Performance Evaluation of Multicast Video Streaming via P2P Networking

Rajesh Bhadada Birendra Rai

Department of Electronics& Communication Engineering,
M.B.M. Engineering College, J.N.V.University, Jodhpur

ABSTRACT
In the past few years, the multicast video streaming via P2P networking has become a very successful and increasingly popular in distributing multimedia content by encouraging an important number of users to act as both clients and servers. Video on demand (VoD) streaming one such service where videos are delivered to asynchronous users with minimum delay and free interconnectivity. The VoD is costly due to the limited upload capacity of the video server and traditional centralized client/server architecture. Peer-to-Peer streaming techniques are an approach to alleviate server load through peer-assisted sharing. Proxy caching is a key technique to reduce transmission cost for on-demand multimedia streaming. This innovative approach combines the advantages of both proxy caching and peer-to-peer client communications. In this paper, firstly we provide a better understanding of the basic concept of multicast video streaming, media delivery structure, streaming media storage size, bandwidth requirement, video streaming architecture and then present a novel approach for the performance evaluation of multicast video streaming via P2P networking using servers and proxy servers situated between local area networks (LANs). The results demonstrate the benefits of multicast video streaming on the server-proxy paths.

Keywords
Peer-to-Peer (P2P) networking, Proxy caching, Video Streaming, Local Area Network (LAN), Multicast, Video on demand (VoD)

1. INTRODUCTION
The multicast video streaming via peer-to-peer (P2P) networking architectures have rapidly gain the popularity, where clients store the media data after the streaming service, and act as supplying peers by streaming the media data to other requesting clients (peers) in the near future[1,2,4,6]. P2P networks overcome the setback of bottleneck around centralized server due to its distributed design and architecture. The main aim of this paper is to understand the concept of multicast video streaming and study the benefits of using multicast instead of unicast on the server-proxy paths. Proxy caching is an important technique to reduce transmission cost for multimedia streaming. A technique to determine the optimal cache allocation at each proxy server among different videos, and an efficient proxy-assisted transmission scheme when server-client paths are multicast capable are being studied through this paper [9, 10].
The rest of this paper is organized as follows. Section 2 explains the media delivery structure, the requirement of streaming bandwidth and storage space in unicast/multicast and live streaming environments and explanation of P2P network architecture for single/multiple source models. Section 3 describes the overview of the system and its model. Section 4 presents optimal proxy cache allocation technique. The performance evaluations are presented in Section 5, in which the effect of proxy caching on the transmission cost and benefits of using multicast from server to proxy are being observed**.** Finally Section 6 summarises conclusions.

2. MEDIA DELIVERY
The media delivery structure is depicted in Figure1. Media data can be stored and delivered to a client either from the local file system or from a remote server across the network by using streaming techniques [2,3]. Data from live continuous media such as live web cams are also streamed from server to client. A client may be a personal computer or a smaller device such as a handheld computer, personal digital assistant (PDA) or cellular telephone.

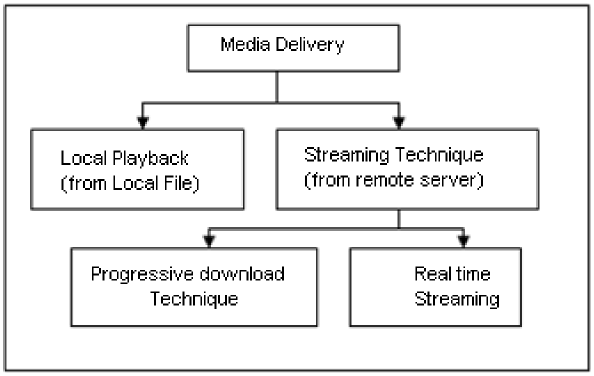

Fig.1. Media Delivery Structure Diagram

In this paper, we consider stored video streaming and its real-time transmission. On-demand delivery of video (stored) in the real-time transmission is the prime part of multimedia. There are two techniques for transmission of stored video namely the progressive download technique and real time streaming technique via P2P networking.

(i) Progressive download technique: In this technique a media file containing audio or video is down loaded and stored on the clients local file system. As the file being downloaded, the client can play back the media file without wait for the file to be down loaded completely in its entirety. This technique is more useful for relatively small media files such as short video clips.

