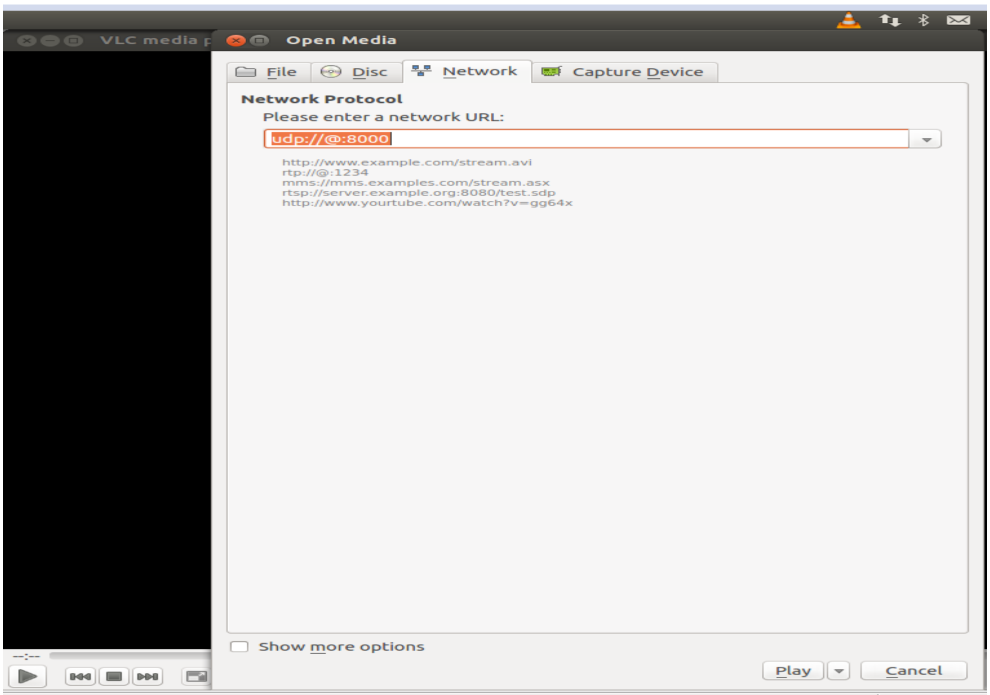


Fig -1 : VLC configuration

<u>Step 3 :</u>

This is the help give to the user if somehow any wring arguments will be passed then this help message will be displayed so that user can check his/her typed command in order to run this project.

The details of all commands have been mentioned in the message box.

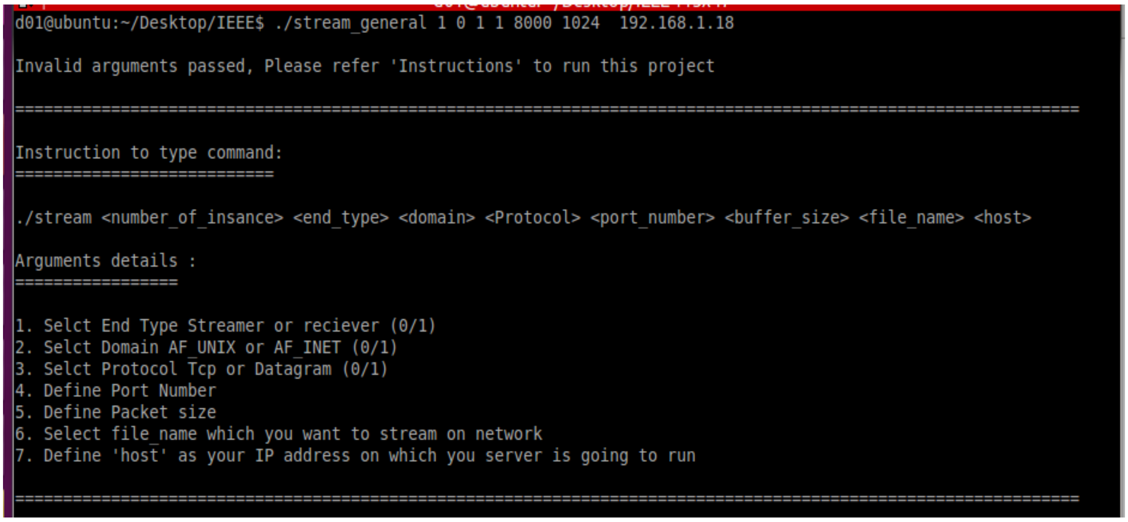


Fig 2 : Error(Help) Message

<u>Step 4 :</u>

Once all the command have been cross checked then user can run the server with the particular file name which he wants to stream but before he starts the server he should make sure that VLC UDP sockets are open else the packet will be lost whatever server will sent.

<u>Step 5 :</u>

Now it's time to start the server/process, the below image shows the server process, as you can see the server is sending 1024 bytes of the data in every 1sec because this size is given by user. Also the encrypted data which have been show it is for the video chunks which are been transmitted. This is the binary format of the data which is getting into the process.

