Volume 3, Issue 6, June 2013 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

Superchunk-Based Proficient Explore in P2P-VoD System

M.Sudharani

P.G Scholar,

Bharat University, Chennai,

Tamilnadu, India.

Abstract—In this project, we seek to provide reliable and fast substance discovery in peer-to-peer (P2P) video-on-demand(VoD) format to smooth the progress of user interactivity under peer dynamics. We first be acquainted with two characteristics of substance detection in P2P-VoD: real-time constraints and inadequate local cache. Tapping on these properties, we suggest a hybrid substance discovery mechanism: Super chunk-based proficient explore Network (SURF Net). SURF Net divides movies into super chunks and chunks. Unwavering peers that are likely to have longer duration in the system are used to assemble an AVL tree to provide super chunk-level data availability information. Peers storing the equivalent superchunk are grouped into a holder-chain, which is then fond of to a swelling in the AVL tree. This structured superimpose is further extensive by a gossip-based formless network for chunk-level information exchange and data communication. Since the AVL tree consists of only unwavering peers, it provides a trustworthy backbone even in highly dynamic surroundings. Our scrutiny and simulation results show that SURF Net supports nearly-constant and logarithmic look for time for seeking inside a video and jumping to a different video, correspondingly.

Key words—substance discovery, peer-to-peer, video-on stipulate.

I. INTRODUCTION

PEER-TO-PEER (P2P) system has emerged as a promising technology to provide video-on-demand. (VoD) service Several P2P-VoD systems [1]–[3] have been deployed and attracted a large population of viewers. Compared to P2P live streaming, P2P-VoD system supports user interactivity such as VCR operations, which changes user viewing location. As a result, data at different users is highly diverse. Hence, an efficient and fast content discovery algorithm is thus critical to P2P-VoD systems, especially in terms of reducing buffering delay after a seek is performed. Content discovery refers to the mechanism to discover which peers are holding specific data that the peers are interested in. Although content discovery algorithms have been widely discussed in P2P file downloading systems [4]–[9], P2P-VoD introduces different challenges. We identify two unique characteristics of P2P-VoD system, which have strong implications on the design of content discovery algorithm, as follows:

1) real-time constraint: timely content location and delivery to ensure continuous video playback. This is extremely challenging when seek operation is supported;

2) limited local cache: small cache on peers' disk to store video files. Since P2P-VoD system provides watch-when-downloading user experience, users are not willing to cache a large number of files on their local disk. Thus, only a small portion of video is cached (usually less than 1 GB).The first characteristic, the real-time constraint, imposes a stringent requirement on content discovery in terms of delay. Several systems have been designed to meet the

(DHT) to locate a specific chunk. DHT forms a highly distributed and resilient structured overlay. However, in a dynamic environment, it introduces nontrivial search delay since search messages may be lost due to peer churns. PPLive [1] uses gossip method in an unstructured overlay to discover chunks. Peers exchange data availability information periodically with neighbors and select which neighbor to download from based on this information. However, when a hunt for is perform and the requisite data is not obtainable at any neighbors, it may take a long time to find the statistics. Moreover, gossip manner introduces nontrivial slide. PPLive's solution to this is to deploy centralized trackers to index which peers replicate a given movie and only use gossip method for intra-movie information discovery. Although tracker can speed up a search, we do not consider centralized solution as it introduces a single point of failure and may become a bottleneck of the system. The second characteristic inspires us to propose a hybrid method: Super chunk-based efficient search Network (SURF Net). Since data at each peer is quite limited, it is possible to group peers storing the same data together in an appropriate granularity. To this end, video files are first separated into chunks, and then numerous chunks are group into a superchunk. Peers form a structured overlay according to which superchunk they hold. The structured overlay consists of two layers: AVL tree and holder-chain. The AVL tree isconstructed using superchunk ID and formed by stable peers that are supposed to be online for a long time so as to provide a stable backbone for content discovery. Holder-chain is a link list which groups all peers holding the same superchunk together and is attached to a stable peer in the AVL tree. Meanwhile, all peers form an unstructured overlay in which peers periodically exchange chunk-level data availability information with neighbors. Therefore, content discovery involves two steps: superchunk search and chunk search.

