A HYBRID PUSH-PULL OVERLAY NETWORK FOR PEER-TO-PEER VIDEO STREAMING

Seyed Mehran Ronaghi[1] and Behzad Akbari[2]

[1]Department of Electrical and Computer Engineering, Qazvin Islamic Azad University, Qazvin, Iran
[2]Department of Electrical and Computer Engineering, Tarbiat Modares University, Tehran, Iran

ABSTRACT

In this paper, we have proposed a hybrid push-pull protocol for peer-to-peer live video streaming. The main goal of this research is to minimize the network end-to-end delay in comparison to pure mesh networks. Hybrid protocols, in most cases, suffer from complex construction and maintenance. Therefore, our proposed protocol uses a pure mesh topology and a single layer video coding. In summary, our push-pull protocol has two parts. The pull-based part which is done on the mesh network, and the push-based part which consists of two phases: parent selection and tree construction. When a push procedure appears, it is very important to prevent data redundancy. To satisfy this condition, we have introduced a parent selection method. In this method, by parent selection based on the minimum arrival time, the most stable node will be selected. This node has the advantage of maximizing the expected service time of the tree. Using this method, there is no need for maintaining any extra information and topology control data. Finally, we do performance evaluation using OMNeT++ simulator. The simulation results show that the proposed architecture has better performance in start-up delay, end-to-end delay, and distortion than pure mesh-based network.

KEYWORDS

Video Streaming, Peer-to-Peer, Overlay Multicasting, Push-Pull, Mesh Structure

1. INTRODUCTION

Video Streaming over the Internet has recently attracted a lot of users. Despite the remarkable progress in network technologies, both in the access layer and the core layer, and providing users with high-speed Internet connection, the development of large-scale video streaming system still faces many challenges. These challenges are posed by the very strict requirements for quality of video streaming service. Video streaming applications require high bandwidth. They are very sensitive to packet delay and loss. One solution for video streaming over the Internet is client-server service model. Using this model for large-scale video streaming applications over the Internet requires a large number of video servers and high bandwidth. In addition, the reliability of the model is a question because the service is dependent on one or several machines. Therefore, the best service model for this type of applications is the multicast service model.

Peer-to-peer networks have recently emerged as a new paradigm to build distributed network applications such as multicasting. In peer-to-peer systems, the Internet hosts organize themselves into an overlay network over the Internet and in the application layer of the Internet, and share their own resources such as processing power, memory and uploading bandwidth. In an overlay network, each peer stores data and forwards them to other peers. In this model, the uploading bandwidth of end users is efficiently utilized to reduce the bandwidth burdens otherwise placed on the servers. However, the deployment of overlay multicasting for video streaming faces many

DOI : 10.5121/ijp2p.2015.6101

challenges. These challenges are posed by the inherent characteristics of overlay networks. Nodes in overlay networks are the Internet hosts. Low uploading bandwidth of many Internet users and heterogeneity of their bandwidth, their departures, and the lack of quality of service guarantee on the Internet, make good quality video delivery in an overlay network challenging.

In this paper, we consider the inherent limitations of overlay networks and positive characteristics of them, and propose a structure to improve the quality of video delivery in peer-to-peer live video streaming systems.

The rest of the paper is organized as follows: in the next section, we explain technical background on peer-to-peer video streaming and related work on hybrid push-pull model. In section 3, proposed method to improve overlay network structure for live video streaming is presented. In section 4, the proposed method is evaluated. Conclusion and future work are presented in section 5.

2. Background and Related Work

Peer-to-peer streaming systems can be broadly classified into two categories based on the overlay network structure: tree-based and mesh-based [1]. Tree-based systems in comparison to mesh-based systems, in terms of data delivery delay are better and optimization of their overlay network structure, in terms of quality of service is less complex. In contrast, construction of resistant tree to peer churn is very challenging. Multi-tree methods are an attempt to solve this problem, but optimization of multi-tree structures in terms of quality of service is very difficult and sometimes impossible.

Unlike tree structures, construction of mesh structures is simple and this type of networks is resistant to peer churn, but searching video chunks in them is time consuming and causes long delay in playing video.

Regarding the characteristics of peer-to-peer overlay network structures, the mesh structure can be the appropriate structure for video streaming. In spite of various advantages, mesh-based systems suffer from some problems. The most important one is delay. The solution proposed to solve this problem is a hybrid structure. Hybrid structures are usually constructed by combining both tree and mesh structures. In these structures, the tree structure is used to push the video contents, and the mesh structure is used to pull the missing video chunks. In other words, the data transmission mechanism is push and pull.

