Proceedings of the 6th WSEAS International Conference on Multimedia Systems & Signal Processing, Hangzhou, China, April 16-18, 2006 (pp179-185)

Adjustable Interleaving Staircase-Harmonic Broadcasting Scheme for Highly-Demanded Videos

HUNG-CHANG YANG[1], HSIANG-FU YU[2], LI-MING TSENG[1], YI-MING CHEN[3]
[1]Dep. of Computer Science & Information Engineering, National Central University, Taiwan
[2]Computer Center, National Central University, Taiwan
[3]Dep. of Information Management, National Central University, Taiwan

Abstract - With the growth of broadband networks, Video-on-Demand (VoD) has become realistic. Many significant broadcasting schemes have been proposed to reduce the viewers' waiting time. However, clients are required to receive video segments altogether from all channels. This disregard not only needs a lot of client bandwidth, but also incurs more buffer requirements. To escape from these constraints, this work proposes the adjustable interleaving staircase-harmonic broadcasting scheme (AISHB), which offers a tradeoff between any two of three resources: server bandwidth, client buffer spaces, and client bandwidth. When client bandwidth is not limited, the scheme requires a client to buffer only 25% of a playing video and the waiting time is slightly higher than the optimal waiting time. In comparison with the fast broadcasting, recursive frequency-splitting, and harmonic broadcasting, AISHB saves the buffer requirements by 50%, 33%, and 33%. If client bandwidth is restricted, AISHB achieves the smallest waiting time among all currently known broadcasting schemes.

Keywords: VoD, hot video broadcasting, waiting time, buffer requirements, client bandwidth requirements

1. Introduction

Video-on-Demand (VoD) allows clients to watch a video of their choice at the time of their choice. A VoD system is typically implemented by a client-server architecture, and may easily run out of bandwidth because the growth in bandwidth can never keep up with the growth in the number of clients. This results in tremendous demand for computing power and communication bandwidth on the system. To alleviate the stress on the bandwidth and I/O demands, one way is to simply broadcast popular videos. According to [4], a few (10 or 20) very popular videos constitute 80% of viewers' requests. Because the server's broadcasting activity is independent of the arrivals of requests, this approach is appropriate to popular videos that may interest many viewers at a particular period of time.

One method of broadcasting a popular video is called segment-based broadcasting schemes [1-3, 5-20, 22-24], which substantially reduce bandwidth requirement for video services. The scheme divides a video into segments in advance. A video server then simultaneously and periodically broadcasts the segments on different data channels. When clients wish to watch a video, they wait for the beginning of the first segment on the first channel. Thus, their maximum waiting time equals the length of the first segment. When the clients are watching the video, their set-top boxes (STB) or computers download sufficient data from the other channels to enable them to play the segments of the video in turn.

Most previous broadcasting schemes [6, 9-14, 16-19] were proposed to reduce the clients' waiting time. These schemes usually require clients to receive video segments altogether from all channels. Consequently, clients with limited bandwidth, such as XDSL, cannot enjoy video-broadcasting services. To solve the problem, this work proposes the adjustable interleaving staircase-harmonic broadcasting scheme (AISHB), which offers a tradeoff between any two of three resources: server bandwidth, client buffer spaces, and client bandwidth. When client bandwidth is not limited, the scheme requires a client to buffer only 25% of a playing video and the maximum waiting time is slightly higher than that of the harmonic broadcasting (HB) scheme [9]. In comparison with the fast broadcasting (FB) [10], recursive frequency-splitting (RFS) [16], and HB schemes, AISHB saves the buffer requirements by 50%, 33%, and 33%. If client bandwidth is restricted, AISHB has the smallest waiting time than the client-centric approach (CCA) [2], greedy disk-conserving broadcasting (GDB) [5], striping broadcasting (StB) [15], and interleaving staircase broadcasting (ISB) [20] schemes. For example, at a server bandwidth of ten times the video playout rate, a client bandwidth of triple the playout rate, and a client buffer of 25% of the video size, AISHB has 87%, 87%, 40% and 20% lower waiting time than CCA, GDB, StB, and ISB. To our best knowledge, AISHB obtains the smallest waiting time at a limited client bandwidth.

