Quality of Service Evaluation for Peer Selection in Peer-to-Peer Streaming Services

Supitchaya Chanpong, Sinchai Kamolphiwong, and Boonchai Ngamwongwattana

***Abstract*—Providing the quality video content in P2P IPTV services, it is agreed that have good neighour selection algorithm can improve a service quality. Content distribution requires a high performance overlay structure to deliver satisfying quality of service (QoS). Good neighbours can vastly influence the QoS of the application running on those peers. Recently, peer selection algorithms are mainly based on delay or bandwidth, such parameters that vary by time may not specify peer quality. Hence, we have proposed an improvement of peer selection algorithm by evaluating quality of video content delivered using end-to-end measuring approach. We calculate the video time delay from buffer map of each peer which can reflect the target peer quality in the system perspective. Also, we consider each peer in the new joining peer perspective by paying attention to the path from each peer to the new joining peer. With this algorithm, we can discover the real proper and good neighbours which can provide a good service to a new joining peer.**

***Index Terms*—Peer-to-Peer Streaming, quality of service evaluation, neighbour selection algorithm, P2P IPTV.**

I. Introduction

IPTV is a service that increases the convenience in accessing television to users by using the Internet technology; as a result, it gains more popular rapidly. However, the traditional Internet television services are based on client-server approach which limits the available resources and cannot support the rapidly increasing requirements. So peer-to-peer (P2P) system has been considered as an alternative, and become an increasingly popular approach for streaming live media over the Internet due to its potential scalability and ease of deployment. This approach is referred to as P2P streaming.

There are 2 important design issues for constructing P2P streaming networks: (i) How to form an overlay topology between peers, and (ii) How to transmit video content efficiently. Recent approaches can be categorized into 2 categories (i) peers form a tree-shaped overlay and video content was delivered from the origin server to peers, so called the tree-push approach, (ii) peers form a mesh-shaped overlay and pull video from each other to deliver content, so called mesh-pull approach. Over years, a lot of tree-push systems were proposed and evaluated in academic with some success achievement. However, they have never deployed for commercial while mesh-pull IPTV systems achieved a number of deployment such as [1], [2], [3] and [4]. Mesh-based overlays make the system more robust to network fluctuations without the need of a global mechanism to maintain the overlay. As a result, mesh-pull systems are more suitable for P2P streaming in dynamic environments.

Providing the efficiency P2P IPTV services must take into considerations of how peers can select their neighbors properly, and construct a data distribution structure among the collaborative peers effectively. Content distribution requires a high performance overlay structure to deliver satisfying quality of service. A good neighbour can vastly influence the QoS of the application running on that peer. Consequently, it is the neighbour selection that determines the perceived quality at each peer.

There are a number of studies on neighbour selection in P2P streaming that can be categorised into 2 categories: considering time delay between peers, and considering available network bandwidth. Peer selection based on time delay implemented by using cross layer of underlying network to measure RTT value between peers, e.g. in [5], [6], [7], [8] and [9]. In the other hand, peer selection based on bandwidth is achieved by improving the variable neighbour selection to determine the number of neighbour suitable for a peer based on outgoing bandwidth of those peers [10]. These can utilise the upload capacities of those peers and improve a network performance. However, the measured value of the outgoing bandwidth is unreliable because it is fluctuated and depend on many factors.

In this paper, we have proposed the neighbour selection algorithm which evaluated the quality of video content based on end-to-end approach. The most important measurement of service quality of a streaming system is the continuity of video playback at the user host. If a particular chunk arrives after the playback deadline, the peer has 2 options: (i) freeze video in recent frame and wait for the missing chunk (ii) skip the playback of frames in the chunk. Packet loss in P2P streaming is not only caused by the transmission loss in network but also depend on the lack of bandwidth and content at the supplying peers. The peers with larger playback lags will not upload useful chunks to peers with smaller lags. Motivated by this concern, we calculate playback time directly from buffer map of each target peer which can indicate the target peer quality in the system perspective. Also, we consider each peer in the new joining peer perspective by measuring end-to-end delay from each neighbour peer to new joining peer. With this algorithm, we can discover the real proper neighbour providing the P2P IPTV services to the new joining peer with the most recently playback time.

