IEICE TRANS. INF. & SYST., VOL.E95–D, NO.12 DECEMBER 2012

LETTER Special Section on Parallel and Distributed Computing and Networking

A Push-Pull Chunk Delivery for Mesh-Based P2P Live Streaming

Chee Yik KEONG[†], *Nonmember*, **Poo Kuan HOONG**[†a)], ***Member***, ***and*** **Choo-Yee TING**[†], *Nonmember*

SUMMARY In this paper, we propose an adaptive chunk scheduling for mesh-based peer-to-peer live streaming system, a hybrid class of push and pull chunk delivery approach. The proposed rule-based push-pull scheduler simultaneously pull video chunk from lower latency peers to fill up missing chunks and push video chunk adaptively for rapid chunk delivery. We performed comparative simulation study against *rarest first* push-pull and *status-wise* push-pull to prove the efficiency of our proposed algorithm. Mesh-push is made possible by effectively exploiting the information through buffer map exchange. The findings of performance evaluation have suggested a better video continuity and achieved lower source to end delay.
key words: *P2P live streaming, mesh-based network, chunk scheduling, hybrid push-pull*

1. Introduction

Peer-to-Peer (P2P) video streaming has emerged as an alternative to conventional streaming systems. In P2P streaming, a video source server only needs to relay video chunks to some users, and the users in turn share their received chunks with their neighbors. The chunk distribution scenario applies the use of mesh-based network topology, where each peer is served by a few neighbors.

A P2P mesh-based network is designed for simplicity and scalability, with each peer has a set of neighbors and exchange data availability information periodically. To complete the video chunk distribution among neighbors, chunk scheduling plays an important module in P2P video streaming. In a nutshell, P2P mesh-based streaming protocol requires each peer to pull video chunks from its neighbors. This approach is favored because of the explicit request for missing video chunks can reduce chunk redundancy and traffic overhead. However, due to the streaming performance and delays associate by every explicit chunk request, tree structure push approach was proposed with mesh-based chunk scheduling for rapid chunk delivery [1], [2].

This paper focuses on P2P mesh-based chunk scheduling module by proposing rule-based push-pull algorithm and evaluates the efficiency of the proposed algorithm against rarest first push-pull and status-wise push-pull algorithm. The numbers of neighbors are taken into consideration to search for the optimal number of neighbors as partnership in an overlay network.

Manuscript received January 9, 2012.
Manuscript revised May 8, 2012.
[†]The authors are with the Faculty of Computing and Informatics, Multimedia University, Cyberjaya, Malaysia.
a) E-mail: khpoo@mmu.edu.my
DOI: 10.1587/transinf.E95.D.2958

2. Chunk Scheduling

2.1 Rarest First Push-Pull

In rarest first chunk scheduling, it selects the video chunk which the fewest of its neighbors have. This policy guarantees higher video chunk availability and prevents peer starvation [3]. The push and pull mechanism in rarest first algorithm relies on chunk rarity, which represents the occurrence of an individual chunk among the neighbors.

Duhwan et al. have, however, modified rarest first push-pull approach for P2P live streaming network because conventional rarest first policy ignores the urgent chunks for video playback [2]. For P2P live streaming, video chunks must be received and forwarded before the playback deadline where each video chunk contains time sensitive data. Their approach introduces each peer pulls common or urgent chunks from set of neighbors and voluntary pushes rare video chunks to neighbors by comparing the rarity index of each video chunk among neighbors.

2.2 Status-Wise Push-Pull

In status-wise algorithm, the push-pull decision is based on round robin strategy [4]. It works by an extensive signal between peers to switch status modes– push and pull mode. A peer is in push mode during playback even slots and in pull mode during odd slots. If a peer is in pull/push mode, a request will be sent to the selected neighbor for video chunks distribution. Upon confirmation, video chunks distribution starts by pull/push approach. At the end of the playback slot, the peer switches mode. However, in a real world scenario, upload bandwidth is a factor that leads to inadequate upload capacity to push video chunks during playback even slots. Hence, the even time slot for push mode is entirely wasted without any push approach.

2.3 Rule-Based Push-Pull

The idea of our proposed push-pull approach is to incorporate delay awareness and chunk position estimation in pull or push decision of a video chunk. During the buffer map exchange process, peers are required to record the round-trip time (RTT) which indicates the delay between two communication paths. The pull decision is based on lowest RTT, request from neighbor that guarantees lower network latency.

Input: Current peer: *peer_i*
Pull: if(bufferMapExchangeStart**)**
Send chunk request to neighbor with lowest RTT;
Push: if(getLastChunk() != latestReceiveChunk) //fresh chunk
Estimate each neighbor current chunk position;
if(estimatedChunkPosition > latestReceiveChunk)
if(neighbor pull request > *peer_i* pull request to neighbor) //high pull ratio
Push latestReceiveChunk to neighbor;
//push to the neighbor that fulfills rules

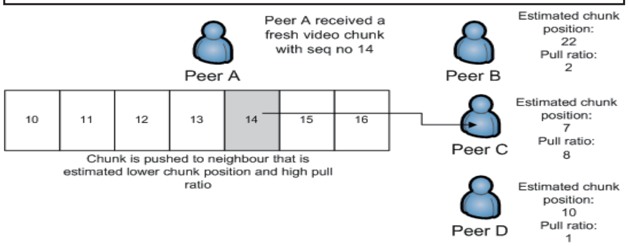

Fig. 1 Peer A pushes fresh video chunk 14 to Peer C based on estimated current chunk position and high pull ratio.