In this paper, we try to exploit the advantages of tree and mesh structures and combine them together and introduce a new hybrid push-pull protocol for live video streaming. The main goal of this design is to minimize the network end-to-end delay and improve the quality of video delivery in comparison to pure mesh networks. In this section, we survey the state of the art push-pull peer-to-peer video streaming systems.

2.1. New Cool Streaming

The New Cool Streaming [2] is the reformed version of the Cool Streaming [3] in which instead of the pull mechanism, the push-pull is deployed.

In the New Cool Streaming, the video is divided into N sub-stream without any coding techniques. These sub-streams contain the video block and all nodes can request each sub-stream separately. The push mechanism in the New Cool Streaming works in the way that: the information about sub-streams and their contents (video blocks) is carried in the nodes' buffer maps that are exchanged between them periodically. Each node based on the missing video block, requests that block within a single pull request. Then, the provider will push the sub-stream that contains that video block to the node.

Overhead reduction and time saving during block transmission from the sender to the receiver are the most significant advantages of the New Cool Streaming protocol.

2.2. PRIME

Peers in PRIME [4] preserve a randomly connected and directed overlay (mesh-shaped overlay). The parent-child relationship governs on PRIME in which the content is delivered from parent to child nodes. PRIME does its job based on two phases: diffusion phase and swarming phase. Diffusion phase: where data rapidly flows away from source, and is followed by a swarming phase. Swarming phase: where peers exchange their packets (through pull approach). In the swarming approach, each overlay node advertises to its neighbours which packets (or blocks of packets) it has received, and the neighbours explicitly request blocks as needed.

Impact of overlay connectivity and source behaviour on the performance of content delivery is examined in this paper. Other key factors, which are examined on the performance of this method, are bandwidth bottleneck, content bottleneck and peer population. In this protocol, increase in loss rate causes rapid drop in the delivered quality for large peer degrees. In addition, the percentage of peers with high quality in a bidirectional overlay is 10%-20% less than the unidirectional overlay over the sweet range of peer degree. Furthermore, bidirectional connections reduce the number of swarming shortcuts among diffusion sub-trees and thus increase the percentage of content bottleneck during the swarming phase.

The main goal of PRIME is to minimize the bandwidth bottleneck by satisfied bandwidth-degree and also to minimize the content bottleneck by utilizing the outgoing bandwidth and deriving a pattern of content delivery. The advantages of the swarming approach are its simplicity (it requires no complex distributed algorithm to build trees) and its robustness (any neighbour can be called upon to contribute blocks of data, so the loss of any given neighbour does not cause a discontinuity in data delivery).

In the swarming method, a basic trade-off between control overhead and delay should be considered.

2.3. mTreebone

The key idea of the proposed method given in mTreebone [5] is to identify a set of stable nodes to construct a tree-based backbone, called treebone, with most of the data being pushed over this backbone. These stable nodes, together with others, are further organized through an auxiliary mesh overlay. In this paper, stable node refers to the nodes with relatively long lifetime. This protocol minimized start-up latency and transmission delay.

The most plus points in the mentioned protocol are low overhead, low delay, high efficiency and robustness. The protocol also avoids data redundancy and reduces data loss in the tree repairing process. In addition, the protocol accommodates node dynamics and uses all the available bandwidth. But due to peer churn, the performance of system decreases. Two other main drawbacks are bandwidth-constrained case and Flash crowd case. The protocol should conduct experiment of larger scale over the Internet.

3. The Proposed Method for Video Streaming

Hybrid protocols, in most cases, suffer from construction complexity and maintenance difficulty. It is obvious that complex construction and maintenance cause high overhead in the control unit. Therefore, our proposed protocol focuses on a pure mesh structure and uses a single layer video coding in its video delivery system. In this method, we use a frame-based video delivery method. In other words, our definition of a video chunk is a video frame. In summary, our push-pull protocol has two parts. The pull-based part which is done on the mesh structure, and the push-based part which consists of two phases. The first phase is to select a parent node in the mesh structure, and the second phase is to construct a dynamic tree over the mesh structure in order to push the most important video frames within it. In the following subsections, we explain the protocol mechanism in detail.