The rest of this paper are organized as follows: we first provide an overview of mesh-pull system in Section 2.

Manuscript received March 21, 2012; revised May 9, 2012.

S. Chanpong and S. Kamolphiwong are with the Department of Computer Engineering, Faculty of Engineering, Prince of Songkla University, Thailand.

B. Ngamwongwattana is with the NECTEC, NSTDA, Ministry of Science and Technology, Thailand (e-mail: ferin.mac@gmail.com; tel.: +66814167477).

Section 3 presents our proposed peer selection algorithm. Section 4 presents the simulation results and analysis. Finally, we conclude our paper in Section 5.

II. Overview of Mesh-Pull P2P Live Streaming System

Mesh-pull streaming systems, including PPLive [11], PPStream [12], and CoolStreaming [1], have a common architecture: A live video stream is divided into media chunks. A peer that interested in the video requests a list of peers currently watching the video from the system; then, establishes partner relationships with a subset of peers on the list. Each peer receives buffer maps, which indicates the chunks the partner available, from its neighbour partners. Each host requests the chunks needed in the near future from its partners using a scheduling algorithm.

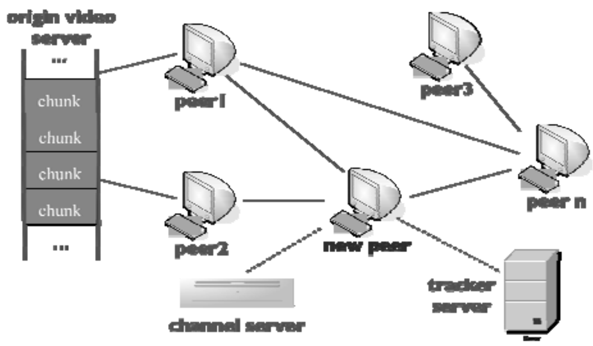

Fig. 1. Mesh-pull P2P live streaming architecture.

The peer includes a P2P streaming engine and a media player. The streaming engine retrieves chunks from partner peers and stores the retrieved chunks in its cache, and sends a copy of each chunk to the media player as shown in Fig. 2. The peer sends a buffer map to each of its partner peers. All partners learn from the buffer map then request for specific chunks the peer has. The peer then sends the requested chunks to its partners.

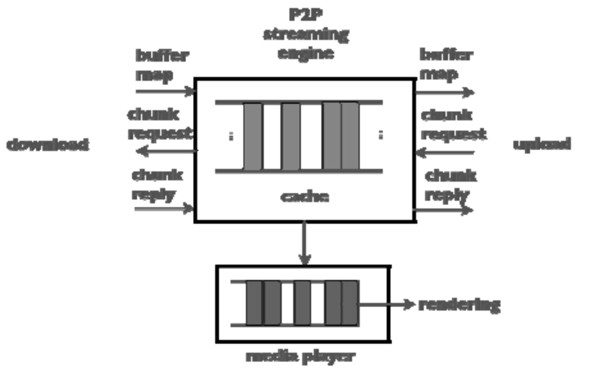

Fig. 2. P2P streaming engine and media player.

Peers send buffer map messages to each other, to indicate which chunks a peer currently has buffered and can be shared. The buffer map message includes the ID of the first chunk is the chunk offset, the length of the buffer map, and a string of zeroes and ones indicating which chunks are available. The structure of a buffer map shown in Fig. 3. The peer can request one or more chunks that a partner has advertised in the buffer map. A peer may download chunks from several peers simultaneously. The streaming engine continually searches for new partners where it can download chunks efficiently.

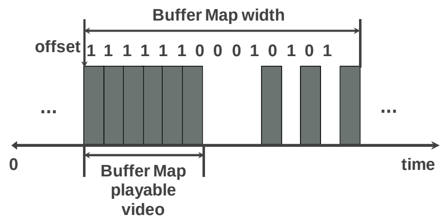

Fig. 3. A video peer's buffer map.