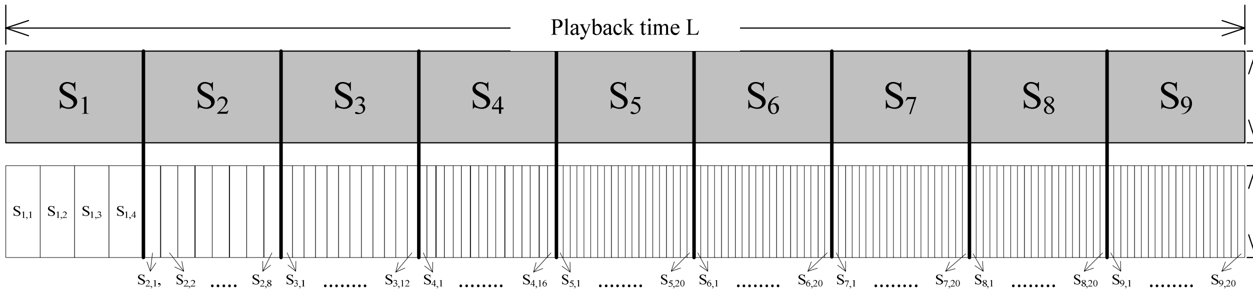

Fig.1: The fragmentation of a video by AISHB

The rest of this paper is organized as follows. In Section 2, the paper presents the AISHB scheme. Some comparisons on client waiting time, buffer and bandwidth requirements are analyzed in Section 3. Brief conclusions are drawn in Section 4.

2. Adjustable Interleaving Staircase-Harmonic Broadcasting Scheme (AISHB)

As mentioned above, the HB scheme has the best performance on clients' waiting time, and the strength of the SB scheme is small client buffer requirements. This work proposes the adjustable interleaving staircase-harmonic broadcasting scheme (AISHB), which integrates HB and SB to obtain the strengths of small waiting time and buffer requirements. The AISHB scheme guarantees that a client only needs to buffer 25% of a playing video, and exhibits a tradeoff between any two of three resources: server bandwidth, client buffer spaces, and client bandwidth.

On the server side, the AISHB scheme has the following steps:

1. The server equally divides a video into N segments. Let L be the length of the video. Thus, the length of each segment is $d = L/N$. We also let such time length be a time slot. Suppose that S_i is the i^{th} segment of the video. The concatenation of all the segments constitutes the whole video, $S = S_1 \bullet S_2 \bullet \cdots \bullet S_N$.
2. The basic concept of AISHB is to take advantage of HB and SB to broadcast segments at the lower server's and client's bandwidth. For easy explanation, we assume that the segments numbered 1 to N_{HB} are distributed by HB, and the remaining segments are distributed by SB. The server further equally divides segment S_i into im sub-segments, where $1 \le i \le N_{HB}$ and m called interleaving factor is an integer. For segment S_i, $N_{HB}+1 \le i \le N$, the server equally divides it into $(N_{HB}+1)2^{\left\lfloor \log_2 \left\lceil \frac{i-N_{HB}}{N_{HB}+1} \right\rceil \right\rfloor} m$ sub-segments. Suppose that $S_{i,j}$ is the j^{th} sub-segment of S_i, the concatenation of all the sub-segments constitutes the whole segment. Figure 1 demonstrates the video partition of the AISHB scheme, where $N = 9$, $m = 4$,

