A Segment-Alignment Interactive Broadcasting Scheme

Hung-Chang Yang[1], Hsiang-Fu Yu[1], Li-Ming Tseng[1], Yi-Ming Chen[2]
cyht@dslab.csie.ncu.edu.tw, yu@dslab.csie.ncu.edu.tw, tsenglm@csie.ncu.edu.tw, cym@im.mgt.ncu.edu.tw
[1]Department of Computer Science and Information Engineering, National Central University, Jung-Li, Taiwan 32054, R.O.C.
[2]Department of Information Management, National Central University, Jung-Li, Taiwan 32054, R.O.C.

ABSTRACT

With the growth of broadband networks, Video-on-Demand (VoD) has become realistic. Many significant broadcasting schemes have been proposed to reduce the bandwidth requirement for stored popular videos, but they cannot be used to support VCR functionality perfectly. Herein, we propose a new broadcasting scheme, called the *Segment-Alignment Interactive Broadcasting* (*SAIB*) scheme, which can be built on the top of any segment based broadcasting scheme and supports VCR functionality perfectly. From our analysis, we find that our SAIB scheme is suitable for smooth fast forward and forward jump playback. A simulation is employed to evaluate several segment based broadcasting schemes: FB, PB, RFS and NPB. The results reveal the FB scheme has the lowest blocking probability and additional bandwidth overhead. On the contrary, the NPB scheme has the highest results. These results are due to the size and broadcasting period of segments are different for each segment based broadcasting scheme.

Keywords: hot video broadcasting, video streaming, video on demand service, VCR functionality

1. Introduction

With the growth of broadband networks, Video-on-Demand has become realistic. Many studies have investigated VoD. One of the important areas of research is the issue of how to distribute the top ten or twenty so-called "hot" videos more efficiently. One category is the batching broadcasting [2]. Its advantages are easy to realize the VCR functionality and have the shorter startup waiting time at client end. But the bandwidth requirement is still huge when a lot of clients use the system simultaneously. Another category is the segment based broadcasting [2]. It transfers each video according to a fixed schedule and consumes a constant bandwidth regardless of the number of requests for that video. That is, the number of clients watching a given video is independent of their bandwidth requirements. The segment based broadcasting substantially reduces the bandwidth requirement for hot video, but it cannot provide VCR functionality perfectly.

In order to using segment based broadcasting to reduce the bandwidth requirement and providing VCR functionality perfectly like batching broadcasting, there are some problems must be solved as following:

1. Due to the characteristic of broadcasting, how to support VCR functionality without affecting other clients?
2. How to support fast forward function when the posterior segments are not in the client's disk of set-top-box (STB)? The system cannot predict the position which client executes forward jump function, how to solve this problem?
3. When the client uses pause or stop function, the system still transfers the video data, what the client must do? And further, if the fast rewind function is requested, what the client must do?

Our object is to propose an interactive broadcasting scheme that can be applied to any segment based broadcasting scheme and support VCR functionality perfectly. The client can use any VCR functionality, such as fast forward, fast rewind, forward jump, pause and stop playback at any time.

2. Related Work

2.1 Hot video broadcasting

Steven W. Carter [2] classifies the hot video broadcasting into two categories: batching broadcasting and segment based broadcasting. Both of them suppose a STB at the client end that can buffer portions of the playing video on disk. The characteristics of them are described following.

There are many available related batching broadcasting schemes, such as Patching [9] of K. A. Hua, the Stream Tapping [2] of Steven W. Carter and the Hierarchical Multicast Stream Merging [5] of D. L. Eager and M. K. Vernon, etc. The characteristic of this category is: The new dient is allowed to tap into data streams originally created on the video server for other clients, and then store the data until they are needed. In the best case, clients can get most of their data from existing data streams, which greatly reduce the duration of their own stream.

