IJACT 14-1-2

A Stream based Patching for True VoD

Sun Jin Oh[1]

[1]Department of Computer & Information Science, Semyung University, Korea
sjoh@semyung.ac.kr

Abstract

In this paper, we propose a stream based patching technique in order to reduce demands of network output bandwidth required to provide the true video-on-demand (VoD) service in a Multimedia server. The proposed scheme calculates required streams to clients accurately by analyzing temporal relationships between already arrived requests in a multimedia server and still progressing streams, and then transmits streams freshly. Since our stream based patching technique induces new linking stream which is playing a link role between long and short patching streams, it reduces demand of network output bandwidth required to provide the true VoD service in a multimedia server. Accordingly, we know that the proposed scheme improves service latency and defection rate than those of existing patching technique by a simulation study.

Keywords: *Stream based Patching Scheme, True VOD Service, Multimedia Server.*

1. Introduction

Video on Demand service, a streaming service of multimedia contents for the client's demand, is the most popular multimedia service among real time multimedia application through Internet nowadays. It is because the VoD service can provide the multimedia contents to the clients through Internet in any place and at any time. In a traditional client/server architecture based VoD system, a client requests video to a multimedia server. Then, a server accesses video data stored in the storage, and transmits video stream to a client using its network bandwidth. It is hard to service the clients' requests promptly if many clients' requests come together asynchronously because network bandwidth of a VoD server is limited. Therefore, new technique that can service more clients by reducing the resource requirement of a server is studied actively. In order to increase the number of clients who can be serviced with limited network bandwidth of a server, multicast technique such as Patching [1] and Double Patching [2] is proposed. The unicast technique transmits one video stream to only one client at the same time. But, the multicast technique uses network output bandwidth of a server efficiently by sharing one video stream to several clients.

A patching is a multicast technique that sends only the leading portion of video data for the client's request and shares transmitting video stream cached already in a local buffer in order to reduce the network output bandwidth of a server for the true VoD service. Double patching scheme [2] makes the long patching stream to include not only the essential data for current client but also the extra data for the service of future client's request. Consequently, a multimedia server can reduce fairly big amount of transmitted total video data by sharing these extra data with future clients.

In this paper, we propose stream based patching scheme that can reduce the bandwidth requirements of a server to provide the true VoD service by transmitting linking stream after sending L-stream according to the

Manuscript Received: Apr. 7, 2014 / Revised: May. 13, 2014 / Accepted: Jun. 6, 2014
Corresponding Author: sjoh@semyung.ac.kr
Tel:+82-43-649-179, Fax: +82-43-649-1747
Dept. of Computer & Information Science, Semyung University, Korea

arrived requests in a multimedia server. From the simulation study, we know that proposed stream based patching scheme always reduces service latency, client's defection rate, and amount of downloading data relative to the existing patching techniques.

The remainder of this paper is organized as follows. In section 2, we describe the related works for true VoD services. In section 3, we propose the stream based patching scheme proposed in this paper. In section 4, we evaluate the performance of proposed stream based patching scheme by a simulation study and compared with the existing patching technique, and finally concludes the paper in section 5.

2. Related Works

Recently, several techniques which shares server bandwidth among clients are proposed. These techniques are using limited network input and output bandwidth of a VoD server efficiently to provide the service to more clients. Pyramid broadcasting [3], skyscraper broadcasting [4], and Pagoda broadcasting [5] are techniques which broadcast same video repeatedly for the large scale video service. These techniques divide single video by different sized n frames, and broadcast repeatedly through n assigned channels among channels which are logically divided from server bandwidth. By setting the size of the first frame to the smallest, and then making the size of next frame bigger, it confines the client service latency into the size of the first frame in the worst case regardless of the number of video requests. But, these broadcasting techniques can be applied to the most popular videos in expenses. In order to maintain the service latency for all videos regardless of their popularities, service should be scheduled directly when the request is issued by a client [2].

The unicast technique, transmitting each video stream whenever the client requests video, is simple without any service latency, but it does not use the network output bandwidth of a server efficiently. Since entire data of the video are transmitted for each request of video, if the number of client's requests are rapidly increased all network output bandwidth of a server will be occupied. So the available bandwidth of a server is hardly found and a VoD server suffers significant bottleneck. In order to improve this problem in the unicast technique, multicast techniques sharing single video stream with many clients, such as batching [6], piggybacking [7], patching [1], optimal patching [8], scaling patching [5], and channel-reservation patching [9] are proposed.

