Adaptive Playback Buffer for Wireless Streaming Media

Wanqing Tu
Department of Computer Engineering
and Information Technology,
City University of Hongkong,
Kowloon, Hongkong, P.R.China.
E-mail: tu.wanqing@student.cityu.edu.hk

Weijia Jia
Department of Computer Engineering
and Information Technology,
City University of Hongkong,
Kowloon, Hongkong, P.R.China.
E-mail: itjia@cityu.edu.hk

Abstract—**The wireless streaming media communications are fragile to the *delay jitter* because the conditions and requirements vary frequently with the users' mobility. Buffering is a typical way to reduce the *delay jitter* of media packets before the playback, however, it will incur a longer *end-to-end delay*. Our motivation in this paper is to balance the elimination of *delay jitter* and the decrease of *end-to-end delay*. We propose a novel *adaptive playback buffer* (APB) based on the probing scheme. By utilizing the probing scheme, instantaneous network situations are collected, and then using with the *delay margin* and the *delay jitter margin*, the playback buffer is adaptively adjusted to represent the continuous and real-time streaming media at the receiver. Unlike the previous studies, the novelty and contributions of the paper are: a) Accuracy: by employing the instantaneous network information, the adjustment to the playback buffer correctly reflects the current network situations, which makes the adjustment effective; b) Efficiency: by utilizing the simple probing scheme, APB achieves the current network situations without the complex mathematic prediction, and makes it efficient to adjust the playback buffer. Performance data obtained through extensive simulations show that our APB is effective to reduce the *delay jitter* and to decrease the *buffer delay*.**

I. INTRODUCTION

Streaming media (e.g. audio, video and multimedia documents) are continuous and time-based that utilize the streaming transmission technologies in the Internet/Intranet. Wireless interactive streaming media applications, such as the wireless multimedia chats and the wireless games, which are characterized with the frequently generated communications between two users, become more and more popular in recent years. In these applications, the multimedia data are transmitted to the destination through some physical media (e.g. radio). During the transmission, besides the fluctuations in network throughput, due to the users' mobility, the conditions and requirements in wireless communications vary frequently, hence, the wireless streaming media are fragile to the *delay jitter*. *Delay jitter* is the variance in one-way latency and is calculated based on sending and receiving time stamps of consecutive packets sent out. *Delay jitter* will cause highly objectionable disruption or stalling in playback. Hence, eliminating the *delay jitter* before the playback is necessary for the wireless streaming media.

A typical way to reduce the frequency of playback disruption is to employ a playback buffer at the receiver. The receiver delays the playback and places the streaming media data in the playback buffer before the first packet is represented. When the data in the buffer reaches the predetermined threshold, known as the *playback point*, the buffered data will start to play. With such buffering, the receiver can pre-fetch data that is not immediately needed when the available bandwidth is above the media rate, which prepares for providing data to the receiver when the available bandwidth is below the media rate. [1] describes such playback buffer with a fixed volume. In [1], the idea to reduce the frequency of *delay jitter* is that the playback of the i-th packet should be later than its arrival. According to this, the playback buffer volume B is calculated as:

$$B = 2\lceil((d_n)_{max} - (d_n)_{min})r\rceil \quad (1)$$

where $(d_n)_{max}$ and $(d_n)_{min}$ refer to the maximum and the minimum network transmission delay respectively, r is the playback rate, $\lceil((d_n)_{max} - (d_n)_{min})r\rceil$ represents the smallest integer that greater than $((d_n)_{max} - (d_n)_{min})r$. In such a way, the playback disruption will not occur when the available bandwidth is temporarily below the media rate, unless the buffer is also empty. The mechanism is simple. However, the additional and fixed buffering of media data incurs a longer *end-to-end delay* before the first packet is represented at the receiver. (In the rest of this paper, the term of *end-to-end delay* refers to the first packet *end-to-end delay* of one communication in the interactive wireless streaming media applications.) Interactive streaming media communications have a strong requirement for the *end-to-end delay*. There is a trade-off between reducing the *delay jitter* and decreasing the *end-to-end delay*.