The related segment based broadcasting scheme are Pyramid Broadcasting [4] scheme, Fast Broadcasting (FB) scheme, Pagoda Broadcasting (PB) scheme [12], Recursive Frequency Splitting (RFS) scheme [18], New Pagoda Broadcasting (NPB) [12], Adaptive Live Broadcasting (ALB) scheme [20], Staircase Broadcasting scheme [6], and Harmonic Broadcasting (HB) scheme [7], etc. The characteristic of this category is: These schemes divide a video into multiple equal-size segments and distribute these segments through several independent data streams. In addition, they require STB to receive all the segments from the data streams when the client starts to watch the video.

2.2 Broadcasting schemes with VCR functionality

The technique of batching broadcasting schemes with VCR functionality relies on switching among multicast streams and cooperating with the unicast stream to provide interactive functionality, such as the Batching Broadcasting [1] in the early stage, the MVoD System with VCR Functionality [15] proposed by Wing-Fai Poon and Kwok-Tung Lo, Single-Rate Multicast Double-Rate Unicast Scheme (SRMDRU) [16] and Interactive Stream Tapping [3] proposed by Paris. The drawback of this category is its high cost. It needs huge bandwidth to achieve the shorter startup waiting time when there are many clients at the same time.

The segment based broadcasting schemes store all of the independent data streams. When client requests fast forward or forward jump function, the system creates unicast stream or multicast stream to provide VCR functionality. Such as VCR Functions in Staggered Broadcasting [10] of Jin B. Kwon and Heon Y. Yeom, the IVoD System [11] proposed by Yiu-Wing Leung and Tony K. C. Chan, Interactive Pagoda Broadcasting (IPB) [14], Broadcasting-Based Interaction Technique (BIT) [17] and Cost-Effective Interactive Broadcasting (CEB) [19]. In this category, they don't require the huge bandwidth requirement as batching broadcasting, but they cannot provide VCR functionality perfectly. That is, they only support forward jump function and fast forward function based on segment but not based on frame.

To aim at the drawbacks of above two broadcasting, we propose a scheme which can support fast forward function based on frame like the batching broadcasting and don't require huge bandwidth like the segment based broadcasting.

3. Segment-Alignment Interactive Broadcasting Scheme

The segment-alignment interactive broadcasting (SAIB) scheme is based on segment based broadcasting, and assumes the client has enough disk space to store whole video data. So it doesn't need additional bandwidth requirement to use pause, stop and fast rewind functionality. In order to reduce the additional bandwidth requirement to execute fast forward and forward jump function, the system creates a multicast stream to transfer the unavailable video data. The client receives the data stream in the disk of STB so that the fast forward function and forward jump function can be smooth playback. We'll clearly discuss how to operate the pause, stop, fast rewind, fast forward and forward jump functionality following.

3.1 VCR Controls

Herein, we assume there is enough disk space to store whole video data at client end, and it cannot cause latency when executing VCR functionality.

Pause:

The pause function is to stop playing video temporarily and will return to play video later. Since the client has enough disk space to store whole video data. When the client executes pause function, the video is stopped playing but the client still receives video data from network.

Fast rewind:

The fast rewind function is to use 2 times playback rate of normal playback to play video reversely. Since the previous video data has been received in the disk. When the client executes fast rewind function, the video is played reversely. At the same time, the client still receives video data from network.

Stop:

The stop function is to stop playing video and never return to play the video later. When the client executes stop function, the video is stopped playing and the client stop receiving video data from network.

Fast forward:

The fast forward function is to use 2 times playback rate of normal playback to play video. Since the posterior video data may not been received in the disk currently and will not been received on time. When the client executes fast forward function, the system will check what are the unavailable video data. Then it creates a multicast data stream to transfer the unavailable video data whose playback rate is 2 times of normal playback rate. The client uses 2 times of normal playback rate to play video and receives video data from network concurrently.

Forward jump:

The forward jump function is to move the playback point to the posterior position designated by client and continue to play video. When the client executes the forward jump function, the system will check what are the unavailable video data and creates a multicast data stream to transfer the unavailable video data. The video is moved to the posterior position designated by client and played. At the same time, the client still receives video data from network.