III. The Proposed Peer Selection Algorithm

A. *Pee Selection Algorithm*

When a new peer joins the system, it first chooses a desired video channel from video programmes. A request is sent to a tracker. The tracker then sends a list of peer providing who are receiving the same video channel to new peer with channel information (chunk rate). New peer floods a probe message limiting TTL to all peers in the list. Then each target peer reply ACK probe message back to new peer with its information (chunk offset ID) and measuring path information (end-to-end delay). New peer evaluates the quality of each target to select the proper target peer as its neighbour, establish the neighbour relationship to request for the video content.

As mentioned above, our proposed algorithm focus on evaluating the quality of target peers; to guarantee that the selected peer have the most recently video chunk, except considering the peer performance in system perspective, the content delivery path from that peer to requesting peer is also important. If the target peer has a good performance but its path to the requesting peer is bad (such as more delay or it has bandwidth bottleneck), the requested contents may not be transmitted successfully or it take more delay and arrive after the playback deadline. Motivated by this, the proposed algorithm also consider path quality in term of end-to-end delay, to find the better link performance together with peer quality to discover the most proper peers for each new joining peer.

B. *Video Quality Service Evaluation*

The requesting peer or the new joining peer evaluates the quality of each target peer in term of the video playback time. The score of each target, the requesting peer calculated, reflects the estimated video playback time in the requesting peer if it receives media chunks from that target. That means the target peer giving the most score can provide the video with the most recently playback time. To this end, we first estimate the chunk offset ID in each target peer at the time requesting peer sends probe messages, as the following equation.

$$ID_{t_0} = ID_{t_i} - (Delay \times Rate)$$

Where ID_{t0} denotes the estimated chunk offset ID of target peer at the requesting time, ID_{ti} denotes the offset ID of target peer when it receive probe message, Delay denotes the end-to-end delay between the requesting peer and target, and Rate denotes the video chunk rate which is constant for each video channel.

Now we know the chunk offset ID of each target at the same time (reference from the time sending probe message). The offset is the ID of the first chunk in the buffer map that reflects the playback point of that peer. With the above equation, we can estimate the playback point of each target at the same time. Then we consider the path between the requesting peer and its target peers in term of end-to-end delay, we calculate a score for each target to estimate the playback time at the requesting peer if it receives video chunks from that target; as the following equation. The target peer giving the best score can provide the most recently video playback time.

$$Score = \frac{ID_{t_0}}{Rate} - Delay$$

C. Simulation Scenario

Source (s) streaming media contents at chunk rate of 10 chunks/s. Delay between peer1 (p1) and peer2 (p2) and the requesting peer (n) is 20 ms and 15 ms respectively, as shown in Fig.4.

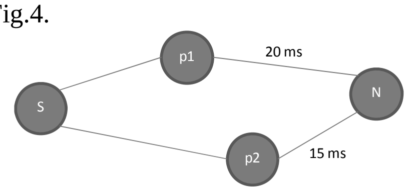

Fig. 4. Playback time lags of each peer in the proposed selection.

The requesting peer sends probe message at time t = 5 s to peer1 and peer2. Peer1 replies probeACK with chunk offset ID 40. While peer2 replies with video offset 39. From the proposed equation, we calculate a score of peer1 and peer2 as 3.96 and 3.87 respectively; that means peer1 can provide the video content with the most recently playback time even if delay from peer1 to the requesting peer is more than peer2.

D. Equation Analysis

From the above sample scenario the video playback rate is 10 chunks/s; at time t = 5 s the offset ID of source must be 50, at that time the requesting peer sends probe message to peer1 and peer2. Peer1 receives probe message at time 5.020 s and reply with offset ID 40; from the first equation we can calculate the offset ID of peer1 at time 5 s is 39.80, that means at time 5 s the video playback time of peer1 is 3.98 and if the requesting peer receives video chunks from peer1, its playback time at t = 5 s is 3.96 as the second equation. The calculation of peer2 is in the same way.