Considering of the trade-off, Wenfei Xu *et al.* [2] propose the dynamic playback buffer based on the *delay balance algorithm*. The *delay balance algorithm* adjusts the playback buffer based on the prediction. It predicts the current buffer delay according to the previous playback performances, then utilizes the predicted buffer delay to adjust the buffer volume. The *delay balance algorithm* is to guarantee that there is no underflow in the buffer during the communications, hence, the buffer volume BL_t is

$$R_p(d_n)_{max} \geq BL_t \geq 0, t > (d_n)_{max} \quad (2)$$

where R_p is the media data transmission rate, $(d_n)_{max}$ is the maximum network transmission delay. The algorithm

0-7803-8783-X/04/$20.00 © 2004 IEEE

decreases the buffer volume when the network situations become stable, hence, the dynamic playback buffer guarantees a relatively shorter delay as it reduces the *delay jitter*. However, the prediction cannot deal with the "burst" of the network. Moreover, it is complex and inaccurate to achieve $(d_n)_{max}$ by using the prediction.

This paper proposes a novel *adaptive playback buffer* (APB) based on the probing scheme. By utilizing the probing scheme, instantaneous network situations (i.e. network transmission delay) are collected, and then using with the *delay margin* and the *delay jitter margin* (the *delay margin* and the *delay jitter margin* are defined in Section III), the playback buffer is adaptively adjusted to represent the continuous and real-time streaming media at the receiver. Unlike the previous work, the novelty and contributions of the paper are:

- *Accuracy*: by employing the instantaneous network information, the adjustment to the playback buffer correctly reflects the current network situations, which makes it effective to adjust the playback buffer;
- *Efficiency*: by utilizing the simple probing scheme, APB achieves the current network situations without the complex mathematic predictions, and makes the adjustment efficient.

The rest of the paper is organized as follows. In Section II, we present our motivation to design APB and the criteria to evaluate the playback performances of streaming media. We then describe the *adaptive playback buffer* in Section III. Section IV evaluates the performances of APB and other playback buffers by using the computer simulations. In Section V, we conclude the paper.

II. Motivation and Criteria

Buffering the data at the receiver until the predetermined threshold is reached will eliminate the *delay jitter* and continuously represent the streaming media for the users. However, the real time of wireless streaming media is affected directly. Our motivation in this paper is to balance the elimination of *delay jitter* and the decrease of *end-to-end delay* to achieve the acceptable playback performances at the receiver. The followings are the major criteria to evaluate the playback performances of wireless streaming media communications.

- *Continuity*. It refers to the streaming meida data can be represented without pattern dither and audio disruption at the receiver. *Continuity* can be achieved by reducing the *delay jitter*. The *sufficient and necessary condition* for realizing *continuity* is:

  Sufficient and necessary condition for Continuity: Random two adjacent transmission delay intervals of the multimedia stream are equal. Namely,

  $$TD_1 = TD_2 = ... = TD_i = ... = TD_n \qquad (3)$$

  where TD_i is the i-th transmission delay interval. [3] thinks that the *delay jitter* not exceeding 10ms is acceptable for the video streams with the compressed TV quality.
- *Real Time*. The wireless interactive streaming media communications are *real-time* if the users can receive the media data within the acceptable delay bound. Namely, the receiving synchronizes with the sending of media streams. *Real time* can be achieved by decreasing the *end-to-end transmission delay*. [3] and [4] think that the *end-to-end delay* exceeding 250ms is unacceptable, in general, the users are satisfied with the delay performance if the *end-to-end delay* is under 200ms. The *sufficient and necessary condition* for realizing *real time* is:

  Sufficient and necessary condition for Real Time: The *end-to-end transmission delay* of each multimedia packet is less than one synchronization time unit. Namely,

  $$TD_i \leq T, i \in [1, n] \qquad (4)$$

  where TD_i is the *end-to-end transmission delay* of the i-th packet, T is the synchronization time unit (e.g. 250ms).