3.2 Segment-Alignment Interactive Broadcast -ing Scheme

To support fast forward function based on frame and forward jump function, the system must check what are the unavailable video data and creates a multicast data stream to transfer the unavailable video data. If the shared probability of the additional data stream can be increased, the network bandwidth requirement will be reduced further. Herein, we propose a method to achieve this objective. At first, we align the segments that are needed to execute fast forward or forward jump function to the boundary of scheduled segments of some segment based broadcasting scheme. Then we can know what are the lacked segments according to the transmission schedule of that scheme. Following are the introduction of how to align the segments.

3.2.1 Fast Forward

Some assumptions are described following: The length of segment is L seconds, and we define the length of a time slot is L seconds. The length of beginning position of executing fast forward function to the boundary of next time slot is X seconds. The playback rate is 2 times of normal playback rate. Further, we divide all time slots into 2 sections. According to the beginning and ending time of executing fast forward function, we do some discussions following.

Assume the beginning time of executing fast forward function belongs to the first one section of some time slot. We play the remainder portion of the segment belongs to the time slot and the whole portion of the segment belongs to the next time slot in X seconds by using some appropriate playback rate M. The appropriate playback rate M can be computed by equation $\frac{X+L}{X}$. If the value of X is close to L, then the value of M is close to $\frac{L+L}{L}$=2. If the value of X is close to $\frac{L}{2}$, then the value of M is close to $\frac{\frac{L}{2}+L}{\frac{L}{2}}$=3. Thus the playback rate of the first two segments is between 2 and 3, and the playback rate of later segments is 2. That is, the later segments can be played by the playback rate designated by client. The figure 1 shows the situation.

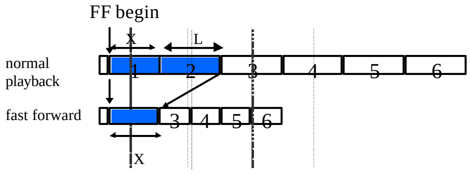

Fig. 1. The beginning time of executing fast forward function belongs to the first one section of some time slot.

Assume the beginning time of executing fast forward function belongs to the second one section of some time slot. We play the remainder portion of the segment belongs to the time slot and the whole portion of the segment belongs to the next time slot in $X+\frac{L}{2}$ seconds by using some appropriate playback rate M. The appropriate playback rate M can be computed by equation $\frac{X+L}{X+\frac{L}{2}}$. If the value of X is close to $\frac{L}{2}$, then the value of M is close to $\frac{\frac{L}{2}+L}{\frac{L}{2}+\frac{L}{2}}$=1.5. If the value of X is close to 0, then the value of M is close to $\frac{0+L}{0+\frac{L}{2}}$=2. Thus the playback rate of the first two segments is between 1.5 and 2, and the playback rate of later segments is 2. That is, the later segments can be played by the playback rate designated by client. The figure 2 shows the situation.

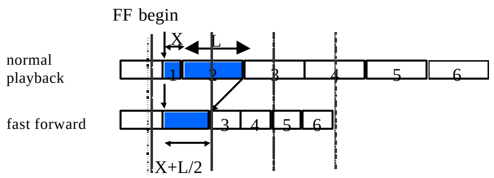

Fig. 2. The beginning time of executing fast forward function belongs to the second one section of some time slot.

Assume the ending time of executing fast forward function belongs to the first one section of some time slot. We play the segment that belongs to the first one section of that time slot by normal playback rate in the remainder time of that time slot. The later segments are played by normal playback rate in the following time slots. Thus the client may watch some redundant video data of the segment belongs to the first one section of that time slot. Figure 3 shows the situation.

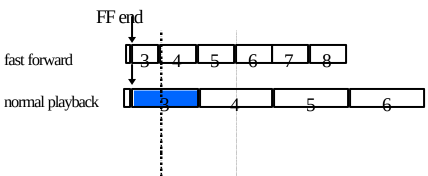

Fig. 3. The ending time of executing fast forward function belongs to the first one section of some time slot.