(ii) Real time streaming: When the media file is streamed to the client but is only played and not stored by the client, it is called real time streaming. As the media file is not stored on the client system, real time streaming is preferred over the progressive download for large media files. It eliminates need for storage on the user system, like full video Internet radio and TV broadcasts.

2.1 Streaming bandwidth and storage space

The storage space for streaming media is calculated from the streaming bandwidth and length (in seconds) of the media by using the formula as given:

Number of MBytes transferred =

$$\frac{\text{Length (in seconds)} \times \text{Bit rate (in bit / seconds)}}{(8 \times 1024 \times 1024)}$$

Example: 1hour of video encoded at 500 Kbit/s requires $\frac{(3,600 \times 500,000)}{(8 \times 1024 \times 1024)}$ i.e. approximately 215MBytes of storage.

If the file is stored on a server for on-demand streaming and this stream is viewed by 200 people at the same time, then the bandwidth requirement by using unicast, multicast and live streaming environments will be determined as follows:

(i) Unicast Streaming: It requires multiple connections from the same streaming server even when it streams the same content as depicted in Figure 2. Unicast protocols send a separate copy of the media stream from the server to each recipient [1, 8]. Unicast is the norm for most internet connections, but does not scale well when many users want to view the same program concurrently using a unicast protocol, the bandwidth requirement is: 500 Kbit/s × 200 = 100,000 Kbit/s = 100 Mbit/s of bandwidth. This is equivalent to around 45 GByte/ hour.

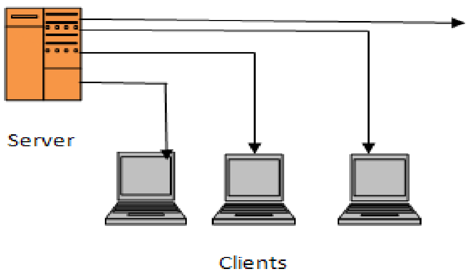

Fig.2. Unicast Streaming

(ii) Multicast Streaming: Multicasting broadcasts the same copy of the multimedia over the entire network to a group of clients as depicted in Figure 3. Using a multicast protocol the server sends out only a single stream that is common to all users. Hence, such a stream would only use 500 Kbit/s of serving bandwidth. Multicast protocols are developed to
reduce the data replication (and consequent server/network loads) that occurs when many recipients receive unicast content streams independently. These protocols send a single stream from the source to a group of recipients

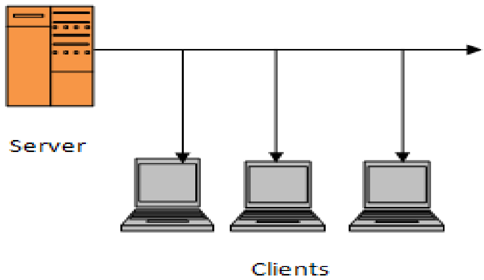

Fig.3. Multicast Streaming

(iii) Live streaming: The calculation for live streaming is similar. Let us assume that the speed of the encoder be 1000 Kbit/s. If the show duration last for 3 hours, with 200 viewers then the bandwidth requirement is calculated as follows :

$$\frac{\text{Encoder speed (in Kbps)} \times \text{No. of seconds} \times \text{No. of viewers}}{(8 \times 1024)}$$

$$\frac{1000\ (\text{Kbps}) \times 3 \times 3600\ (= 3\ \text{hours}) \times 200\ (\text{No. of viewers})}{(8 \times 1024)}$$

= 264 GBytes approximately.

2.2 Peer-to-peer (P2P) systems:

A peer is considered to be a provider and a consumer of resources along with serving capacity into the system. These systems significantly reduce the load in the server. Unlike centralized server- client based schemes, P2P systems are distributed systems consisting of peers interconnected with each other, able to self organize into network topologies in order to share resources like content, bandwidth, CPU cycles and storage. The P2P network streaming architecture refers to the manner used for multimedia content transfer and the entities which are involved during the streaming mechanism [8,15]. A given peer can play three different roles in the perspective of P2P streaming:

- Source: Peer containing the media contents and intended to share with other peers. Peer can store whole or a part of a given content
- Intermediate: An intermediate peer, receive a given content and then transmit it to the next intermediate peer. Intermediate peer serves as a transport node to facilitate the streaming mechanism
- Destination: It is the client who requests for the content. Client peer can obtained media contents from one or more sender peers depending on the architecture.