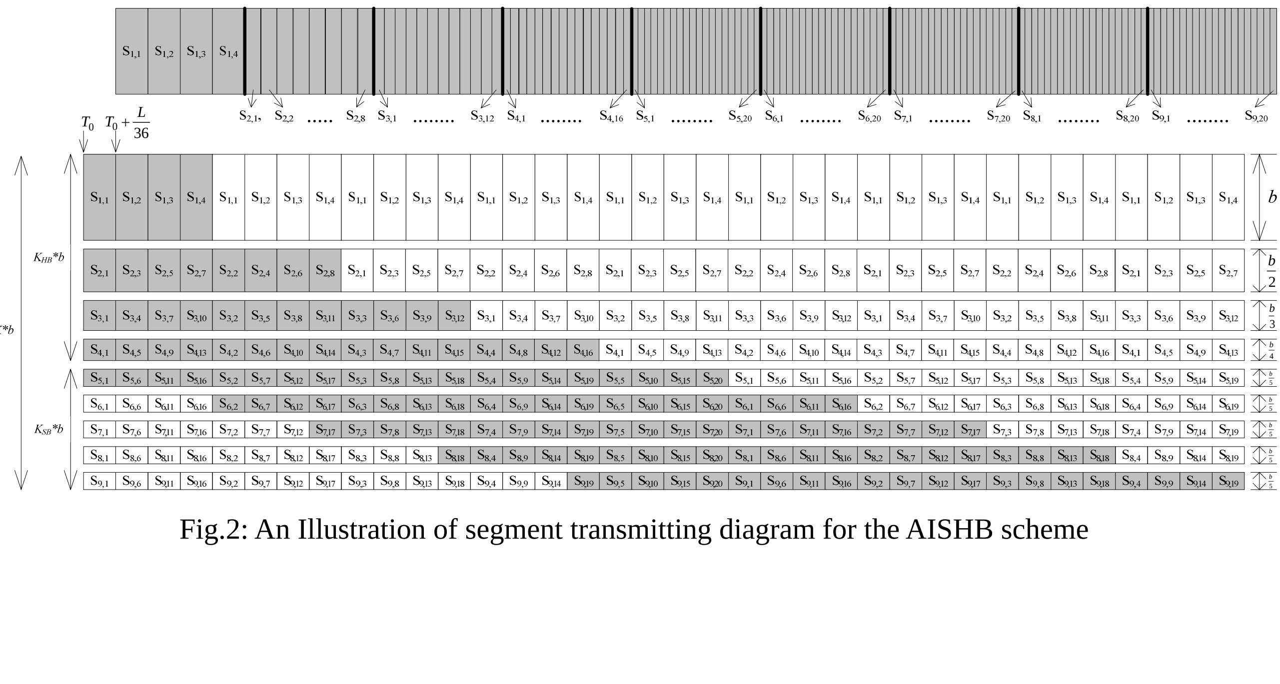

Fig.2: An Illustration of segment transmitting diagram for the AISHB scheme

Proceedings of the 6th WSEAS International Conference on Multimedia Systems & Signal Processing, Hangzhou, China, April 16-18, 2006 (pp179-185)

and $N_{HB} = 4$.

3. The server then allocates N channels for transmitting the segments. Each channel C_i is for segment S_i. Its sub-segments are broadcasted in the interlacing order on this channel periodically, as indicated in Fig. 2. Here, $K * b$ represents the server bandwidth requirements. $K_{HB} * b$ and $K_{SB} * b$ stand for the bandwidth required for segment transmission by HB and SB. We discuss their values in next section. In the figure, for channel C_i, where $1 \le i \le N_{HB}$, its bandwidth requirement is b/i. For channel C_i, where $N_{HB} + 1 \le i \le N$, its required bandwidth is $b/((N_{HB}+1)2^{\left\lfloor \log_2 \left\lceil \frac{i-N_{HB}}{N_{HB}+1} \right\rceil \right\rfloor})$.

At the client end, we assume that there are sufficient disk spaces to buffer portion of the playing video. For watching a video, the following steps are involved:

1. The client begins to download video data once the start of segment S_1 occurs on channel C_1. Let T_0 be the time to download video segments. From channels C_1 to $C_{N_{HB}}$, the client downloads all of the sub-segments of segments S_1 to $S_{N_{HB}}$ concurrently. Note that each sub-segment is received once. That is for segment S_i, where $1 \le i \le N_{HB}$, the client downloads it during time T_0 to $T_0 + id$. From channels $C_{N_{HB}+1}$ to C_N, the client begins to receive the sub-segments of S_i at $T_0 + (i - (\left\lfloor \log_2 \left\lceil \frac{i-N_{HB}}{N_{HB}+1} \right\rceil \right\rfloor + 1)(N_{HB}+1))d$, and stops downloading at $T_0 + id$. Once receiving all the sub-segments of segment S_i, the client assembles them into S_i.
2. The client starts playing the video at time $T_0 + d/m$. In Fig. 2, the gray blocks represent the video segments received by the client, and the client begins to play these segments at $T_0 + L/36$.

3. Analysis and Comparison

3.1 Correctness

Yang's and Tseng's studies [21][16] indicate that a video server must broadcast a segment S_i at least once in every i time slots to keep a continuous playout on the client side. The AISHB scheme follows the rule and broadcasts segment S_i, $1 \le i \le N_{HB}$ in every i time slots. In addition, the server broadcasts segment S_i, where $N_{HB} + 1 \le i \le N$, in every $(N_{HB}+1)2^{\left\lfloor \log_2 \left\lceil \frac{i-N_{HB}}{N_{HB}+1} \right\rceil \right\rfloor}$ time slots. Due to $N_{HB} \le (N_{HB}+1)2^{\left\lfloor \log_2 \left\lceil \frac{i-N_{HB}}{N_{HB}+1} \right\rceil \right\rfloor}$, the rule is true for AISHB.