To achieve the continuous and real-time playback of wireless streaming media at the receiver, an *adaptive playback buffer* is designed in Section III.

III. Adaptive Playback Buffer (APB)

In this section, we first provide the architecture of the *adaptive playback buffer*, then we give a novel method to adjust the playback buffer based on the instantaneous network situations, the *delay jitter bound* and the *end-to-end delay bound*.

Fig. 1 shows the architecture of the *adaptive playback buffer* in the wired-cum-wireless network. *APB Controller* is the component to adaptively adjust the playback buffer. The achieved instantaneous network situations (i.e. current network transmission delay) are sent to the *APB Controller*. It then calculates the current playback buffer delay. At last, *APB Controller* adjusts the playback buffer. To adjust the playback buffer, it is important to know that the acceptable performances should not exceed the *delay jitter bound* (J) and the *end-to-end delay bound* (DB). We can adjust APB with J and DB to achieve the real-time and continuous streaming media at the receiver. Define the *delay jitter margin* as the difference between the current *delay jitter* and J. Define the *delay margin* as the difference between the current *end-to-end delay* and DB. Our basic idea of adjusting APB is, using with the current playback buffer delay, to decrease the *end-to-end delay* by utilizing the *delay jitter margin* and to eliminate the frequency of the *delay jitter* by utilizing the *delay margin*.

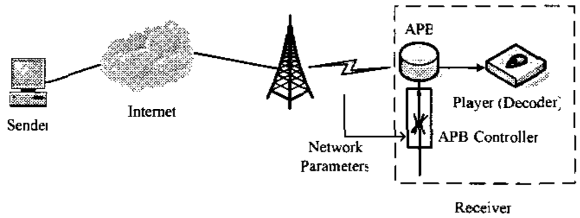

Fig. 1. The wired-cum-wireless network with APB in the receiver.

A. Network Transmission Delay

The network transmission delay d_n is a key parameter to fluent the work of *APB Controller*. In APB, we utilize the probing scheme to achieve d_n. Although the probing scheme has been studied by many well-known systems [5][6][7], it is novel to use it in the playback buffer.

The current network transmission delay $d_n(p,n)$ in Fig. 1 is expressed as

$$d_n(p,n) = \sum_{i=1}^{m} \frac{p}{c_i} + \sum_{i=1}^{m} \frac{q_i(n)}{c_i} + v + \frac{p}{c'(n)} + d(n) \quad (5)$$

where m is the number of wired links between the sender and the receiver, p is the packet size, c_i is the capacity of the i-th wired link, $q_i(n)$ is the current queueing delay at the router connecting to the i-th wired link, v is the sum of m wired links' propagation delay, $c'(n)$ is the current capacity of wireless links between the base station and the mobile nodes, $d(n)$ includes the current propagation delay and contention delay of wireless links.

In APB probing scheme, we propose using periodic light weight probes which carry the timestamps to collect the network delay properties, round trip time (RTT), of the wired and wireless links in the network. Suppose $RTT_{sr}(n)$ record the round trip time value of current probe from the sender s to the receiver r. Considering the burst and fluctuation of data communications, the exponential average value of RTT_{sr}, $EARTT_{sr}$, is utilized to estimate the network performance. The current value $EARTT_{sr}(n)$ is calculated by

$$EARTT_{sr}(n) = \beta(EARTT_{sr}(n-1)) + (1-\beta)RTT_{sr}(n) \quad (6)$$

where $EARTT_{sr}(n-1)$ is the exponential average value of $RTT_{sr}(n-1)$ and β ($0 < \beta < 1$) is a smoothing factor that determines how much weight is given to the old value $EARTT_{sr}(n-1)$. We would like to give the greater weight to more recent instance RTT. Thus, in our experiments, we use $\beta = 0.1$.