3.1. Pull-based Part

Mesh-based video streaming has a simple algorithm. When a peer joins the network, firstly, it contacts a node so-called tracker that has global information about existing peers in the network. The peer requests a number of neighbours. The tracker sends back the addresses of some peers corresponding to the requested number with their arrival time. Then, the newly joined peer contacts each of them and requests to make a relationship. On the opposite side, if the peer has a free position for the node's request, will accept it; otherwise, will deny it. When the newly joined peer receives the acceptance message, it sends an acknowledgment message to confirm the neighbouring connection between itself and the opposite peer.

After the connection is established between two nodes, they start to periodically exchange the buffer map. The buffer map is a map that shows the available video frames in the node's buffer. Each peer looks at its buffer and finds the first unset frame, which is going to be played. When the neighbours' buffer maps arrive, the peer seeks the unset frame in its neighbours' buffer maps list to find out which of neighbours has the frame. In a simple way, the peer requests the neighbour who has the frame. Sometimes a situation occurs in which more than one neighbour has the foregoing frame. In this case, we consider that the peer randomly selects a neighbour who has the frame. If a neighbour could not send back the requested frame for any reason within a specific time, the peer will request the frame again but from another neighbour. The specific time for requesting and retrieving the frame is defined as Round Trip Time (RTT). If the requested frame does not arrive within a time less than the RTT seconds, the peer will repeat its request.
Mesh structure used in this paper, is similar to what was proposed in Coolstreaming/DONet [3]. This method has a simple scheduling algorithm for requesting the frames from neighbours. The fundamental advantage of this method is that it eliminates the requirement for constructing and maintaining any specific overlay network.

3.2. Push-based Part

To achieve our goal that is deploying the push mechanism on a pure pull-based mesh structure, we introduce a new frame-based scheduling mechanism in our protocol. As mentioned in the previous subsection, we defined each video frame as a chunk in the network. We use the MPEG4-Part1 compression standard and its typical group of picture (GoP) structure. In this structure with size of 12, there exists one I-frame, three P-frames and eight B-frames.

I-frames are just still pictures coded using a variant of JPEG, also using full resolution luminance and half-resolution chrominance along each axis. To code these frames, motion estimation and compensation techniques are not used. In other words, these frames are not dependent on other frames. P-frames, in contrast, code inter-frame differences. They are predicted from their previous I-frames or P-frames. B-frames are similar to P-frames, except that they are bidirectional and interpolated from their previous and next I-frames or P-frames.

By considering these principles, if the I-frame of a GoP is lost, then all the received frames of corresponding GoP will be lost. If P-frames of a GoP are also lost, the following correctly received frames will be lost. Due to importance of I-frames and P-frames, in proposed method, we push I-frames and P-frames on the mesh network.

When a push procedure appears, it is very important to prevent data redundancy. To satisfy this condition, we introduce a parent selection method. In this method, the most stable node will be selected as the parent in order to maximize the expected service time of the tree. Recent studies have found that, in overlay multicast systems, nodes already with higher ages tend to stay longer [6]. Hence, a node's age partially reflects its stability.

Previous studies on video client behaviour have suggested that node duration generally follows a heavy-tailed distribution [7-9], in particular, the Pareto distribution. In other words, the age threshold for a node arriving at time t is 30% of the residual session length [5].

Therefore, the node with highest age will be selected as the parent. In other words, among the peer's neighbours, the neighbour who has the minimum arrival time is the best candidate to be chosen as the parent node.

Hence, when a peer wants to select the parent, it compares its neighbours' arrival time and chooses the neighbour with the minimum arrival time among the neighbours. Then the child sends the push request to the parent. On the other hand, when a node receives a push request, if it has free position to accept a child, it will accept; otherwise, it will deny informing the child that it has no available seat for the request sender.

As we select only one parent for each child, the data redundancy will be controlled in a true manner. In addition, by parent selection based on minimum arrival time, the most stable node will be selected. The most stable node has the advantage of maximizing the expected service time of the tree. Using the parent selection method causes construction of a dynamic tree over the mesh structure without maintaining additional structure.

After the parent selection process, when a parent node receives I-frame or P-frame, it will push that frame to its children. On the other hand, in order to prevent data redundancy, if a peer has a parent, it does not ask I-frames and P-frames. If the frame is not pushed and retrieved before RTT seconds to its playback time, it will be requested by the child node.

