Volume 2, Issue 10, October 2012 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

Live Streaming Based on Peer Division Multiplexing for Next Generation IPTV Network

S Praveen Kumar
PG Student, CSE Department , CMRIT,
Hyderabad, Andra Pradesh, India

A Vasanthi
Asso. Prof, CSE Department, CMRIT
Hyderabad , Andra Pradesh, India

P.Velangini Mary
PG Student, CSE Department , Dr.PaulRaj Engg.& Tech,
Badrachalam, Andra Pradesh, India

K.Rajeev Reddy
Asst. Prof, CSE Department, Shadan College of Engg. Tech.
Peerancheruvu, Andra Pradesh, India

*Abstract: **Next generation broadcast network concentrates on IPTV (Internet Protocol TeleVision). The main hurdle in IPTV is streaming of audio and video signals. A number of commercial systems are built to study and analyze the behaviour of live streaming of audio and video signals. Peer to Peer multiplexing (P2P) provides a good solution for this problem. In this paper a variation of P2P multiplexing is proposed which is called as receiver based P2P multiplexing. To analyze the performance of the proposed multiplexing techniques the very famous European network "Zatto" is considered. This paper also describes the network architecture of Zatto and uses the data collected from the provider to evaluate the performance of the proposed variation in P2P multiplexing.***

Keywords: IPTV, Live Streaming, Peer to Peer Multiplexing.

I. INTRODUCTION

Current generation broadcast network for TV is DTH which will be slowly replaced by the next generation Internet Protocol Television (IPTV) network. There is an emerging market for IPTV. Numerous commercial systems now offer services over the Internet that is similar to traditional over-the-air, cable, or satellite TV. Live television, time-shifted programming, and content-on-demand are all presently available over the Internet. Increased broadband speed, growth of broadband subscription base, and improved video compression technologies have contributed to the emergence of these IPTV services [1][6]. IPTV systems deliver video and audio channels to viewing devices by switching a single channel to multiple sources. IP Television networks are primarily constructed of computer servers, gateways, access connections and end user display devices. Servers control the overall system access and processing of channel connection requests and gateways convert the IP television network data to signals that can be used by television media viewers.

Content aggregation is the process of combining multiple content sources for distribution through other communication channels. A head end is part of a television system that selects and processes video signals for distribution into a television distribution network. The core network is the central network portion of a communication system. The core network primarily provides interconnection and transfer between edge networks. An access network is a portion of a communication network (such as the public switched telephone network) that allows individual subscribers or devices to connect to the core network. A premises distribution network (PDN) consists of the equipment and software that are used to transfer data and other media in a customer's facility or home. A viewing device is a combination of hardware and software that can convert media such as video, audio or images into a form that can be experienced by humans. The network archicture of IPTV is shown in Fig1.

Peer to peer multiplexing is mainly used for Live streaming of video and audio. User draw a distinction between three uses of peer-to-peer (P2P) [1] networks: delay-tolerant file download of archival material, delay-sensitive progressive download (or streaming) of archival material, and real-time live streaming. In the first case, the completion of download is elastic, depending on available bandwidth in the P2P network. The application buffer receives data as it trickles in and informs the user upon the completion of download. The user can start playing back the file for viewing in the case of a video file. Bit torrent and variants are examples of delay-tolerant file download systems. In the second case, video playback starts as soon as the application assesses it has sufficient data buffered that, given the estimated download rate and the playback rate, it will not deplete the buffer before the end of file. If this assessment is wrong, the application would have to either pause playback or rebuffed or slow down playback. While users would like playback to start as soon as possible, the application has some degree of freedom in trading off playback start time against estimated network capacity. Most video-on-demand systems are examples of delay-sensitive progressive-download application. The third case, real-time live streaming has the most stringent delay requirement. While progressive download may tolerate initial buffering of tens of seconds or even minutes, live streaming generally cannot tolerate more than a few seconds of buffering. Taking into account the delay introduced by signal ingest and encoding, and network transmission and propagation, the live streaming system can introduce only a few seconds of buffering time end-to-end and still be considered "live". The Zattoo peer-to-peer live streaming system was a free-to-use network serving over 3 million registered users in eight European countries at the time of study, with a maximum of over 60 000 concurrent users on a single channel. The system delivers live streams using a receiver-based, peer-division multiplexing scheme.