Assume the ending time of executing fast forward function belongs to the second one section of some time slot. We play the segment that belongs to the second one section of that time slot by normal playback rate in the remainder time of that time slot. The later segments are played by normal playback rate in the following time slots. Thus some video data of the segment belongs to the second one section of that time slot the client may lose to watch. Figure 4 shows the situation.

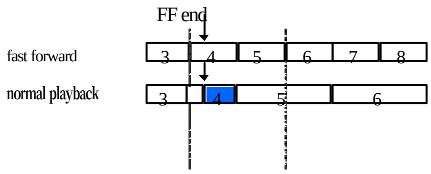

Fig. 4. The ending time of executing fast forward function belongs to the second one section of some time slot.

3.2.2 Fast Forward Example

Herein, we give an example of executing fast forward function, and use the FB scheme as the based broadcasting scheme. Figure 5 depicts the situation that the beginning time of executing fast forward function belongs to the first one section of time slot two (playback segment 2 by using normal playback rate) and the ending time of executing fast forward function belongs to the first one section of time slot six (playback segment 9 by using 2 times of normal playback rate). As shown in figure 5, the unavailable video data are segment 2 and 6 that need to be transfer by creating a multicast data stream.

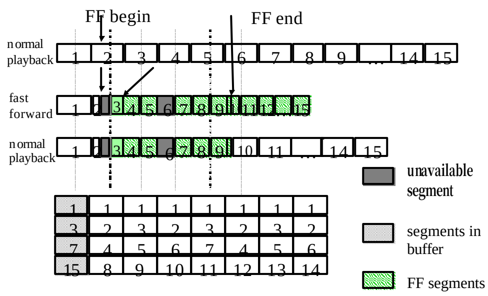

Fig. 5. The situation that the beginning time of executing fast forward function belongs to the first one section of time slot two and the ending time of executing fast forward function belongs to the first one section of time slot six.

3.2.3 Forward Jump

Assume the length of beginning position of executing forward jump function to the boundary of next time slot is D_{begin} seconds. The length of ending position of executing forward jump function to the boundary of next time slot is D_{end} seconds. Figure 9 depicts the forward jump function. According to the beginning and ending time of executing forward jump function, we do some discussions following.

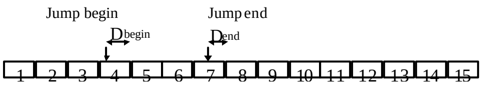

Fig. 9. Depicts the forward jump function.

If (D_{begin}>D_{end}) and (D_{begin}–D_{end})≤$\frac{1}{2}$, then the ending position of executing forward jump function moves forward (D_{begin}–D_{end}) seconds. Thus the client may watch some redundant video data of the segment belongs to the time slot of ending position. Figure 10 shows this situation.

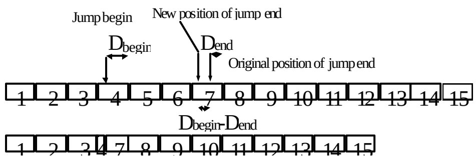

Fig. 10. The situation that (D_{begin}>D_{end}) and (D_{begin}–D_{end})≤$\frac{1}{2}$.

If (D_{begin}>D_{end}) and (D_{begin}–D_{end})>$\frac{1}{2}$, then the ending position of executing forward jump function moves to next time slot which position is D_{begin} seconds away from the boundary of next time slot. Thus the client may lose to watch some video data. Figure 11 shows this situation.

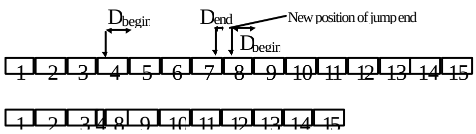

Fig. 11. The situation that (D_{begin}>D_{end}) and (D_{begin}–D_{end})>$\frac{1}{2}$.

If (D_{begin}<D_{end}) and (D_{end}–D_{begin})<$\frac{1}{2}$, then the ending position of executing forward jump function moves backward (D_{end}–D_{begin}) seconds. Thus the client may lose to watch some video data. Figure 12 shows this situation.