However, AISHB must additionally guarantee to broadcast all sub-segments on time. For each segment other than the first segment, its data arrival rate is smaller than its playout rate. If a sub-segment is played and downloaded simultaneously, it probably happens that the data to be consumed is not received yet, as illustrated in Fig. 3. In the figure, sub-segments $S_{3,1}$, $S_{3,4}$, $S_{3,7}$, $S_{3,10}$, $S_{4,1}$, $S_{4,5}$, $S_{4,9}$, and $S_{4,13}$ are delivered late although the entire segments S_3 and S_4 are distributed on time. Thus, if a client plays segment S_i at time $T_0 + (i-1)d$, the playing may be interrupted. From the figure, we can also find that if the playing time of these segments is changed to delay $d/4$, then they will be played continuously. Therefore, to guarantee the

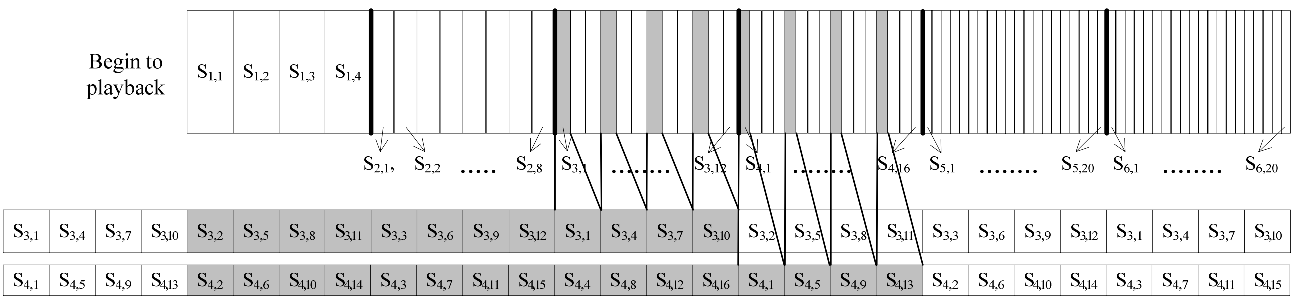

Fig. 3: An illustration that the sub-segments $S_{3,1}$, $S_{3,4}$, $S_{3,7}$, $S_{3,10}$, $S_{4,1}$, $S_{4,5}$, $S_{4,9}$, and $S_{4,13}$ can not be delivered on time, where $m = 4$

continuous playback, the client must delay to consume segment S_i at time $T_0+(i-1)d+d/m$. That is, the video should be played in the order of $S{=}S_1\bullet S_2\bullet\cdots\bullet S_N$ at T_0+d/m.

3.2 Server Bandwidth Requirements vs. Client Waiting Time

For the AISHB scheme, if a client downloads the first segment immediately once he or her requests a video, the client's waiting time $\delta_{\min}$ is minimum and thus $\delta_{\min}=d/m$.

If the client just misses the start of the first segment, then the client will have the maximum waiting time $\delta_{\max}=d+d/m$. (1)

From Fig. 2, the total bandwidth allocated is

$$K*b=K_{HB}*b+K_{SB}*b$$
$$=\sum_{i=1}^{N_{HB}}\frac{b}{i}+\sum_{i=N_{HB}+1}^{N}\frac{b}{(N_{HB}+1)2^{\left\lfloor \log_2\left\lceil \frac{i-N_{HB}}{N_{HB}+1}\right\rceil\right\rfloor}} \quad (2)$$

Thus, $K_{SB}=\sum_{i=N_{HB}+1}^{N}\frac{1}{(N_{HB}+1)2^{\left\lfloor \log_2\left\lceil \frac{i-N_{HB}}{N_{HB}+1}\right\rceil\right\rfloor}}$ and $K_{HB}=\sum_{i=1}^{N_{HB}}\frac{1}{i}$. Note that given N and N_{HB}, or N and K_{SB}, the values of other variables can be driven by the above equations.

Given a video of 120 minutes, we can obtain the relationship between server bandwidth and client waiting time according to equations (1) and (2), as shown in Fig. 4, where $K_{SB}=1$. (The value of m is explained in next subsection.) The figure indicates that the AISHB scheme consumes the smallest bandwidth, except as compared to the HB scheme. For example, if the waiting time equals one second, the HB, AISHB, CHB, MSB, RFS, ISB and FB schemes consume 9.46*b*, 9.66*b*, 9.96*b*, 9.98*b*, 10*b*, 12.76*b* and 13*b*, respectively.