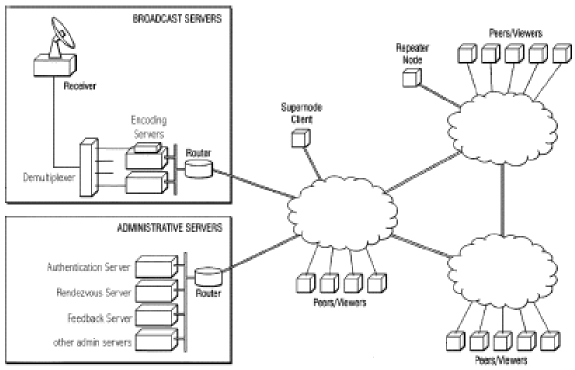

Fig. 1 Architecture of IPTV

To ensure real-time performance when peer uplink capacity is below requirement, Zattoo subsidizes the network's bandwidth requirement, as described in. After delving into Zattoo's architecture in detail, large-scale measurements collected during the live broadcast of the UEFA European Football Championship, one of the most popular one-time events in Europe, in June 2008 [5]. During the course of that month, Zattoo served more than 35 million sessions to more than 1 million distinct users.

II. SYSTEM ARCHITECTURE

The Zattoo system rebroadcasts live TV, captured from satellites, onto the Internet. The system carries each TV channel on a separate peer-to-peer delivery network and is not limited in the number of TV channels it can carry. Although a peer can freely switch from one TV channel to another, thereby departing and joining different peer-to-peer networks, it can only join one peer-to-peer network at any one time.

Users are required to register themselves at the Zattoo Web site to download a free copy of the Zattoo player application. To receive the signal of a channel, the user first authenticates itself to the Zattoo *Authentication Server*. Upon authentication, the user is granted a ticket with limited lifetime. The user then presents this ticket, along with the identity of the TV channel of interest, to the Zattoo *Rendezvous Server*. If the ticket specifies that the user is authorized to receive signal of the said TV channel, the Rendezvous Server returns to the user a list of peers currently joined to the P2P network carrying the channel, together with a signed channel ticket. If the user is the first peer to join a channel, the list of peers it receives contain only the Encoding Server. The user joins the channel by contacting the peers returned by the Rendezvous Server, presenting its channel ticket, and obtaining the live stream of the channel from them.

Zattoo uses the Reed–Solomon (RS) error correcting code (ECC) for forward error correction [2]. The RS code is a systematic code: of the n packets sent per segment, $k<n$ packets carry the live stream data, while the remainder carries the redundant data [3]. Due to the variable-bit rate nature of the data stream, the time period covered by a segment is variable, and a packet may be of size less than the maximum packet size.

III. RECEIVER BASED P2P MULTIPLEXING

A. P2P Multiplexing

FWhen a new peer requests to join an existing peer, it specifies the substream(s) it would like to receive from the existing peer [4]. These substreams do not have to be consecutive. Contingent upon availability of bandwidth at existing peers, the receiving peer decides how to multiplex a stream onto its set of neighboring peers, giving rise to our description of the Zattoo live streaming protocol as a receiver-based, peer-division multiplexing protocol.

To minimize per-packet processing time of a stream, the Zattoo protocol sets up a virtual circuit with multiple fan outs at each peer. When a peer joins a TV channel, it establishes a peer-division multiplexing (PDM) scheme among a set of neighboring peers by building a virtual circuit to each of the neighboring peers. Baring departure or performance degradation of a neighbor peer, the virtual circuits are maintained until the joining peer switches to another TV channel. With the virtual circuits set up, each packet is forwarded without further per-packet handshaking between peers.

The PDM establishment process consists of two phases: the *search* phase and the *join* phase.

Search Phase: To obtain a list of potential neighbors, a joining peer sends out a SEARCH message to a random subset of the existing peers returned by the Rendezvous Server. The SEARCH message contains the substream indices for which this joining peer is looking for peering relationships. The joining peer continues to wait for SEARCH replies until the set of potential neighbors contains at least a minimum number of peers, or until all SEARCH replies have been received.