Batching [6] multicasts single stream of the requests for the same video by gathering the requests in some period of time. Batching can share network bandwidth efficiently, but it cannot support true VoD service because service latency is occurred until starting video playback after service request.

Patching [1] enables a new client to share streaming video data to service the client's video request immediately. In systems similar to patching[1, 2, 8, 9], we assume that a client has maximum two downloadable streams, two loaders to store downloaded stream in a buffer, and a video player plays video data continuously.

A server sends entire video data to the client through regular stream (R-stream) when it receives a video requests from a client. After then, a server investigates the skew, which is the interval between starting time of the latest R-stream and the time of new request, whenever the new request for the same video arrives. If skew to the R-stream exceeds particular temporal threshold called patching window, a server sends new R-stream to a new client. Otherwise, a server makes a new client to share recent R-stream, and sends patching stream (P-stream), the leading portion of the video data, through a new channel. Therefore, patching supports the true VoD service by transmitting P-stream and R-stream as soon as the client's request arrives.

Furthermore, patching uses bandwidth efficiently by sharing it with many clients through multicasting transmitted R-stream. Unfortunately, in patching, P-stream is getting longer as the skew of recent R-stream and new request is longer. In order to shorten the P-stream, double patching induce additional long patching stream (L-stream) to share not only R-stream but also P-stream by a client. It uses temporal thresholds called multicast window and patching window. Multicast window is the minimum interval of consecutive two R-streams, and patching window is the minimum interval of consecutive two L-streams.

3. A Stream based Patching Scheme for True VoD

A Multicast technique similar to patching works only for the requests of the same video. So, patching technique is applied for each video if the number of video is increased. Then, total multimedia server bandwidth can be determined by calculating network bandwidth requirement needed to provide true VoD service for each video individually. Therefore, we assume that each multimedia server has only one video to simplify the problem as in double patching.

The L-stream is an important stream in the double patching. L-stream should include leading portion of data to cover the skew of R-stream for the current request, and we call it as essential data. But, L-stream includes not only essential data but also extra data which is video data to be played for some period. This extra data is video data needed to share L-stream with a new client in a future, and 2 is the maximum number of streams downloadable by one client in the patching system spontaneously. Extra data of L-stream has also following problems. After beginning L-stream transmission, some of the ending portion of extra data in L-stream is useless unless latest arrived request in a L-stream patching window is arrived exactly at the end of the patching window. In the worst case, all extra data will waste server network output bandwidth uselessly if no requests are arrived in L-stream patching window. The waste of extra data always occurs if any new requests are arrived at the end of L-stream patching window, and a client downloading L-stream always has to download extra data even though L-stream skew is less than threshold. Therefore, we can schedule accurate length of stream based on the server request arrived already without considering the future request of uncertain arrival time. We also propose the multicast technique eliminating client's unnecessary data download. Proposed technique reduces the output bandwidth requirement of a server to support true VoD service by eliminating server's unnecessary data transmission which can be occurred in the double patching technique. Server network output bandwidth also reduces the average service latency of the double patching and client's defection rate in the same circumstances.

The proposed stream based patching scheme uses 4 streams; regular stream (R-stream), long patching stream (L-stream), short patching stream (S-stream) and linking stream (K-stream). In a proposed stream based patching scheme, however, L-stream includes essential data within skew of recent R-stream. So, a client sharing this L-stream only can download video data for two times of skew between S-stream and L-stream during the time from the end of L-stream to the beginning of R-stream they can download. Linking stream (K-stream) is the stream that includes data un-receivable by a client who is serviced by S-stream while sharing L-stream. This K-stream plays a linking role between L-stream and R-stream downloadable by a client in order to playback consecutive video data. So, a client plays video data in the order of S-stream, L-stream, K-stream and R-stream if client has to download these streams. In this stream based patching scheme, VoD server can multicast single L-stream to share with all clients serviced by S-stream to share L-stream without unicasting K-streams individually.

The start time of K-stream needed for all requests is the end time of L-stream transmission, and the beginning data included in a stream is same even though the length of stream needed for each request is different. Therefore, a server can calculate the longest K-stream accurately using the arrival time of the latest request as soon as completing L-stream transmission.