We define two kinds of network architectures namely single source and multiple source

(i) Single source: In this case, the multimedia content is stored into only one source peer in the network [8]. The content peer starts transmitting the content to all client peers requesting for it and the intermediate nodes can play a more important role. Figure 4 gives an example of single source
streaming towards two peer clients

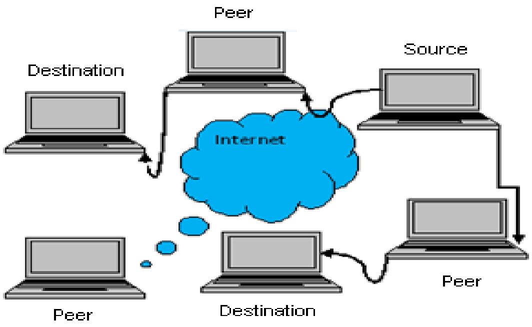

Fig.4. Single-Source P2P streaming model

(ii) Multiple sources: In this case each client peer receives packets of a multimedia content from multiple sender peers while each sender peer can send packets to one or multiple client peers.

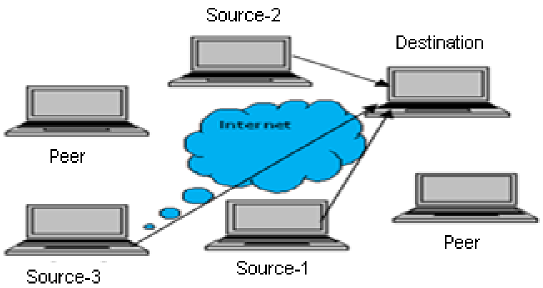

Fig.5. Multi-Source P2P streaming model

The contents can be retrieved from several peers into the network simultaneously as depicted in Figure 5. The role of intermediate peer is limited to the transfer of the received packet towards the destination peer.

3. OVERVIEW OF THE SYSTEM

Architecture of streaming video in the internet as shown in figure 6 has been considered, where a server and a set of proxies are connected [9, 13, 14, 15].

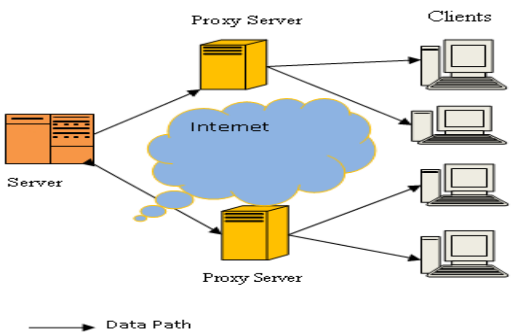

Fig.6: Streaming video in the Internet

The each proxy server is held responsible for multiple clients and they are closely located with relatively low communications costs. The server is located in remote from clients, and the communications among them incur much higher costs. We consider multicast is available from server to proxy and vice versa. We assume that clients always request playback from the beginning of a video. The proxy server intercepts the clients request and, if a prefix of the video is present locally, streams the prefix directly to the clients. If the video is not stored in its entirety at the proxy server, then makes the contact with the remotely located server for the suffix of the stream. The server multicasts the required suffix to the proxy servers, and the proxy servers multicasts the incoming data to the clients that request the video.

System Model: To provide a formal model of the system, Table 1 introduces the notations and the main concepts that will be used in this paper. Superscript and subscript have been used to represent the index of the proxy and the video [5,7,9,10,11].