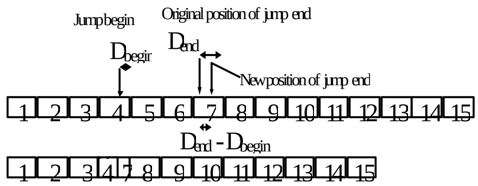

Fig. 12. The situation that (D_{begin}<D_{end}) and (D_{end}–D_{begin})<$\frac{1}{2}$.

If (D_{begin}<D_{end}) and (D_{end}–D_{begin})≥$\frac{1}{2}$, then the ending position of executing forward jump function moves to last time slot which position is D_{begin} seconds away from the boundary of next time slot. Thus the client may watch some redundant video data. Figure 13 shows this situation.

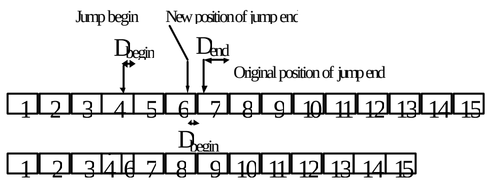

Fig. 13. The situation that (D_{begin}<D_{end}) and (D_{end}–D_{begin})≥$\frac{1}{2}$.

If (D_{begin}=D_{end}), then the ending position of executing forward jump function keeps the same. Figure 14 shows this situation.

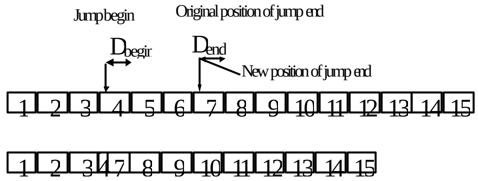

Fig. 14. The situation that (D_{begin}=D_{end}).

4. Simulation

To evaluate the performance of the SAIB scheme, we wrote a simulation program. We assume that the times of user requests for a particular video are distributed according to a poisson distribution. The numbers of VCR requests are distributed according to an exponential distribution. That is because the majority of users will execute fast forward or forward jump function at the early time, and the number of VCR requests will decrease with the time. The durations of VCR requests are distributed according to a random distribution. We assume that a video lasts 120 minutes, which is close to the average duration of a feature video. The simulation has been employed to evaluate several segment based broadcasting schemes, such as FB, PB, RFS and NPB, which number of channels is 4. The upper bound of system channels is 10.

Figure 15 and 16 depict the blocking probability of fast forward and forward jump function, respectively. We can find that the FB scheme has the lowest blocking probability, and the NPB scheme has the highest results.

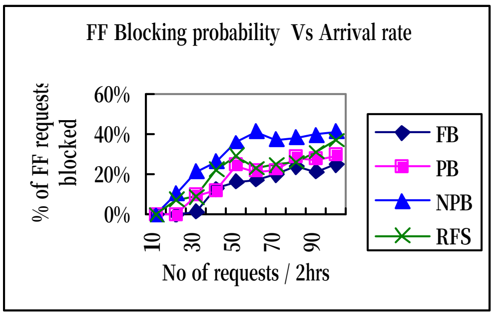

Fig. 15. The blocking probability of fast forward function vs. user arrival rate.

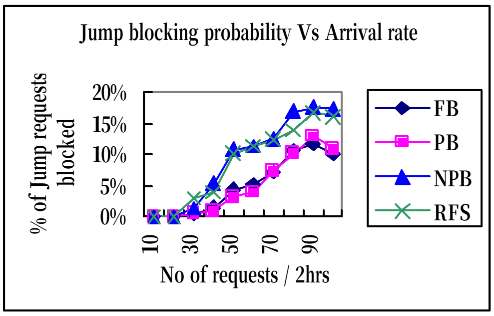

Fig. 16. The blocking probability of forward jump function vs. user arrival rate.

Figure 17 and 18 depict the additional bandwidth overhead of fast forward function and forward jump function, respectively. We can find that the FB scheme still outperformed PB, RFS and NPB. The FB scheme has the lowest additional bandwidth overhead, and the NPB has the highest results.