In another aspect, we analyse the equation from the playback time lags. Considering peer1 at time t =5.020 s, the offset ID is 40 while source play chunk is ID = 50.2, that means peer1 has chunk lagging from source 10.2 chunks or 1.02 s. If the requesting peer receives the video chunk from peer1, total lag from source to itself is 1.04 s. Notices that at time t = 5 s playback time will be 5 - 1.04 = 3.96 s, that equal to the score calculated from the equation.

IV. Simulation Result Analysis

This section presents the simulated results of the proposed peer selection algorithm by using the OMNeT++ [13]. We compare the proposed algorithm with randomly peer selection algorithm in terms of playback time lags of each target peer. Our scenario has 100 nodes connected in mesh topology with random link bandwidth between 400 kbps and 1 Mbps. The media playback bit rate is 381 kbps in average.

The results show that the proposed peer selection can choose peer to form the better performance overlay network which makes a lower playback time lags via each target peer comparing with the randomly peer selection scheme. Fig 5 shows the playback time lags of each node comparing between randomly selection and the proposed selection; while table 1 compares the 100-nodes mesh network playback time lags of the proposed selection with the randomly peer selection scheme.

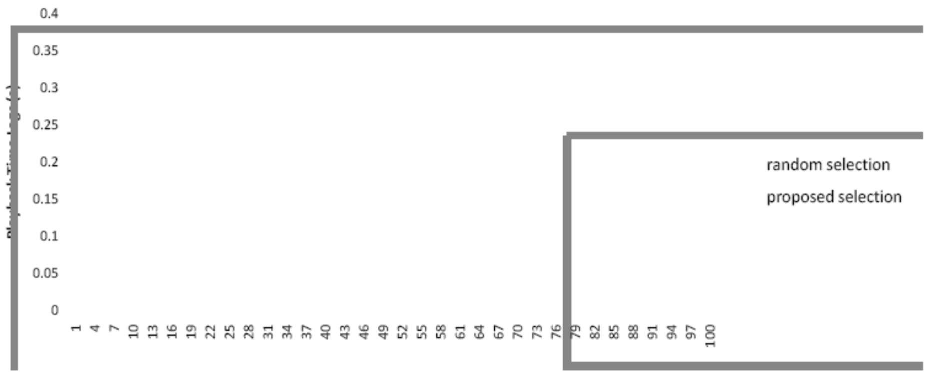

Fig. 5. Comparison of end node video playback time lags.

TABLE I: Video Playback Time Lag Comparison of A 100-Node Mesh Network

Playback Time Lags	Random Selection (ms)	Proposed Selection (ms)
Max	372	104
Min	42	42
Average	163	84

V. Conclusion

This paper has proposed a novel neighbour selection algorithm for IPTV P2P networks. The scheme considers both peer performance and link performance along the path between the video source and a new join peer, known as end-to-end approach. Instead of considering link bandwidth and link delay, our approach focuses on video delay, as well as other related parameters, e.g. a number of hop count. To that end, we consider neighbor (or target peers) discovery scheme who have the best performance to provide a good level of video feeding. We evaluate a video stream path performance via target peers, in terms of it most updated video chunk, delay time, and a number of hops count. Video chunk is analised directly from a peer buffer map, and is considered together with link performance based on end-to-end delay from a video source to a new join peer via each target peer. We estimate a score value for a video playback time of each selected target peer. With this algorithm, we can discover the real proper neighbour for a new joining peer to distribute good quality IPTV service. We compared the proposed scheme other schemes, e.g. randomly peer selection. We have shown that the proposed scheme can minimise the lag time of video playback between a video source and a new join peer.

ACKNOWLEDGEMENTS

This paper is under the NTC scholarship programme supported by the Telecommunications Research Industrial and Development Institute (TRIDI), the National Broadcasting and Telecommunications Commission (NTC), Thailand.

REFERENCES

[1] Xinyan Zhang, Jiangchuan Liu, Bo Li, and Y.-S. . Yum, “CoolStreaming/DONet: a data-driven overlay network for peer-to-peer live media streaming,” in Proceedings IEEE INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies, 2005, vol. 3, pp. 2102– 2111 vol. 3.

[2] X. Hei, C. Liang, J. Liang, Y. Liu, and K. W. Ross, “A Measurement Study of a Large-Scale P2P IPTV System.”