In APB probing scheme, $\frac{EARTT_{sr}(n)}{2}$ is the current network transmission delay $d_n(p,n)$.

B. Adaptive APB Size

In this subsection, we give the APB method to adaptively achieve the current APB size based on the *delay margin* and $d_n(p,n)$.

Generally speaking, the current *end-to-end transmission delay* $d_{e2e}(n)$ in Fig. 1 is:

$$d_{e2e}(n) = d_n(p,n) + d_b(n) + d_c \quad (7)$$

where $d_b(n)$ is the current buffer delay of streaming media, d_c is the codec delay. For the audio streams, d_c is always a constant by the same codec; for the video streams, d_c is a constant for the frames with the same type (e.g. Introcoded frames in MPEG-1) by the same codec. Hence, APB thinks that d_c is a constant after it recognizes the frame type which can be achieved through the codec scheme. APB intends to guarantee the real time as the playback buffer reduces the *delay jitter*. The communications with the delay not exceeding DB are real-time communications. In general, there is a difference between $d_{e2e}(n)$ and DB. It is the current *delay margin*. The current *delay margin* provides the current buffer delay $d_b(n)$. In APB, $d_b(n)$ is calculated by

$$d_b(n) = DB - d_n(p,n) - d_c \quad (8)$$

Thus, the current APB size $APBS(n)$ is

$$APBS(n) = d_b(n)r = (DB - d_n(p,n) - d_c)r \quad (9)$$

where r is the streaming media transmission rate.

The following is the analysis of the variation range of APB volume. Since d_c is some constant for both audio streams and the video frames with the same type by the same codec, (8) shows that $d_b(n)$ is a monotonic decreasing function with regard to $d_n(p,n)$. The maximum buffer delay $(d_b(n))_{max}$ is achieved when $(d_n(p,n))_{min}$ appears. Namely,

$$\begin{aligned}(d_b(n))_{max} &= max\{DB - d_c - d_n(p,n)\} \\ &= DB - d_c - (d_n(p,n))_{min}\end{aligned}$$

$$= DB - d_c - min\{\sum_{i=1}^{m} \frac{p}{c_i} + \sum_{i=1}^{m} \frac{q_i(n)}{c_i} + v + \frac{p}{c'(n)} + d(n)\} \quad (10)$$

Correspondingly, the minimum buffer delay $(d_b(n))_{min}$ is obtained when $(d_n(p,n))_{max}$ appears. $(d_b)_{min}$ is expressed by:

$$\begin{aligned}(d_b(n))_{min} &= min\{DB - d_c - d_n(p,n)\} \\ &= DB - d_c - (d_n(p,n))_{max} = 0 \quad (11)\end{aligned}$$

The variation range of APB volume depends on the variation range of $d_b(n)$. Because the streaming media data are framed at the sender by the coder, to guarantee the continuous playback, one intact frame is necessary for the decoder at the receiver. Thus, the variation range of APB volume VR is:

$$max\{r(d_b(n))_{min}, v\} \leq VR \leq r(d_b(n))_{max} \quad (12)$$

where v is the maximum frame size of streaming media. (12) follows:

$$v \leq VR \leq r(d_b(n))_{max} \quad (13)$$

C. APB Adjustment

In this subsection, we introduce the adjustment to APB to achieve the continuous and real-time playback. To adjust APB, *step length* is utilized. *Step length* is the number of buffer units that APB can add/decrease each time without incurring the *delay jitter* and longer *end-to-end delay*. The current APB size is calculated by (9), then the current variation of APB size is

$$\triangle APBS(n) = \frac{[APBS(n) - APBS(n-1)]}{u} \quad (14)$$

where $APBS(n-1)$ is the last APB size, u is the bit number that each APB unit contains. To avoid the playback disruption, the adjustment to APB should not exceed the *delay jitter bound*