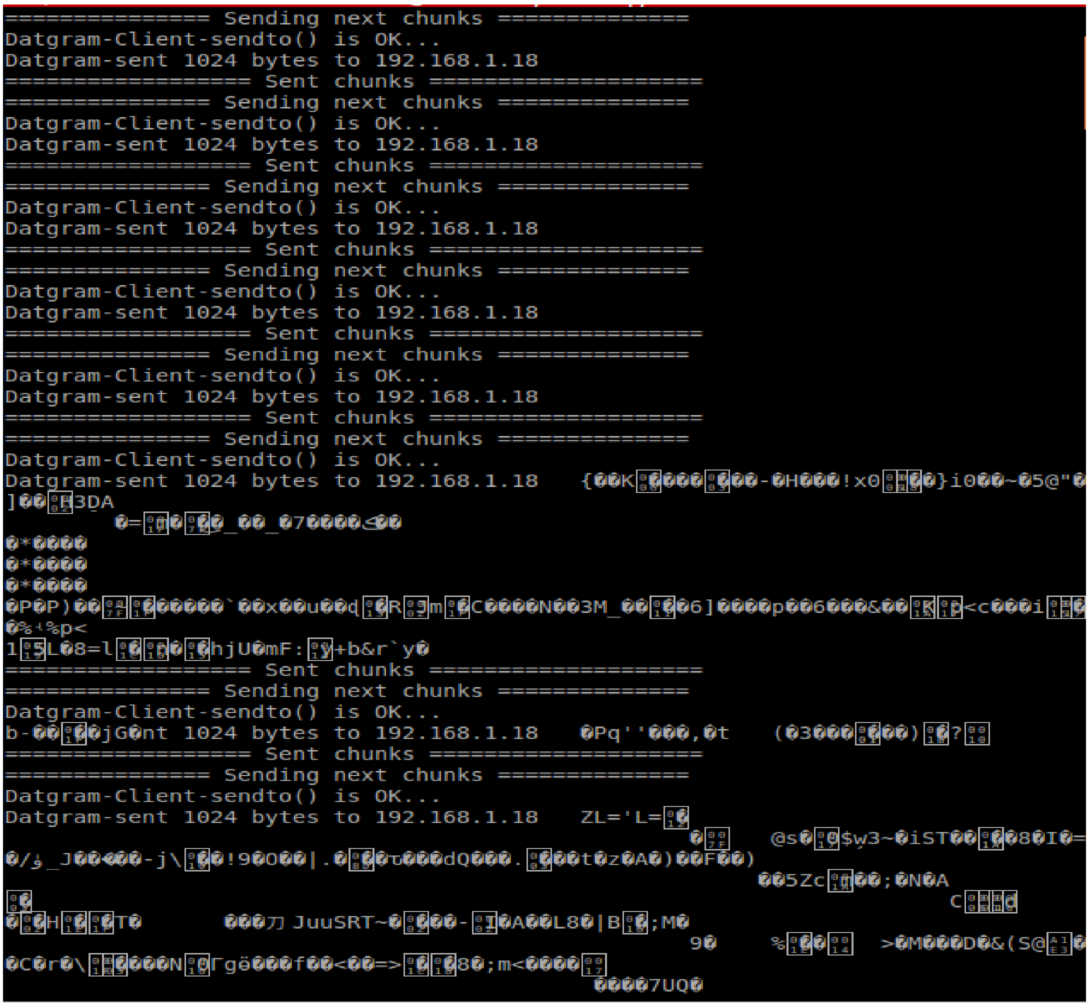


Fig : 3 Server Start Process

7. Summary and Conclusion

I successfully implemented video streaming and I got desired output
Due to this project I understand the behavior of network and how to deal with it. This project based on bandwidth related hoping so I clearly understand that high performance video required high bandwidth for streaming purpose otherwise video streaming has to face noise and delay. This factors affecting on performance of video streaming. In my implemented project I got expellant performance of video streaming. There is no packet loss or delay occurs and if packet loss occurred then it must be because of the network interference and connection with the client. So according to me it is a real time video streaming over wireless network.

8. References

[1] Performance Evaluation of Video Streaming in Multihop Wireless Mesh Networks - Xiaolin Cheng , Prasant Mohapatra , Sung-Ju Leez, Sujata Banerje.

[2] “The Network Simulator–ns-2.” [Online]. Available: http://www.isi.edu/ nsnam/ns/.

[3] D. Li and J. Pan, “Evaluating MPEG-4 AVC video streaming over IEEE 802.11 wireless distribution system,” in Proc. 9th IEEE Wireless Communications and Networking Conference (WCNC’08), Las Vegas, NV, USA, March 2008.

[4] N. Cranley and M. Davis, “Performance evaluation of video streaming with background traffic over IEEE 802.11 WLAN networks,” in Proc. 1st ACM Workshop on Wireless Multimedia Networking and Performance Modeling, pp. 131–139, 2005.

[5] http://unixhelp.ed.ac.uk/CGI/man-cgi?inet_ntoa

[6]http://www.beej.us/guide/bgnet/output/html/multipage/zindex.html

[7] http://linux.die.net/man/3/getaddrinfo

[8]http://msdn.microsoft.com/enus/library/windows/desktop/ms738564%28v=vs.85%29.aspx

[9] http://pubs.opengroup.org/onlinepubs/009695399/functions/inet_addr.html

[10]http://www.gta.ufrj.br/ensino/eel878/sockets/inet_ntoaman.html