4. PERFORMANCE EVALUATION

We use OMNeT++ [10] as the platform, which is a modular and discrete event simulator for simulating various communication networks. We also use the INET framework [11] to simulate the Internet in OMNeT++. The INET framework implements physical, data link, network, and transport layer protocols in OMNeT++. We use OverSim framework [12] to construct the overlay network in OMNeT++. OverSim is a reusable framework in OMNeT++ for simulating overlay networks. OverSim uses INET framework for simulating underlay layers. Table 1 shows the simulation parameters.

In this section to quantify the QoS, we define some metrics and carry out an analysis of simulation results based on these metrics. These metrics are start-up delay, end-to-end delay, and distortion.

Table 1. Simulation parameters.

Parameter	Value
Maximum packet size	1000 Bytes
Peer-side buffer	40 seconds
Buffer map exchange period	1 second
Video codec	MPEG4-Part1
Video frame per second	25 frames
Number of frames in a GoP	12 frames
Number of B-frames between frames	2 frames
Selected trace file	StarWarsIV
Average video bit rate	512 Kbps
Number of neighbors	Random(3,5)
Node's bandwidth	Random(764Kbps,1.5Mbps)
Video chunk size	1 frame
Simulation duration	200 seconds

4.1. Start-up Delay

The time between connecting to the mesh and starting video playback is called start-up delay. It is a random variable and depends directly on start-up buffering and the network delay. Start-up buffering is time of video that is needed to be buffered before starting playing. We do all the simulations with start-up buffering 8 seconds.

Figure 1 shows average start-up delay in proposed method compared with pure mesh-based network. As depicted in Figure 1, the proposed method has less start-up delay compared with pure mesh-based network.

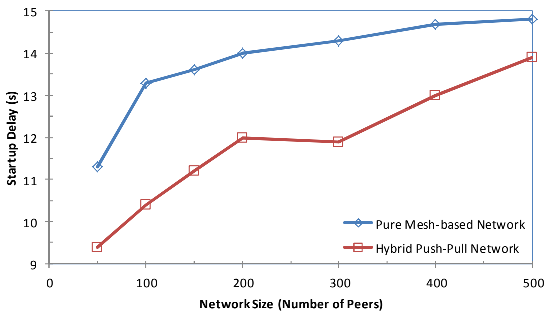

Figure 1. Average start-up delay in proposed method compared with pure mesh-based network

4.2. End-to-End Delay

The time between creating a frame in the source node (server) and playing it in the destination node is defined as end-to-end delay. Figure 2 shows average end-to-end delay in proposed method compared with pure mash-based network. As depicted in Figure 2, the delay produced in the proposed method is lower than the pure mesh-based network because the push method reduces the number of request/response messaging procedures; hence, the end-to-end delay decreases.

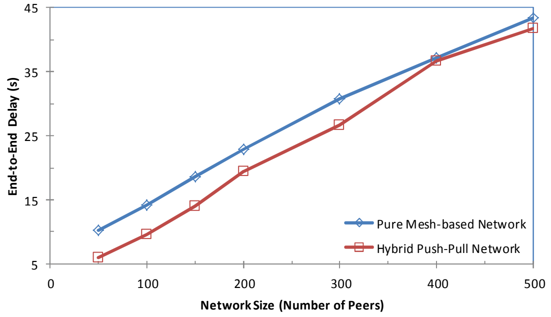

Figure 2. Average end-to-end delay in proposed method compared with pure mesh-based network

4.3. Video Distortion

Video degradation due to video content loss is defined as video distortion. This metric can be obtained from the following formula.

$$\text{Distortion}=\left(1-\frac{\text{Total number of received frames}}{\text{Total number of frames}}\right)\times 100$$

A frame may be lost due to network congestion, error, delay, no dependent frames in video player.

As depicted in Figure 3, video distortion in proposed method is less than pure mesh-based network. The reason is that the push method prevents late arrival loss of frames, which causes the frame to be useless for playing.

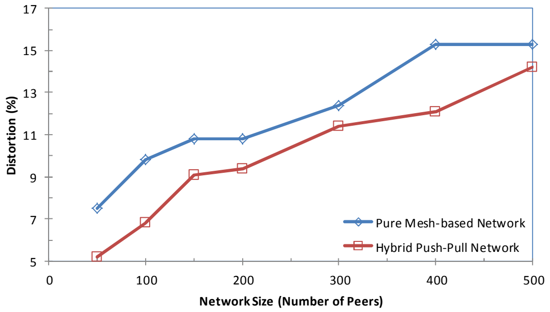

Figure 3 shows average video distortion in proposed method compared with pure mesh-based network.