TABLE 1: Notations used

Parameter	Definition
N	Number of videos
V_i	Length of video i (units)
P	Number of proxies
μ	Caching grain (smallest unit of cache allocation)
n_i	Space of video i (units)
b_i	Mean bandwidth of video i (bits per sec.)
c_s	Transmission cost from server to proxy (per bit)
c_p	Transmission cost from proxy to its client (per bit)
β	The multicast scaling factor
A_i^p	Access probability of video i at proxy P
λ_i^p	Request rate for video i at proxy P
λ^p	Aggregate request arrival rate for videos at proxy P
λ_i	Aggregate request arrival rate for video i at all proxies
S_p	The cache space (units) of proxy P
v_i^p	Length (sec) of cached prefix for video i at proxy P
$\vec{v}_i$	Storage vector of videos i, $\vec{v}_i = (v_i^1, v_i^2, v_i^3, \ldots, v_i^p)$
$C_i(\vec{v}_i)$	Transmission cost per unit time for video i when the storage vector for video i is $\vec{v}_i$

A server with a repository of N Constant-Bit-Rate (CBR) videos and p proxies has been considered. Let A_i^p be the access probability and λ_i^p be the access rate of video i at proxy P respectively. It is assumed that the access probability of all video and the aggregate access ratet to the video repository are known a priori. Hence $\sum_{i=1}^{N} A_i^p = 1$ and let λ_i be the aggregate request arrival rate for video i at all the proxy server, then $\lambda_i = \sum_{p=1}^{P} \lambda_i^p$. Let λ_i^p be the aggregate access rate to the video repository at proxy P then $\lambda_i^p = \lambda^p A_i^p$. It is also assumed that caching grain of size u to be the smallest unit of cache allocation and all allocations are in multiples of this unit. The caching grain can be one bit or one minute of data, etc. We express the space of video i and the cache space at each proxy server in multiples as the caching grain. Video i has playback bandwidth b_i bps, length V_i seconds, and space n_i units, u $n_i = b_i V_i$. Proxy p can store S_p units where $S_p \leq \sum_{i=1}^{N} n_i$.

The storage vector $\vec{v_i} = (v_i^1, v_i^2, v_i^3, \ldots, v_i^p)$ specifies that a prefix of length v_i^p seconds for each video i is cached at proxy P, I = 1,2,3,……., N . It may be noted that the videos cached at the proxy server cannot exceed the storage limitation of the proxy server, that is $\sum_{i=1}^{N} b_i v_i^p \leq \mu S_p$. For simplicity of exposition, Network propagation latency is ignored by us. On receiving a client request for a video, the proxy server determines a transmission schedule that depends on the transmission scheme in use. This transmission schedule specifies, for each video frame, when and on what transmission channel (unicast or multicast connection) it will be transmitted by the proxy server. The proxy server also determines and requests the suffix from the server.

A reception schedule is transmitted from the proxy server to the client. It specifies, for each frame in the video, when and from which transmission channel the client should receive that frame. Note that a client may need to receive data from multiple transmission channels simultaneously. Frames received ahead of their playback times are stored in a client-side work ahead buffer. For simplicity, we shall assume the client has sufficient buffer space to accommodate an entire video clip. Let c_s and c_p be the cost associated with transmitting one bit of video data from server to proxy server and proxy server to its client respectively using unicast.

To minimize the mean transmission cost per unit time is aggregated over all videos in the repository, i.e. $\sum_{i=1}^{N} C_i(\vec{v_i})$ where $C_i(\vec{v_i})$ is the transmission cost per unit time for video i when a prefix of length v_i of the video is cached at the proxy server. When a proxy server and its clients are located in a local area network (LAN) environment, the bandwidth required to send one bit from the proxy to multiple clients using multicast is still one bit. Therefore, the transmission cost to send one bit from the proxy to multiple clients is still from server to proxy server paths, we assume the cost to transmit a bit of data from the server to m proxies using multicast is $\beta m c_S$ where $\beta \in (\frac{1}{m}, 1)$ and is referred to as the multicast scaling facto*r*. The minimum value for β is $\frac{1}{m}$ in which case the cost of transmitting a bit of data from the server to m proxies is c_s same as in a LAN environment. The cost when using multicast in a wide area network depends on a variety of factors including the multicast tree topology and the size of the multicast group. We make it a note that when c_p = 0 and $\beta m c_S$ =1 the transmission cost reduces to be the server bandwidth usage. When c_p =1 and c_s = 0, the transmission cost reduces to be the amount of outgoing traffic at the proxy server.

4. OPTIMAL PROXY CACHE ALLOCATION

A general technique to determine the optimal proxy prefix cache allocation for any given proxy assisted transmission scheme has been proposed.