II.Proposed System

Our work is different from the existing content discovery schemes for storage-forwarding systems in the following:

1) We integrate a structured overlay with gossip protocol, which can provide efficient and reliable lookup;

2) We take peer lifespan into account and build the backbone of the structured overlay using stable peers;

3) The AVL tree is constructed using superchunk ID and its depth is only decided by the number of superchunks.

Since the number of peers is usually larger than the number of superchunks, lookup in SURFNet is scalable even in very large P2P-VoD systems.

It constructs an AVL tree with stable peers to provide superchunk-level data availability information and aggregates superchunk holders into holder-chains.

III.Related Work

Since P2P-VoD systems provide watch-when-downloadinguser experience, users are not willing to cache the wholevideo. Therefore, it is critical to determine how video content is distributed over peers. In buffer-forwarding systems, a peer caches the video content it has recently played out for a period of time before discarding it. The buffered data is used to serve other peers. Storage-forwarding utilizes static local storage, usually on hard disk, to cache video content. Since local storage is much larger than buffer in memory, peers can cache more data for longer time, which increases the data sharing among peers. Moreover, even when a peer changes its playing point, the old data remains in storage for quite a long time. The downstream peers are not abandoned immediately, which provides sufficient time for them to switch parents, thus improving the robustness of the system. The basic problem of content discovery in buffer- forwarding is how to find peers whose playing points are close to the required data. In tree-based buffer-forwarding systems, such as P2Cast , P2VoD , and Stream , peers are mainly grouped according to their playing points or buffer status. Thus, both data distribution and lookup can be supported by the tree structure. However, mesh-based systems disseminate video data through unstructured overlay, which is inefficient in search. To facilitate the content location in mesh-based buffer-forwarding systems, many structures are proposed to build an index overlay besides the data overlay. Such structures include ring-assisted topology (RINDY) , dynamic skip-list (DSL), and AVL tree.

Content discovery in storage-forwarding is more challenging. As the content is sparsely distributed over the network, peers' cache status cannot be indexed by their playing points. A more efficient index overlay is hence critical for storage-forwarding system to sustain continuous playback and user interactivity. In this paper, we propose a novel superchunk-based structure to provide efficient content discovery. Many schemes have been adopted to provide fast lookup in storage-forwarding systems. Bullet Media uses DHT to build an index overlay to store data availability information in a mesh-based data distribution network. In Bullet Media, an entire video is decomposed into blocks for retrieval and several continuous blocks are grouped into a chunk.

SURFNet adopts an AVL tree for superchunk search, we stress the difference between SURFNet and existing AVL tree-based schemes. Zhou *et al*. [24] proposed a tree assisted overlay for video distribution, in which all peers are organized in an AVL tree according to their buffer status. Chi *et al.* use similar index overlay. However, they suggest that having a small amount of partners is sufficient for media streaming; thus, it is possible to prune the AVL tree so that the number of nodes in the tree can be reduced Specifically, if a peer's buffer range is fully covered by other peers, then it will be removed fromthe AVL tree. These schemes are efficient in buffer-forwarding systems, but are not suitable for storage-forwarding schemes as peers cannot be indexed based only on the playing points. To solve this problem, SURFNet constructs AVL tree using superchunk ID and extends tree with holder-chains to group peers storing the same superchunk together. Moreover, from the point of view of tree maintenance, SURFNet utilizes adjacent nodes for peer replacement and therefore reduces the probability of unbalance caused by peer departure. Our work is different from the existing content discovery schemes for storage-forwarding systems in the following: 1) we integrate a structured overlay with gossip protocol, which can provide efficient and reliable lookup; 2) we take peer lifespan into account and build the backbone of the structured overlay using stable peers; and 3) the AVL tree is constructed using superchunk ID and its depth is only decided by the number of superchunks. Since the number of peers is usually larger than the number of superchunks, lookup in SURFNet is scalable even in very large P2P-VoD systems.