5. Conclusion and Future Work

In this paper, we proposed a hybrid structure in which a part of video stream in tree and other parts in mesh network are distributed among the members of a multicasting group in order to improve the quality of video delivery in peer-to-peer live video streaming systems. In summary, our push-pull protocol has two parts. The pull-based part which is done on the mesh network, and the push-based part which consists of two phases. The first phase is to select a parent node in the mesh topology, and the second phase is to construct a dynamic tree in order to push the most important video frames within it. Finally, we did performance evaluation of proposed method in comparison to pure mesh-based network using OMNeT++ simulator. The simulation results showed that the proposed method has better performance in start-up delay, end-to-end delay, and distortion than pure mesh-based network.

Our suggestions for future work are using resource allocation techniques in nodes, in particular, parent nodes to achieve more scalability and resistance, also packet loss recovery in overlay connections, and adding incentive mechanisms.

References

[1] Liu Y., Guo Y., and Liang C., “A survey on peer-to-peer video streaming systems”, Peer-to-Peer Networking and Applications, vol. 1, no. 1, pp. 18-28, 2008.

[2] Li B., Xie S., Qu Y., Keung G. Y., Lin C., Liu J., and Zhang X., “Inside the new coolstreaming: principles, measurements and performance implications”, IEEE INFOCOM proc., pp. 1031–1039, 2008.

[3] Zhang X., Liu J., Li B., and Yum T.-S. P., “CoolStreaming/DONet: A Data-driven Overlay Network for Peer-to-Peer Live Media Streaming”, IEEE INFOCOM proc., pp. 2102-2111, 2005.

[4] Magharei N., and Rejaie R., “Prime: peer-to-peer receiver-driven mesh-based streaming”, IEEE/ACM Transactions on Networking, vol. 17, no. 4, pp. 1052–1065, 2009.

[5] Wang F., Xiong Y., Liu J., “mTreebone: A Collaborative Tree-Mesh Overlay Network for Multicast Video Streaming”, IEEE Transactions on Parallel and Distributed Systems, vol. 21, no. 3, pp. 379–392, 2010.

[6] Bishop M., Rao S., and Sripanidkulchai K., “Considering priority in overlay multicast protocols under heterogeneous environments”, IEEE INFOCOM proc., 2006.

[7] Almeroth K. C., and Ammar M. H., “Collecting and modelling the join/leave behaviour of multicast group members in the mbone”, IEEE International Symposium on High Performance Distributed Computing (HPDC), 1996.

[8] Sripanidkulchai K., Maggs B., and Zhang H., “An analysis of live streaming workloads on the Internet”, Internet Measurement Conference, 2004.

[9] Wang F., and Liu J., “A trace-based analysis of packet flows in data-driven overlay networks”, Technical report, 2006.

[10] OMNeT++, Homepage: http://www.omnetpp.org/.

[11] INET, Homepage: http://inet.omnetpp.org/.

[12] OverSim, Homepage: http://www.oversim.org/.

Authors

Seyed Mehran Ronaghi is holderof Master’sdegree course of studies (MScE) in the field ofInformation Technology (IT) with specialization in Computer Communication Networks from Qazvin Islamic Azad University (QIAU) in Iran. He has received his Bachelor’s degree course of studies (BScE) in the field of Computer Engineering with specialization in Computer Software in 2006. He is member of Iran National Scientific Student’s Organization on Electrical Engineering (NSSOEE) at Sharif University of Technology and superior arbiter of 16th and 17th Iranian Student Conference on Electrical Engineering (ISCEE). His research interests are Distributed Systems, Application Layer Multicast (ALM), Multimedia Streaming over the Internet, and Peer-to-Peer (P2P) Systems.

Dr. Behzad Akbari received his B.Sc., M.Sc., and Ph.D. degrees in Computer Engineering from Sharif University of Technology, Tehran, Iran, in 1999, 2002 and 2008, respectively. He is currently an assistant professor in the Department of Electrical and Computer Engineering, Tarbiat Modares University, Tehran, Iran. His research interests are Computer Networks, Multimedia Streaming over the Internet, Overlay and Peer-to-Peer Networking, Peer-to-Peer Video Streaming, Network QoS, Network Performance Evaluation, and Network Security.