Join Phase: Once the set of potential neighbors is established, the joining peer sends JOIN requests to each potential neighbor. The JOIN request lists the substreams for which the joining peer would like to construct virtual circuit with the potential neighbor.

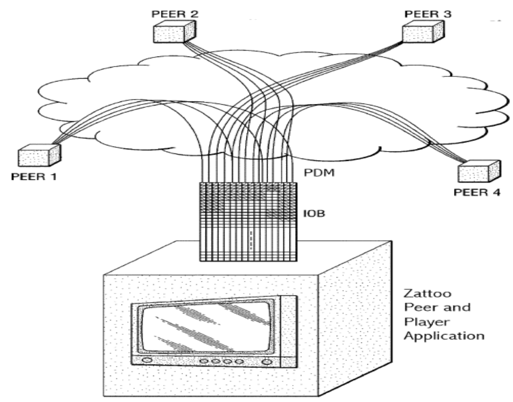

Fig 2. Peer system with an IOB

B. Stream Management

The IOB is referenced by an *input pointer*, a *repair pointer*, and one or more *output pointers*. The input pointer points to the slot in the IOB where the next incoming packet with sequence number higher than the highest sequence number received so far will be stored. The repair pointer always points one slot beyond the last packet received in order and is used to regulate packet retransmission and adaptive PDM [3] as described later. Different peers may request for different numbers of, possibly nonconsecutive, substreams. To accommodate the different forwarding rates and regimes required by the destinations, we associate a packet map and forwarding discipline with each output pointer.

Fig. 3 shows the packet map associated with an output peer pointer where the peer has requested substreams 1, 4, 9, and 14. Every time a peer pointer is repositioned to the beginning of a sub-buffer of the IOB, all the packet slots of the requested substreams are marked NEEDed and all the slots of the substreams not requested by the peer are marked SKIP. When a NEEDed packet arrives and is stored in the IOB, its state in the packet map is changed to READY. As the peer pointer moves along its associated packet map, READY packets are forwarded to the peer and their states changed to SENT. A slot marked NEEDed but not READY, such as slot n+4

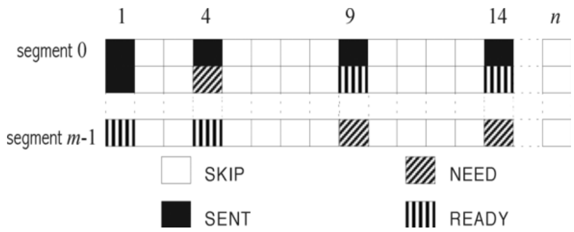

Fig. 3 Packet Map Associated with peer map

In addition to achieving lossless recording, we use retransmission to let a peer recover from transient network congestion. A peer sends out a retransmission request when the distance between the repair pointer and the input pointer has reached a threshold of R packet slots, usually spanning multiple segments. A retransmission request consists of an R bit packet mask, with each bit representing a packet, and the sequence number of the packet corresponding to the first bit. Marked bits in the packet mask indicate that the corresponding packets need to be retransmitted. When a packet loss is detected, it could be caused by congestion on the virtual circuits forming the current PDM or congestion on the path beyond the neighboring peers. In either case, current neighbor peers will not be good sources of retransmitted packets.

C. Adaptive PDM

Peers on the Zattoo network can redistribute a highly variable number of substreams, reflecting the high variability in uplink bandwidth of different access network technologies [7]. For a full-stream consisting of 16 *constant*-bit rate substreams, our prior study showed that based on realistic peer characteristics measured from the Zattoo network, half of the peers can support less than half of a stream, 82% of peers can support less than a full-stream, and the remainder can support up to 10 full streams (peers that can redistribute more than a full stream is conventionally known as supernodes in the literature) [5]. With variable- bit rate streams, the bandwidth carried by each substream is also variable. To increase peer bandwidth usage, without undue degradation of service, we instituted measurement-based admission control at each peer. In addition to controlling resource commitment, another goal of the measurement-based admission control module is to continually estimate the amount of available uplink bandwidth at a peer.