3.3 The Effect of the Factor: *m*

According to equation (1), we can find that the waiting time reduces as the factor m increases. However, the value of m cannot be infinitely large since the value is bounded by the length of a frame, a basic unit of a playing video. Assume that the frame rate of a video is f (frames/second). Given a fixed bandwidth K and K_{SB}, the value of N_{HB} and the number N of segments can be obtained by equation (2). From AISHB, the last segment are divided into most sub-segments which length cannot be less than a frame length; thus,

$$m\le\frac{fL}{N(N_{HB}+1)2^{K_{SB}-1}} \quad (3)$$

Assume that the frame rate of a video is 30 frames/s and the length of a video is 2 hours. According to equations (2) and (3), we obtain the largest value of m, which is then used to make Fig. 4.

3.4 Client Buffer Requirements

Assume that the size of each segment is x. Figure 5 demonstrates how the buffered data (unit in x) change with time. For example, during time T_0 to T_0+d, a client must buffer the first segment from

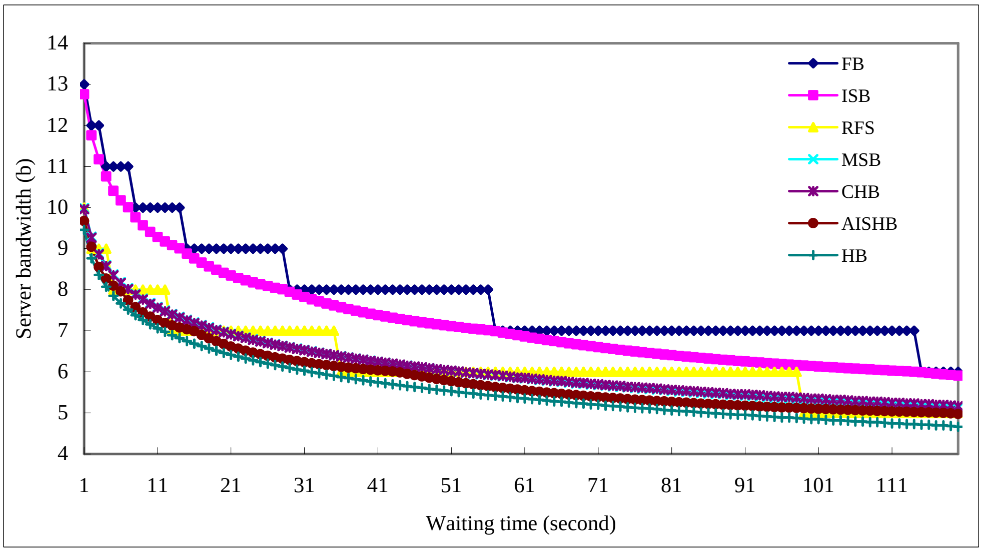

Fig. 4: The required bandwidth versus maximum waiting time

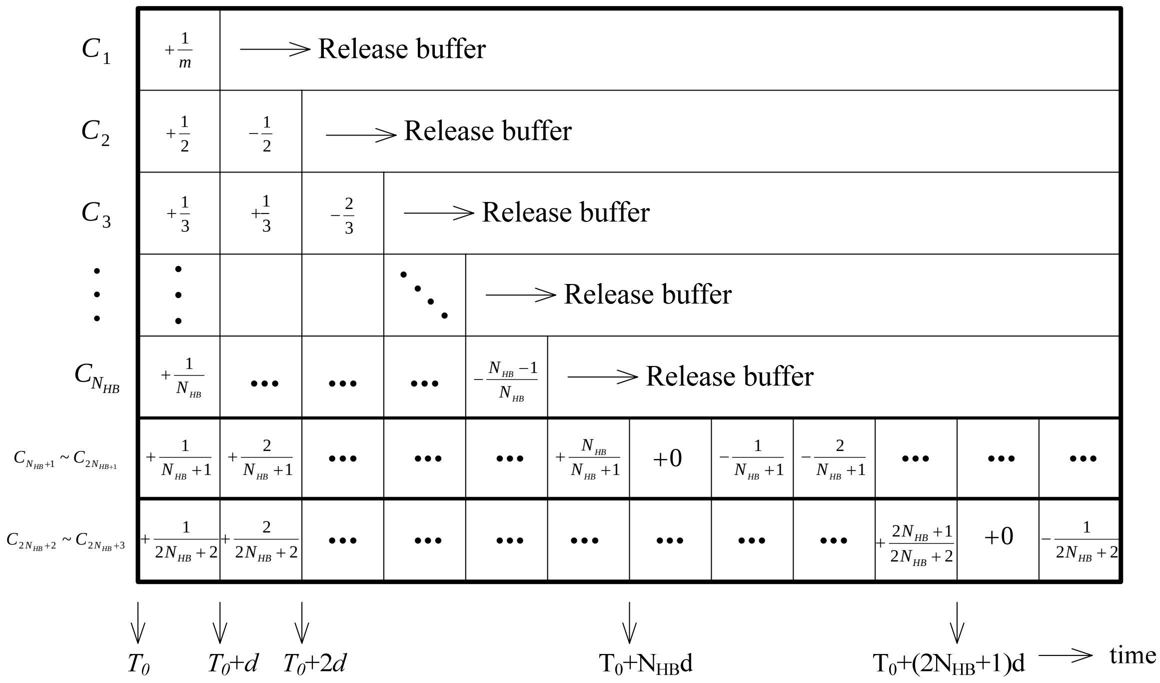

Fig. 5: Rising rate of buffered data in size of data segment, where $K_{SB}=2$

channel C_1, while consuming portion of the segment (*i.e.*, $m-1/m$) because the client starts playing the video at time T_0+d/m. Thus, the figure depicts $+1/m$ in the corresponding rectangle (at the top left corner). From Fig. 2, it is clear that the client's data arrival rate is largest at time T_0 and then decreases with time. Since the data consumption rate equals b, the maximum buffer requirements happen at the time when the data arrival rate also becomes b. Suppose that K_{SB} is an integer larger than zero. From AISHB, the maximum buffer requirements appear at time $T_0+2^{K_{SB}-1}(N_{HB}+1)d$. Since the client has played segments numbered 1 to $2^{K_{SB}-1}(N_{HB}+1)$ and has played $m-1/m$ of the segment numbered $2^{K_{SB}-1}(N_{HB}+1)-1$, the client just buffers the incoming data of the remaining segments at this time. Thus, the maximum buffer requirements are