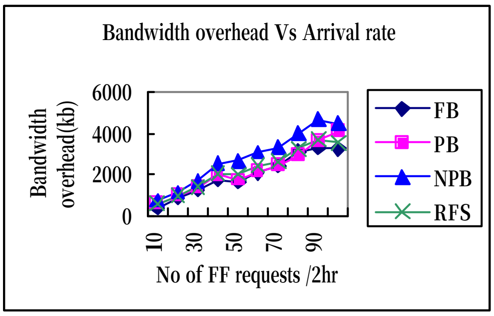

Fig. 17. The additional bandwidth overhead of fast forward function vs. fast forward request arrival rate.

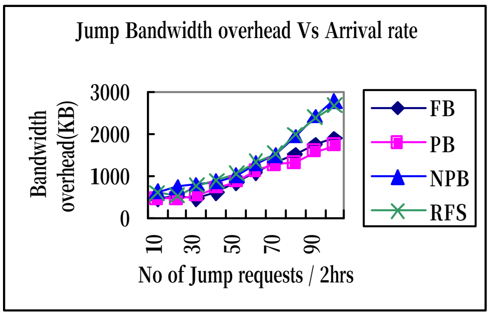

Fig. 18. The additional bandwidth overhead of forward jump function vs. forward jump request arrival rate.

According to the above simulation, we judge that these results are due to the size and broadcasting period of segments are different for each segment based broadcasting scheme. If the size of segments is larger or the broadcasting period of segments is smaller, the shared probability of the additional data stream can be increased. Thus the blocking probability and the network bandwidth requirement can be reduced further.

5. Conclusions

With the growth of broadband networks, Video-on-Demand has become realistic. Many significant broadcasting schemes have been proposed to reduce the bandwidth requirement for stored popular videos. One category is the batching broadcasting which is easy to realize the VCR functionality. But the drawback is its high cost. It needs huge bandwidth to achieve the shorter startup waiting time. Another category is the segment based broadcasting. It transfers each video according to a fixed schedule and consumes a constant bandwidth regardless of the number of requests for that video. In this category, they don't require the huge bandwidth requirement as batching broadcasting, but they only support forward jump function and fast forward function based on segment.

To aim at the drawbacks of above two broadcasting, we propose a new broadcasting scheme, called the *Segment-Alignment Interactive Broadcasting* (*SAIB*) scheme. It can be built on the top of any segment based broadcasting scheme that don't require huge bandwidth and can support fast forward function based on frame like the batching broadcasting. The technique is we align the segments that are needed to execute fast forward or forward jump function to the boundary of scheduled segments of some segment based broadcasting scheme. Therefore the shared probability of the additional data stream can be increased, and the network bandwidth requirement is reduced further.

From our analysis, we find that our SAIB scheme is suitable for smooth fast forward and forward jump playback. A simulation is employed to evaluate several segment based broadcasting schemes: FB, PB, RFS and NPB. The results reveal the FB scheme has the lowest blocking probability and additional bandwidth overhead. On the contrary, the NPB scheme has the highest results. These results are due to the size and broadcasting period of segments are different for each segment based broadcasting scheme.

6. Acknowledgement

This research was supported by the Communications Software Technology project of Institute for Information Industry and sponsored by MOEA, R.O.C. The authors would also like to thank the National Science Council of the Republic of China for fanatically supporting this research under Contract No. NSC-92-2213-E008-004.

7. References

[1] K. C. Almeroth and M. H. Ammar, "The use of multicast delivery to provide a scalable and interactive video-on-demand service," *IEEE Journal on Selected Areas in Communications*, 14(50):1110-1122, Aug. 1996.

[2] Steven W. Carter and Darrell D. E. "Stream tapping:a system for improving efficiency on a video-on-demand server," *Technical Report UCSC-CRL-97-11*, University of California,Santa Cruz,April 1997.

[3] S. W. Carter and D. D. E. Long, "Improving video-on-demand server efficiency through stream tapping", *Proc. 5th Int. Conf. On Computer Communications and Networks*, pp. 200-207, Sep 1997.