[3] Y. Tang Yun, J.-G. Luo Jian-Guang, M. Zhang Meng, S.-Q. Yang Shi-Qiang, and Q. Zhang Qian, “Deploying P2P networks for large-scale live video-streaming service [Peer-to-Peer Multimedia Streaming],” IEEE Communications Magazine, vol. 45, no. 6, pp. 100–106, Jun. 2007.

[4] A. Sentinelli, G. Marfia, M. Gerla, L. Kleinrock, and S. Tewari, “Will IPTV ride the peer-to-peer stream? [Peer-to-Peer Multimedia Streaming],” IEEE Communications Magazine, vol. 45, no. 6, pp. 86–92, Jun. 2007.

[5] Huang YongXiang, Qian DePei, Wu WeiGuo, and Liu Tao, “Whirlpool: Structuring Mesh-Based Protocol for QoS-Aware Peer-to-Peer Streaming,” in Seventh International Conference on Grid and Cooperative Computing, 2008. GCC ’08, 2008, pp. 412–420.

[6] S. M. . Shams, P. E. Engelstad, and A. Kvalbein, “Analysis of peer selection algorithms in cross-layer P2P architectures,” in 2009 IEEE International Conference on Internet Multimedia Services Architecture and Applications (IMSAA), 2009, pp. 1–6.

[7] J. Liang and K. Nahrstedt, “DagStream: Locality Aware and Failure Resilient Peer-to-Peer Streaming,” IN SPIE MULTIMEDIA COMPUTING AND NETWORKING (MMCN, vol. 2006, 2006.

[8] M. Alhaisoni, M. Ghanbari, and A. Liotta, “Improving QoS in P2P Video Streaming,” in First International Conference on Advances in Multimedia, 2009. MMEDIA ’09, 2009, pp. 98–103.

[9] M. Mushtaq and T. Ahmed, “Adaptive Packet Video Streaming Over P2P Networks Using Active Measurements,” in 11th IEEE Symposium on Computers and Communications, 2006. ISCC ’06. Proceedings, 2006, pp. 423– 428.

[10] J. Ghosha, Miao Wang, Lisong Xu, and B. Ramamurthy, “Variable neighbor selection in live peer-to-peer multimedia streaming networks,” in 5th International Conference on Broadband Communications, Networks and Systems, 2008. BROADNETS 2008, 2008, pp. 344–346.

[11] [Online]. Available: http://www.pptv.com/. [Accessed: 11-Jan-2012].

[12] [Online]. Available: http://www.ppstream.com/. [Accessed: 11-Jan-2012].

[13] “OMNeT++,” OMNeT++ Network Simulation Framewrok. [Online]. Available: http://omnetpp.org. [Accessed: 05-Apr-2012].

[14] S. Banerjee, B. Bhattacharjee, and C. Kommareddy, “Scalable application layer multicast,” SIGCOMM Comput. Commun. Rev., vol. 32, no. 4, pp. 205–217, Aug. 2002.

[15] Yang-hua Chu, S. G. Rao, S. Seshan, and Hui Zhang, “A case for end system multicast,” IEEE Journal on Selected Areas in Communications, vol. 20, no. 8, pp. 1456– 1471, Oct. 2002.

[16] D. A. Tran, “ZIGZAG: An efficient peer-to-peer scheme for media streaming,” IN PROC. OF IEEE INFOCOM, 2003.

[17] R. Rejaie, “A Framework for Architecting Peer-to-Peer Receiver-driven Overlays,” IN NOSSDAV’04, p. 42–47, 2004.

[18] X. Liao, H. Jin, Y. Liu, L. M. Ni, and D. Deng, “AnySee: Peer-to-Peer Live Streaming,” in INFOCOM 2006. 25th IEEE International Conference on Computer Communications. Proceedings, 2006, pp. 1–10.

[19] N. Magharei and R. Rejaie, “PRIME: Peer-to-Peer Receiver-Driven Mesh-Based Streaming,” IEEE/ACM Transactions on Networking, vol. 17, no. 4, pp. 1052–1065, 2009.