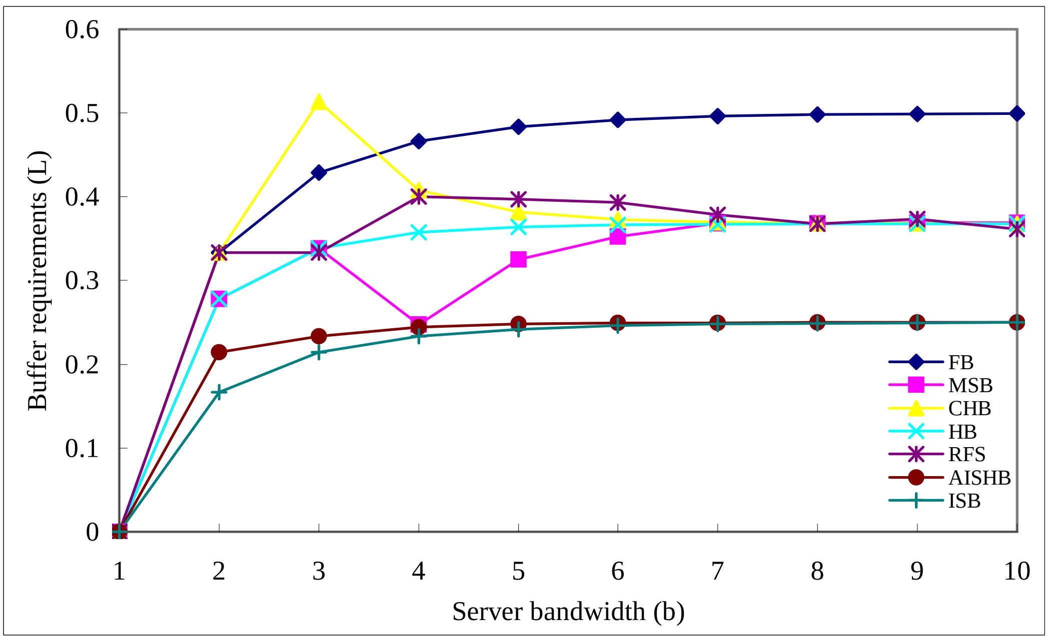

Fig.6: The maximum buffer requirements versus server bandwidth

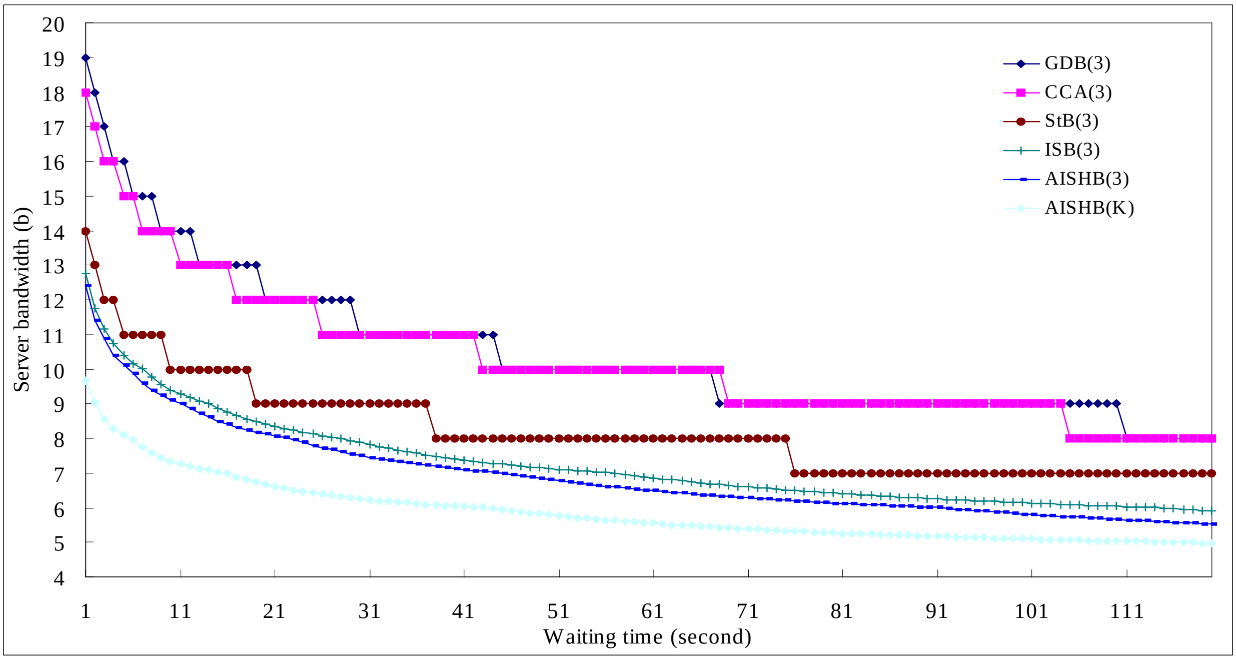

Fig. 7: The required bandwidth versus maximum waiting time at a limited client bandwidth

$$Z = x * \sum_{i=1}^{2^{K_{SB}-1}(N_{HB}+1)-1} \frac{i}{2^{K_{SB}-1}(N_{HB}+1)} + \frac{x}{m}. \quad (4)$$

If the buffer demand is in the percentage of video length, the demand is about 0.25 when N is large. Figure 6 shows that the AISHB scheme requires the smallest buffer spaces, except as compared to the ISB scheme.

3.5 Client Bandwidth Requirements vs. Waiting Time

The AISHB scheme supports a client with limited bandwidth by increasing server bandwidth consumption. Let a client's bandwidth be $U * b$. From AISHB, the maximum data arrival rate of the client appears at time T_0. Assume that K_{SB} is infinitely large. To ensure continuous playing at the client end, the server must broadcast video segments according to the following inequality.

$$b\sum_{i=1}^{N_{HB}} \frac{1}{i} + 2b/(N_{HB}+1) \le U * b \quad (5)$$

Fig. 7 shows the server bandwidth versus the waiting time at a limited client bandwidth according to equation (2) and inequality (5), where the client's bandwidth is $3b$ ($i.e.$, $U = 3$) and the buffer spaces are 25% of video size. AISHB outperforms all the schemes on the waiting time. For example, when the server bandwidth is $10b$, AISHB has 87%, 87%, 40% and 20% lower waiting time than CCA, GDB, StB, and ISB. To our best knowledge, AISHB has the smallest waiting time at a limited client bandwidth.

4. Conclusions

To serve clients with limited resources, this work proposes AISHB, which offers a tradeoff between any two of three resources: server bandwidth, client buffer spaces, and client bandwidth. When client bandwidth is not limited, the scheme requires a client to buffer only 25% of a playing video and the maximum waiting time is slightly higher than the optimal waiting time. In comparison with the FB, RFS, and HB schemes, AISHB saves the buffer requirements by 50%, 33%, and 33%. If client bandwidth is restricted, AISHB achieves the smallest waiting time among all currently known broadcasting schemes.