Let $A_i = m_i \{0 \leq m_i \leq n_i\}$ denotes the set of possible prefixes for video i, where m_i units is the space and $\frac{m_i \mu}{b_i}$ seconds is the length of a possible prefix of video. For a given transmission scheme, the average transmission cost per unit of time for video i, $C_i(\vec{v_i})$ is a function of the prefix V_i cached at the proxy, $0 \leq \vec{v_i} \leq V_i$. We define saving $(\vec{m_i})$, where $\vec{m_i} = (m_i^1, m_i^2, m_i^3, \ldots m_i^P)$, to be the saving in the transmission cost when storing m_i^P units of prefix of video i at proxy server P over the cost when video i is not stored at the proxy server, p = 1, 2, 3 ….. P. Our main aim is to maximize the aggregate savings and, hence, minimize the aggregate transmission cost over all the video. The optimization problem can therefore be formulated as maximize: $\sum_{i=1}^{N} \text{saving}(\vec{m_i})$, and

s.t. $\sum_{i=1}^{N} (\vec{m_i}) \leq S_P$,

$m_i^P \in A_i$, ($1 \leq i \leq N$}, $1 \leq p \leq P$.

Let B be a two-dimensional matrix, where entry B(i, $\vec{j}$), where $\vec{j} = (j_1, j_2, \ldots, j_P)$ represents the maximum saving in the transmission cost when using videos i and j_P units of storage are allocated at proxy server p, ($0 \leq i \leq N$) and j ($0 \leq j \leq S_P$) units of the proxy cache.

$$B(i, \vec{j}) = \begin{cases} 0, & i = 0 \\ \text{Max}\{B(i-1, \vec{j}), B'(i, \vec{j})\}, & i > 0 \end{cases}$$

Where, $B'(i, \vec{j}) = \max\{B(i-1, \vec{j} - \vec{m_i}) + \text{saving}(\vec{m_i})\}$.This matrix is filled in row-order starting from B (0, $\vec{j}$), $\vec{j}$ = 0; …, S_P The value $B(N, S_p)$ is the maximum saving in transmission cost when all N videos have been used. The minimum transmission cost is $\sum_{i=1}^{N} C_i(0) - B(N - S_P)$

Since the saving is relative to storing nothing at the proxy. The optimal cache allocation can now be computed as follows. For each entry, we store a pointer to an entry from which this current entry is computed. By tracing back the pointers from the entry $B(N, S_p)$ the optimal allocation is obtained.

5. PERFORMANCE EVALUATION

The performance of multicast video Streaming via P2P Networking is evaluated in this section. The main point is incurred transmission cost reduction by introducing proxies and client's co-operation, under assumptions as follows. The video storage space in the server contains N=100 CBR (constant Bit Rate) videos clips each of 512 Kbps rate. Their lengths are uniformly distributed in between 100 and 140 minutes; the mean (120 minutes) is a typical length of a movie. We assume the access probabilities of the videos follow a Zipf distribution with skew factor θ = 0.271. The cache grain (unit) is set to the size of one minute video data. All the cache spaces discussed in this section are normalized by the total space of the video repository, and the transmission costs are normalized by the corresponding cost of a system with no cache. The transmission cost is normalized by both the video bandwidth and the value of cs. That is, the normalized transmission cost i $\sum_{i=1}^{N} C_i(\vec{v_i}) / (c_s b_i)$

Let $\hat{c}_p = c_p/c_s$. In this section, we assume $\hat{c}_p \in (0, 1)$. It is observed that $\hat{c}_p=0$ corresponds to $c_p=0$ and $\hat{c}_p=1$ corresponds $c_p = c_s$ .We represent the proxy cache space as a percentage r, represents space of the video repository.

Homogeneous proxy servers have been considered that is, the configurations for all the proxy servers are the same, including proxy cache space, video access probabilities, aggregate arrival rate,67 c_s and c_p, etc. We assume that the aggregate arrival rate ranges from 10 to 500 requests per minute. The total number of proxies P = 10 or 100. The multicast scaling factor ranges in $\beta \in (\frac{1}{P}, 1)$. The performance trends under P=10 and P=100 are similar, hence in this paper P=100 has been assumed.