If a client plays consecutive video, it has to be guaranteed that transmission of K-stream begins as soon as completing L-stream transmission. It can be guaranteed by allocating the channel used to send L-stream to the K-stream when the server finishes L-stream transmission. The K-stream determines to send video data accurately as soon as completing L-stream transmission, and multicasts to all clients sharing L-stream through channel used in L-stream transmission. Figure 1 shows the stream based patching scheme proposed in this paper.

4. Performance Evaluation

In this section we evaluate the performance of proposed stream based patching scheme based on the simulation study by calculating server bandwidth requirement presented by Cai[2]. This model calculates average bandwidth requirement needed to provide true VoD service when a VoD server has single video, and client requests are in poisson distribution. In order to compare the performance of the proposed stream based

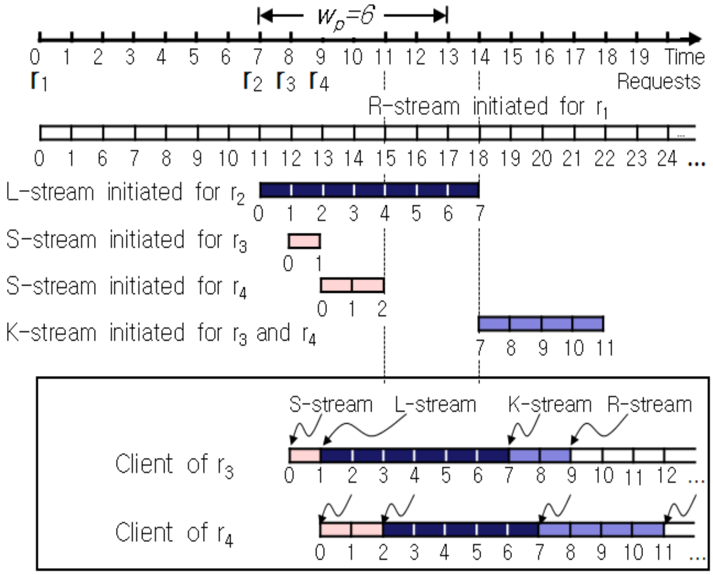
w_p=6
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 Time
r_1
r_2 r_3 r_4
Requests
R-stream initiated for r_1
0 1 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 ...
L-stream initiated for r_2
0 1 2 3 4 5 6 7
S-stream initiated for r_3
0 1
S-stream initiated for r_4
0 1 2
K-stream initiated for r_3 and r_4
7 8 9 10 11
S-stream L-stream K-stream R-stream
Client of r_3
0 1 2 3 4 5 6 7 8 9 10 11 12 ...
Client of r_4
0 1 2 3 4 5 6 7 8 9 10 11 ...

Figure 1.The Stream based Patching Scheme

patching scheme (4SP) with double patching (DP), we evaluate average bandwidth requirement of a server needed to support true VoD service. Table 1 below shows parameters used in the performance evaluation.

Table 1. Simulation parameters

Parameter	Range of values
Video length	10~90 minutes
Client buffer size	5～25 minutes
Mean request inter-arrival time	5～80 seconds
Video playback rate	1.5 Mbps

A simulation study is performed in terms of defection rate and mean service latency of client's requests according to the video request arrival rate. We assume that the length of video 90 minutes, a client is watching video sequentially until the end, client's defection time is random from 10 seconds to 90 seconds.