IV.Algorithm Used

A. Peer Join When a peer joins the structured network, it may join the AVL tree or the holder-chain. If it is a stable peer, then it stays in the AVL tree; otherwise, it joins the holder-chain. Hence, the first question is how to predict peers' lifespan online. In P2P systems, peers could join and leave the system at any time; thus, each peer is an unreliable node, there is a high percentage of peers who have a long lifespan. For example, as a result of [1], there are over70% of peers who stay in the system for over 15 min. Another well-known observation is that a node with a higher "age" is likely to stay longer in the future, which implies that, at a certain point in time, for all online peers in the system, the earlier a peer gets online, the more stable it would be in the future. Therefore, the time when peers get online, namely join time, can be used to identify more stable peers. The overlay construction algorithm will show how stable peers can be naturally promoted by SURFNet using join time.

In SURFNet, the content server also serves as a rendezvous point, which maintains the information of the root node of the AVL tree. When a peer starts, it gets the root node from the content server and searches for the requested chunk of data. When becomes a holder of a new superchunk , it registers itself to the structured search network. first contacts the content server to get the root node of AVL tree. If root node does not exist, then registers itself as root node. If root node is returned by the server, then sends a register message containing and 's IP address to the root node. When a stable peer , responsible for superchunk in the AVL tree, receives a register message, the process which takes place is described in Algorithm
1. As the message forwarding is bounded by the AVL
tree, in a system with superchunks, the number of messages
generated during the registration is .

Algorithm 1: process when receives a register message for superchunk originated by peer

Algorithm 2: replacement process when peer is about to leave

V.Conclusion

In this paper, we propose a novel SURFNet to reduce search latency in P2P-VoD systems. It constructs an AVL tree with stable peers to provide superchunk-level data availability information and aggregates superchunk holders into holder-chains. An unstructured overlay is formed to provide chunk-level information exchange and data transmission. Theoretical analysis and simulation results show that SURFNet provides efficient search even under large amount of data and is able to adapt to large-scale systems.

References

[1] Y. Huang, T. Z. Fu, D.-M. Chiu, J. C. Lui, and C. Huang, "Challenges, design and analysis of a large-scale P2P-VoD system," in Proc. ACMSIGCOMM 2008 Conf. Data Communication (SIGCOMM '08), New York, 2008, pp. 375–388.

[2] B. Cheng, L. Stein, H. Jin, and Z. Zhang, "Towards cinematic internet Video-on-Demand," in Proc. 3rd ACM SIGOPS/EuroSys European Conf. Computer Systems 2008 (Eurosys '08), New York, 2008.

[3] PPStream., [Online]. Available: http://www.ppstream.com/.

[4] S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Schenker, "A scalable content-addressable network," SIGCOMM Comput. Commun. Rev., vol. 31, no. 4, pp. 161–172, 2001.

[5] I. Stoica, R. Morris, D. Karger, M. F. Kaashoek, and H. Balakrishnan, "Chord: A scalable peer-to-peer lookup service for internet applications," SIGCOMM Comput. Commun. Rev., vol. 31, no. 4, pp. 149–160,2001.

[6] A. I. T. Rowstron and P. Druschel, "Pastry: Scalable, decentralized object location, and routing for large-scale peer-to-peer systems," in Proc. IFIP/ACM Int. Conf. Distributed Systems Platforms Heidelberg (Middleware '01), London, U.K., 2001, pp. 329–350.

[7] B. Zhao, L. Huang, J. Stribling, S. Rhea, A. Joseph, and J.Kubiatowicz, "Tapestry: A resilient global-scale overlay for service deployment," IEEE J. Select. Areas Commun., vol. 22, no. 1, pp. 41–53, Jan. 2004.

[8] J. Aspnes and G. Shah, "Skip graphs," ACM Trans. Algorithms, vol. 3, no. 4, p. 37, 2007.

[9] H. V. Jagadish, B. C. Ooi, and Q. H. Vu, "BATON: A balanced tree structure for peer-to-peer networks," in Proc. 31st Int. Conf. Very Large Data Bases (VLDB '05), 2005, pp. 661–672.