(i) The effect of proxy caching on the transmission cost:
We observe that the proxy server caching leads to lower network transmission cost in all the settings. This is expected since data from the server pass the proxy server and hence transmitting a stream directly from a proxy incurs less cost than from the server. We define the relative reduction under optimal proxy cache allocation over no proxy caching to be the difference in the costs under these two settings divided by the cost without proxy caching. Figure **7** plots the relative reduction thus defined when the aggregate arrival rate to a proxy is 50 per minute and β ranges from 0.2 to 0.8. It is observed that a relatively small proxy cache (1%-10% of the video repository) is sufficient to realize substantial savings in transmission cost and the proxy cache size has a diminishing effect on the cost savings.

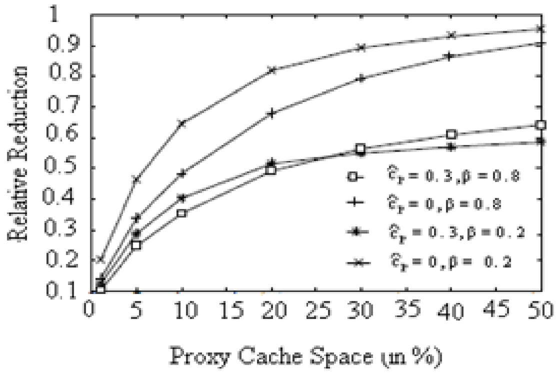

Fig.7. Relative reduction of transmission cost over no prefix Caching

Furthermore, the reduction is more impressive for $\hat{c}_p = 0$ and $\hat{c}_p=$ 0.3.This is because, for lower values of the savings from transmitting directly from the proxy cache to the clients is more impressive. Finally, we observe similar characteristics for other request arrival rates.

(ii)Benefits of using multicast from server to Proxy server path: Figure 8 depicts the transmission cost when using multicast and unicast on the server to proxy server paths as a function of the proxy server cache space for λ^P =50/minute and β = 0.4. Using multicast on the server to proxy server paths leads to a significant saving of transmission cost for small and medium proxy server cache space only. We observe that by using multicast the transmission cost reduces by 32% and 19% over that using unicast for $\hat{c}_p= 0$ and $\hat{c}_p = 0.3$ respectively and unicast architecture, when using β= 0.4 & λ^P =50/minute

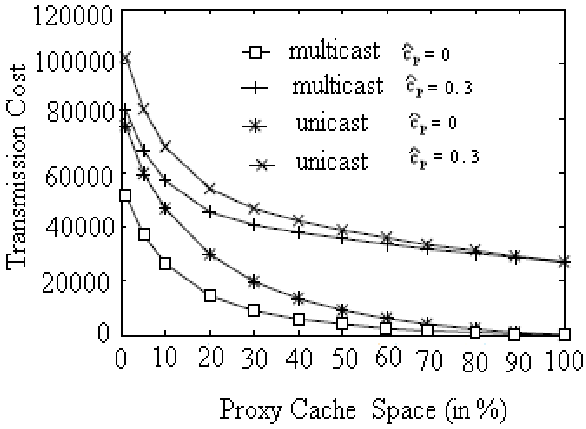

Fig.8. Comparison of the transmission cost between multicast

As the proxy cache increases, more contents are transmitted from the proxy servers cache directly and the cost on the server-proxy paths becomes less dominant in the total cost. That explains why transmission costs using multicast and unicast on the server to proxy server paths become close for large proxy cache spaces. We observe that the cost savings from using multicast on server-proxy paths are more significant for small values of $\hat{c}_p$. This is because, in that case, the cost on the server to proxy server paths is more dominant in the total cost and the benefits of using multicast are more obvious.