Acknowledgement. The authors would like to thank the National Science Council of the Republic of China for financially supporting this research under Contract No. NSC 94-2213-E-008-026.

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] Ying Cai, Kien A. Hua and Simon Sheu, “Leverage client bandwidth to improve service latency of distributed multimedia applications,” *Journal of Applied Systems Studies*, 2(3), 2001

[3] Satish Chand and Hari Om, "Modified staircase data broadcasting scheme for popular videos," *IEEE Trans. on Broadcasting*, vol. 48, no. 4, pp. 274-280, Dec. 2002.

[4] Asit Dan, Dinkar Sitaram, Perwez Shahabuddin, "Dynamic batching policies for an on-demand video server," *Multimedia Systems*, vol. 4, no. 3, pp. 112–121, June 1996.

[5] Lixin Gao, Jim Kurose, and Don Towsley, "Efficient schemes for broadcasting popular videos," *Multimedia Systems*, vol. 8, pp. 284-294, 2002.

[6] Ailan Hu, "Video-on-Demand Broadcasting Protocols: A Comprehensive Study," *IEEE INFOCOM*, pp. 508-517, 2001.

[7] K. A. Hua and S. Sheu, "An efficient periodic broadcast technique for digital video libraries," *Journal of Multimedia Tools and Applications*, 5, pp. 1-20, 1998.

[8] 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.

[9] 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.

[10] 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.

[11] Anirban Mahanti, Derek L. Eager, Mary K. Vernon and David Sundaram-Stukel, "Scalable On-Demand Media Streaming with Packet Loss Recovery," *IEEE Trans. on Networking*, vol. 11, no. 2, pp. 195-209, April 2003,

[12] J-F. Paris, S. W. Carter, and D. D. E. Long, "A low bandwidth broadcasting protocol for video on demand," *Proc. Int'l Conf. on Computer Communications and Networks*, pp. 690-697, Oct. 1998.

[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, "A fixed-delay broadcasting protocol for video-on-demand," *Proc. of Int'l Conf. on Computer Communications and Networks*, pp. 418-423, 2001

[15] Simon Sheu, Kien A. Hua and Ying Cai, "A novel broadcast technique for theaters in the air," *Proc. of WVUME'2000*, pp. 218-225, July 2000.

[16] 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.

[17] S. Viswanathan and T. Imielinski, "Pyramid Broadcasting for video on demand service," in *Proceedings of IEEE Multimedia Computing and Networking Conference*, vol. 2417, pp. 66-77, San Jose, California, 1995.

[18] 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.

[19] Hung-Chang Yang, Hsiang-Fu Yu, Li-Ming Tseng and Yi-Ming Chen, "Interleaving Harmonic Broadcasting and Receiving Scheme with Loss-Anticipation Delivery," *Proceedings of IEEE Symposium on Computers and Communications*, June 2004.

[20] Hung-Chang Yang, Hsiang-Fu Yu, Li-Ming Tseng and Yi-Ming Chen, "Interleaving Staircase Broadcasting and Receiving Scheme with Loss-Anticipation Delivery," *Proceedings of International Conference on Internet Computing*, June 2004.

[21] Z.-Y. Yang, L.-S. Juhn, and L.-M. Tseng, "On Optimal Broadcasting Scheme for Popular Video Service," *IEEE Transactions on Broadcasting*, vol. 45, no. 3, pp. 318-322, September 1999.

[22] Hsiang-Fu Yu, Hung-Chang Yang, Li-Ming Tseng, and Yi-Ming Chen, "Simple VBR Staircase Broadcasting (SVSB)," *Computer Communications*, vol. 28, no. 17, pp. 1903-1909, Oct. 2005.

[23] Hsiang-Fu Yu, Hung-Chang Yang, Yi-Ming Chen, Li-Ming Tseng, and Chen-Yi Kuo, "Smooth Fast Broadcasting (SFB) for Compressed Videos," *Lecture Notes in Computer Science*, 2957, pp. 272-283, Jan. 2004.

[24] Hsiang-Fu Yu, Hung-Chang Yang, Yi-Ming Chen, and Li-Ming Tseng, "A Smooth Recursive Frequency-Splitting Scheme for Broadcasting VBR-encoded Hot Videos," *Lecture Notes in Computer Science*, 3079, pp. 950-960, Sep. 2004.