[4] Tzi-cker Chiueh and Chung-ho Lu. "A periodic broadcasting approach to video-on-demand service," *Proceedings of SPIE-The International Society for Optical Engineering*, 2615:162-9,1996.

[5] D. L. Eager, M. K. Vernon and J. Zahorjan, "Minimizing bandwidth requirements for on-demand data delivery", *Proc. 5th Int. Workshop on Advances in Multimedia Information Systems*, Oct 1999.

[6] L.S. Juhn and L.M. Tseng, "Staircase data broadcasting and receiving scheme for hot video service," *IEEE Transactions on Consumer Electronics*, vol. 43, no. 4, pp. 1110-1117, Nov 1997.

[7] L.S. Juhn and L.M. Tseng, "Harmonic broadcasting for video-on-demand service," *IEEE Transactions on Broadcasting*, vol. 43, no. 3, pp. 268-271, Sep 1997.

[8] L.S. Juhn and L.M. Tseng, "Fast data broadcasting and receiving scheme for popular video services," *IEEE Transactions on Broadcasting*, vol. 44, no. 1, pp. 100-105, Mar 1998.

[9] K. A. Hua, Y. Cai and S. Sheu, "Patching: a multicast technique for true video-on-demand services", *Proc. 6th ACM Multimedia Conf.*, pp. 191-200, Sep 1998.

[10] Jin B. Kwon and Heon Y. Yeom, "Providing VCR Functionality in Staggered Video Broadcasting," *IEEE Transactions on Consumer Electronics*, vol. 48, issue:1, pp. 41-48, Feb 2002.

[11] Yiu-Wing Leung and Tony K. C. Chan, "Design of an Interactive Video-on-Demand

System," *IEEE Transactions on Multimedia*, vol. 5, no. 1, pp. 130-140, Mar 2003.

[12] J-F. Paris, S. W. Carter and D. D. E. Long, "A Hybrid broadcasting protocol for video on demand", *Proc. 1999 Multimedia Computing and Networking Conference*, San Jose, CA, pp. 317-326, Jan 1999.

[13] J-F. Paris, "A simple low bandwidth broadcasting protocol for video on demand", *Proc. 7th Int. Conf. On Computer Communications and Networks*, pp. 690-697, Oct 1999.

[14] J.-F. Paris, "An Interactive Broadcasting Protocol for Video-on-Demand," *Proceedings of the 20th IEEE International Performance, Computing, and Communications Conference (IPCCC 2001)*, Phoenix, AZ, Apr. 2001, pages 347-353.

[15] Wing-Fai POON and Kwok-Tung Lo, "Multicast Video-on-demand System with VCR Functionality", *International Conference on Communication Technology ICCT' 98*, October 22-24.

[16] W.-F Poon, K.-T. Lo and J. Feng, "Design and analysis of multicast delivery to provide VCR functionality in video-on-demand systems," *In Proceedings of the 2nd International Conference on ATM*, pages 132-139, June 1999.

[17] Mounir A. Tantaoui, Kien A. Hua and Simon Sheu, "Interaction with Broadcast Video," *Proceedings of the 10th ACM International Conference on Multimedia*, pp. 29-38, Dec 2002.

[18] Yu-Chee Tseng, Ming-Hour Yang and Chi-He Chang, "A Recursive Frequency-Splitting Scheme for Broadcasting Hot Videos in VOD Service", *IEEE Transactions on Communications*, vol. 50, issue:8, pp. 1348-1355, Aug 2002.

[19] Shu-Zhe Weng, Simon Sheu and Jun-Yi Li, "A Cost-Effective Interactive Broadcasting Protocol for Media Streaming," *Proceedings of the National Computer Symposium 2001*, vol. D, pp. 115-122, Dec 2001.

[20] Hung-Chang Yang, Hsiang-Fu Yu and Li-Ming Tseng, "Adaptive Live Broadcasting for Highly-Demand Videos," *Journal of Information Science and Engineering*, vol. 19, no. 3, pp. 531-549, May 2003.