6. CONCLUSION

Review of the basic concept of media delivery structure, storage space and bandwidth requirement of unicast/multicast/live streaming environment, P2P streaming model as single source and multiple source and proxy server-client catching system that combines the best features of proxy caching and peer-to-peer networking has been carried. Techniques for distributing video streaming among the proxy server and client are surveyed. The performance has been evaluated under various system configurations. The main findings can be summarized as follows (i) proxy server based architecture significantly reduces the transmission cost for on demand video streaming (ii) for the same proxy size, use of prefix caching for a set of videos result in significantly lower transmission costs compared to entire-object caching policies (iii) under optimal prefix caching, a relatively small proxy cache (1%-10% of the video repository) is sufficient to realize substantial savings in transmission cost (iv)The allocation under optimal prefix caching is sensitive to the transmission scheme, the aggregate arrival rate and the value of $\hat{c}_p$. The benefits of using multicast from server to Proxy server path have been considered. Finally it can be seen that the combination of these techniques and P2P networking for multicast video streaming has potential to fulfill future media internet challenges. Future internet initiatives should be taken to take into considerations these techniques while designing new architectures and protocols to enable future personalized applications and services, operating under high heterogeneous and dynamic environments for maximizing the QoS of the users.

7. REFERENCES

[1] Jagadeesh M. Dyaberi, Karthik Kannan, and Vijay S. Pai, “Storage Optimization for a Peer-to-Peer Video-on-Demand Network”, MMSys’10, February 22–23, 2010,

[2] Yifeng He, Ivan Lee and Ling Guan, “Distributed Throughput Maximization in P2P VoD Applications IEEE TRANSACTIONS ON MULTIMEDIA, VOL.11, NO.3, APRIL 2009

[3] Birendra Rai and Rajesh Bhadada, “Futuristic Scope of Automated Video on demand System: Vision and Design” International Conference on Emerging & Futuristic System and Technology, at LIET, Alwar (Raj.), April 2009

[4] Alvaro Chalar Zúñiga, Departamento Arquitectura decomputadoresy Sistemas Operativos, Universidad Autónoma de Barcelona,“System Analysis of a Peer-to-Peer Video-on- Demand Architecture” Kangaroo, 2008

[5] Weiqiang Sun, Kan Lin and Yang Guan, “Performance Analysis of a Finite Duration Multichannel Delivery Metho in IPTV” IEEE TRANSACTIONS ON BROADCASTING, VOL. 54, NO. 3, SEPTEMBER 2008

[6] Bin Cheng, Xiu zheng Liu, Zheng Zhang, Hai Jin,Lex Stein, Xiaofei Liao,“Evaluation and optimization of a peer- to-peer video-on-demand system”, Journal of Systems Architecture 54 (2008) 651– 663, December 2007

[7] Thouin F. and Coates M., “Video-on-Demand Networks: Design Approaches and Future Challenges”, in *IEEE Network* , Volume 21, Issue 2, pages 42-48, March – April 2007.

[8] D. Meddour, M. Mushtaqm, and T. Ahmed. “Open Issues in P2P Multimedia Streaming”. MULTICOMM. Istambul. Jun 2006.

[9] B. Wang, S. Sen, M. Adlerand D. Towsley,“Using Multicast for streaming videos across wide area networks,” Tech. Rep. 03-28, Department of Computer Science, University of Massachusetts, Amherst, 2003

[10] B.Wang, S. Sen, M. Adler and D. Towsley, “Optimal Proxy Cache allocation for efficient streaming media distribution,” in *Pro. IEEE INFOCOM*, 2002.

[11] J. C. I. Chuang and M.A. Sirb, “Pricing Multicast Communication: Acost-based approach, Telecommunication Systems, vol. 17, no.3, pp. 281– 297, 2001.

[12] Dapeng Wu, Yiwei Thomas Hou, Wenwu Zhu, Ya-Qin Zhang, and Jon M. Peha, “Streaming Video over the Internet: Approaches and Directions” IEEE Transactions on Circuits and Systems for Video Technology, Vol. 11, No. 3, pp. 282–299, March 2001

[13] D. Wu,Y. T. Hou,and Y.-Q.Zhang, “Transporting Real-Time Video over the Internet: Challenges and Approaches”, *Proceedings of the IEEE*, vol. 88, no. 12, Dec.2000.

[14] L. Gao and D. Towsley, “Threshold-based multicast for Continuous media delivery”, *IEEE Transactions on Multimedia*, vol. 3, pp. 405– 414, December 2001.

[15] Bruce A.Mah, Srinivasan Seshan, Kimberly Keeton, Randy H. Katz, Domenico Ferrari,“Providing Network Video Service to Mobile Clients1”, Appeared in Proceedings of the Fourth Workshop on Workstation Operating Systems, Napa, CA, October 1993.