Chunk position estimation is introduced as a rule to push video chunk. In order to do so, scheduler estimates the current chunk position of the neighbor. Fresh video chunk received is pushed to a neighbor if it is positioned ahead of estimated video chunk of the neighbor (as shown in Fig. 1).

As mentioned, mesh-push comes at the cost of chunk redundancy, where more than one peer push the same video chunk to the same peer. Our approach resolves this problem is by introducing pull ratio. Pull ratio works as an incentive wise rule to push a video chunk to a neighbor that has a higher frequency of pull request to the peer compared with the pull request from the peer.

3. Performance Evaluation

We performed simulation study to compare our rule-based push-pull (i.e., with and without pull ratio to prove the efficiency in reducing chunk redundancy) with rarest first and status wise push-pull. We built our P2P live streaming modules on OMNET++ simulator with the setup of 1000 peers of random upload/download capacity. Peers are configured to have 5, 10, 15 and 20 neighbors.

Figure 2 (a) shows the way the continuity index behaves under different number of neighbors and different scheduling algorithms. Our findings show that better video continuity can be achieved as number of neighbors increases with the exclusion of status-wise push-pull despite some addition of traffic overhead in updating each peer status mode periodically. Number of neighbors from 10 till 15 was able to increase the probability in finding neighbors for video content, with lower chunk redundancy as shown in Fig. 2 (b). Figure 2 (c) shows that our proposed push-pull algorithm was able to reduce the source-to-end delay effectively vis-à-vis delay awareness for pull and chunk push. However, chunk redundancy is higher compared to status-wise which employs a round robin strategy.

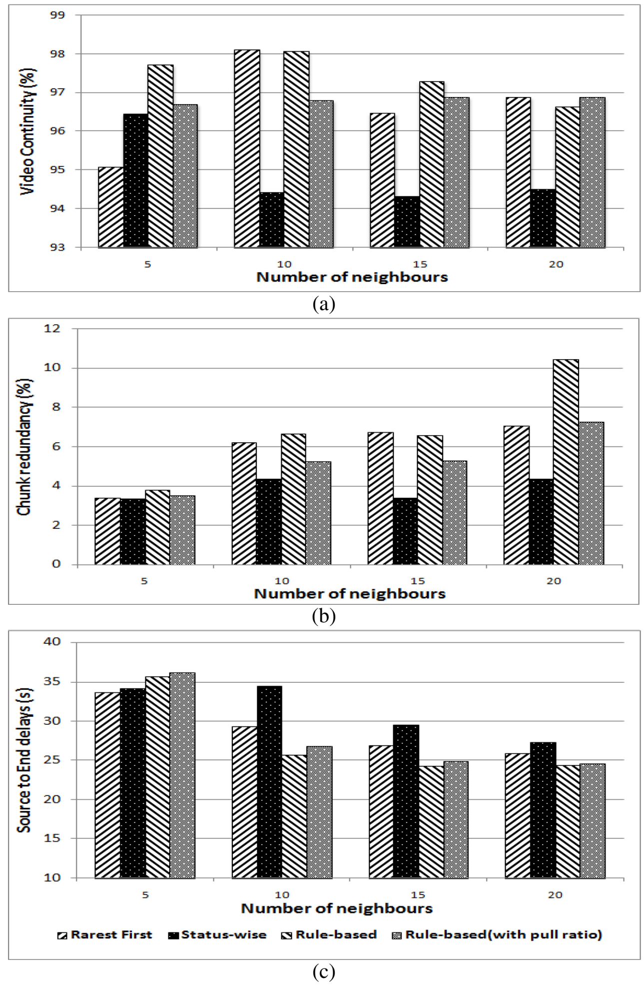

Fig. 2 Video continuity, chunk redundancy and source to end delays against number of neighbors.

4. Conclusion

In this paper we proposed hybrid class of push-pull chunk delivery approach for P2P mesh-based live streaming. In our approach, we had shown that employing a push-pull approach can reduce source-to-end delays and achieve higher video continuity.

References

[1] B. Li, S. Xie, Y. Qu, G.Y. Keung, C. Lin, J. Lie, and X. Zhang, "Inside new Coolstreaming: principles, measurements and performance implications," Proc. IEEE INFOCOM 2008, April 2008.
[2] D. Jo, S. Helal, E. Kim, and W. Lee, "Adaptive push-pull protocols for P2P based video streaming," In press.
[3] J.-L. Kuo, C.-H. Shih, and Y.-C. Chen, "A comprehensive study of delivery strategies with chunk scheduling for mesh P2P live streaming," 5th International Conference on Ubiquitous Information Management and Communication, ICUIMC 2011.
[4] R. Lo, C.A. Russo, and D. Carra, "On some fundamental properties of P2P push/pull protocols," 2nd International Conference of Communications and Electronics, 2008.