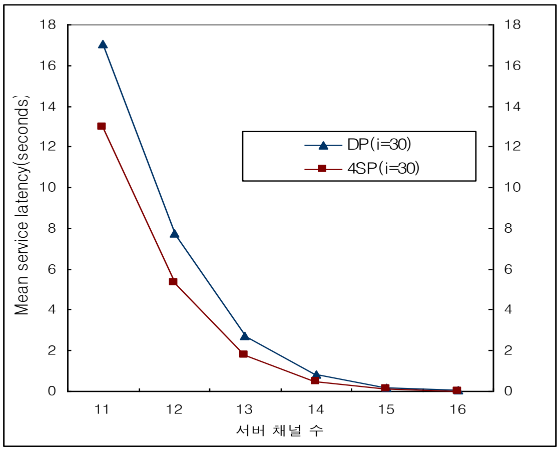

Figure 2 A Service Latency over Server Channels

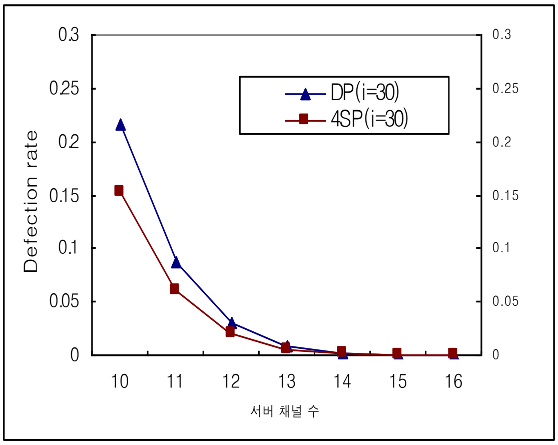

Figure 3 Defection Rate over Server Channels

Figure 2 and Figure 3 show service latency and client's defection rate over the request arrival rate. The anticipated number of server channels for the double patching scheme is 12. According to the simulation result, however, 12 server channels cannot provide true VoD service to all clients as shown in the figures. Double patching can service 63,390 clients and our stream based patching scheme can service 72,275 clients without latency. So we know that our stream based patching scheme can improve its performance by 12.3%. From the results of the simulation study, we know that our stream based patching scheme is much superior more than double patching technique in the same circumstances. Especially, our stream based patching scheme is much superior when the average arrival interval for requests is large.

5. Conclusion

In order to service more clients with limited network output bandwidth of a VoD server, we propose a multicast technique sharing streams. Double patching scheme improves its performance by introducing L-stream that includes extra data for future requests. In this paper, we propose a stream based patching scheme, analyzing requests arrived within patching window of L-stream and scheduling K-stream by calculating required streams for the requests accurately. Proposed stream based patching scheme eliminates server bandwidth wastes those can be generated by extra data of double patching thoroughly. Consequently, it reduces the server network output bandwidth requirement to provide true VoD service. We evaluate the performance of our stream based patching scheme by a simulation study in terms of client's defection rate and service latency, and compared with double patching. From the result of simulation, we know that both client's defection rate and service latency are greatly reduced compared with double patching. Moreover, we know that the number of clients providing true VoD service is increased with decreasing service latency.

References

[1] C. C. Aggarwal, J. L. Wolf, and P. S. Yu, "A permutation-based Pyramid Broadcasting Scheme for Video-on-Demand systems," Proc. of International Conference on Multimedia Computing and Systems, pp. 118-126, 1996.

[2] J. Ying Cai, Wallapak Tavanapong, Kien A. Hua, "A Double Patching Technique for Efficient Bandwidth Sharing in Video-on-Demand Systems," Journal of Multimedia Applications and Tools, Vol. 32, No. 1, pp. 115-136, 2007.

[3] K. Hua, Y. Cai, and S. Sheu, “Patching: A Multicast Technique for True Video-on- Demand Services,” Proc. of ACM Multimedia, pp. 191-200, 1998.

[4] J. F. Paris, S. W. Carter, and D. D. E. Long, "Efficient Broadcasting Protocols for Video on Demand," Proc. of SPIE’s Conference on Multimedia Computing and Networking (MMCN’99), pp. 317-326, 1999.

[5] H. Y. Lee, S. J. Oh, S. J. Ha, I. H. Bae, "Design and Evaluation of a Scale Patching Technique for VOD Servers," LNAI 3214, pp. 219-226, 2004.

[6] Minkyong Kim and Brian Noble, Mobile Network Estimation, In ACM Conference on Mobile Computing and Networking, pp. 298-309, July 2011.

[7] S. J Philip, C. Qiao, Hierarchical Grid Location Management for Large Wireless Ad hoc Networks, MobiHoc '03, pp 33-34, 2003.

[8] Y. Cai, K. Hua, and K. Vu, "Optimizing Patching Performance," In Proc. SPIE/ACM Conference on Multimedia Computing and Networking, pp. 204-215, 1999.

[9] S. J. Ha, I. H. Bae, "Determination of the reservation channel capacity for popular videos in VoD system," Vol. 30, No. 5, pp. 223-231, 2003.