The amount of available uplink bandwidth at a peer is initially estimated by the peer sending a pair of probe packets to Zattoo's Bandwidth Estimation Server. Once a peer starts forwarding substreams to other peers, it will receive from those peers quality-of-service feedbacks that inform its update of available uplink bandwidth estimate. A peer sends quality-of-service feedback only if the quality of a substream drops below a certain threshold [9]. Upon receiving quality feedback from multiple peers, a peer first determines if the identified substreams are arriving in low quality. If so, the low quality of service may not be caused by limit on its own available uplink bandwidth—in which case, it ignores the low quality feedbacks. Otherwise, the peer decrements its estimate of available uplink bandwidth. If the new estimate is below the bandwidth needed to support existing number of virtual circuits, the peer closes a virtual circuit. To reduce the instability introduced into the network, a peer closes first the virtual circuit carrying the smallest number of sub streams.

Each peer on the Zattoo network is assumed to serve a user through a media player, which means that each peer must receive, and can potentially forward, all sub streams of the TV channel the user is watching. The limited redistribution capacity of peers on the Zattoo network means that a typical client can contribute only a fraction of the sub streams that make up a channel. This shortage of bandwidth leads to a global bandwidth deficit in the peer-to-peer network.

In the Zattoo system, two separate centralized collector servers collect usage statistics and error reports, which we call the “stats” server and the “user-feedback” server respectively.The “stats” server periodically collects aggregated player statistics from individual peers, from which full session logs are constructed and entered into a session database. The session database gives a complete picture of all past and present sessions served by the Zattoo system. A given database entry contains statistics about a particular session, which includes join time, leave time, uplink bytes, download bytes, and channel name associated with the session.

Receiver based multiplexing ensures error free and feedback based server in Zatoo collects the feedback and tries to reduce the delay in data delivery.

IV. CONCLUSION

A receiver-based, peer-division multiplexing engine to deliver live streaming content on a peer-to-peer network. The same engine can be used to transparently build a hybrid P2P/CDN delivery network by adding Repeater nodes to the network. By analyzing a large amount of usage data collected on the network during one of the largest viewing events in Europe, we have shown that the resulting network can scale to a large number of users and can take good advantage of available uplink bandwidth at peers. We have also shown that error-correcting code and packet retransmission can help improve network stability by isolating packet losses and preventing transient congestion from resulting in PDM reconfigurations.

REFERENCES

[1] R. auf der Maur, “Die Weiterverbreitung von TV- und Radioprogrammenüber IP-basierte Netze,” in *Entertainment Law*, F. d. Schweiz, Ed., 1st ed. Bern, Switzerland: Stämpfli Verlag, 2006.

[2] “Euro2008,” UEFA [Online]. Available: http://www1.uefa.com/ [3] S. Lin and D. J. Costello Jr., *Error Control Coding*, 2nd ed. Englewood Cliffs, NJ: Pearson Prentice-Hall, 2004.

[3] S. Xie, B. Li, G. Y. Keung, and X. Zhang, “CoolStreaming: Design, theory, and practice,” *IEEE Trans. Multimedia*, vol. 9, no. 8, pp. 1661–1671, Dec. 2007.

[4] K. Shami *et al.*, “Impacts of peer characteristics on P2PTV networks scalability,” in *Proc. IEEE INFOCOM*, Apr. 2009, pp. 2736–2740.

[5] “Bandwidth test statistics across different countries,” Bandwidth- Test.net [Online]. Available: http://www.bandwidth-test. net/stats/country/

[6] X. Hei, C. Liang, J. Liang, Y. Liu, and K. W. Ross, “Insights into PPLive: A measurement study of a large-scale P2P IPTV system,” in *Proc. IPTV Workshop, Int. World Wide Web Conf.*, May 2006.

[7] B. Li *et al.*, “An empirical study of flash crowd dynamics in a P2Pbased live video streaming system,” in *Proc. IEEE GLOBECOM*, 2008, pp. 1–5.

[8] J. Rosenberg *et al.*, “STUN—Simple traversal of User Datagram Protocol (UDP) through network address translators (NATs),” RFC 3489, 1993.

[9] A. Ali, A. Mathur, and H. Zhang, “Measurement of commercial peer-to-peer live video streaming,” in *Proc. Workshop Recent Adv. Peer-to-Peer Streaming*, Aug. 2006.