J. The maximum playback buffer units U that can be adjusted without causing the playback disruption is

$$U = \frac{\lceil Jr \rceil}{u} \tag{15}$$

where r is the streaming media transmission rate, $\lceil Jr \rceil$ is the smallest integer that greater than Jr. Hence, according to (14) and (15), the *step length sl* is

$$sl = min\{|\triangle APBS(n)|, U\} \tag{16}$$

APB is adjusted with sl each time. The positive value of $\triangle APBS(n)$ shows that the adjustment is to enlarge the buffer volume, whereas the negative value of $\triangle APBS(n)$ shows that the adjustment is to reduce the buffer volume.

Furthermore, to achieve a shorter *end-to-end delay*, APB thinks that no overflow and underflow in the playback buffer within an interval T show that the network is stable. In such situation, the fluctuation of *end-to-end delay* is very little. It is unnecessary to buffer the data as long as before. *APB Controller* will reduce the playback buffer by U units each time after it detects the stable network situations.

IV. Experimental Evaluation

A. Simulation Model

In this section, by using the computer simulations, we give the performance comparison of APB with the fixed volume buffer (FVB), the delay balance algorithm (DBA), the minimum volume buffer (MINB) whose volume is APB lower bound v.

We use *ns*-2 [11] to run our simulations on a group of SUN SOLARIS workstations. As shown in Fig.2, the simulation topology is with 19 wired nodes, 1 base station and 3 mobile nodes. The wired network adopts the MCI ISP backbone, and the wired link capacity is 100Mb. The wireless network adopts 802.11b, the base station power has an transmission range with the radius of 250m and a carrier sensing range with the radius of 500m. The period of probing packets is 10ms. The simulation traffic is the 64Kbps video flows in wireless communications.

B. Simulation Metrics

In this simulation, we use the following two metrics to evaluate the two playback performance criteria introduced in Section II of the four provided buffer schemes.

- Average Playback Delay: Define the *playback delay* as the delay that the users receive the first multimedia packet in one multimedia communication. It is the *end-to-end delay* of the first packet including the codec delay, the network delay and the buffer delay. In this simulation, we use the *average playback delay* to evaluate the real time of wireless streaming media communications.
- Average Playback Delay Jitter: Define the playback delay jitter as:

$$DJ(i) = d(i) - d(i-1) \tag{17}$$

where $d(i)$ is the current *end-to-end delay*, $d(i-1)$ is the last *end-to-end delay*. We utilize the *average playback delay jitter* to evaluate the continuity of the communications.

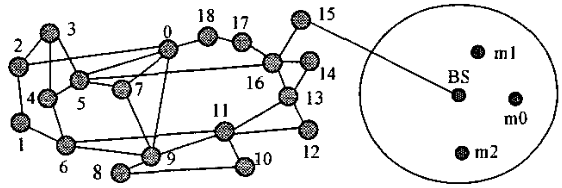

Fig. 2. Simulation topology.

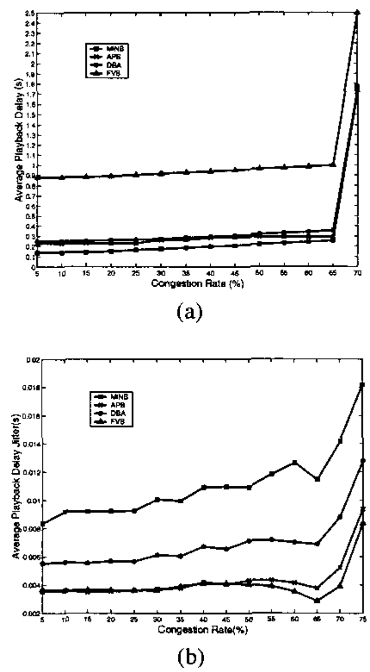

Fig. 3. The performance of *average playback delay* (a) and *average playback delay jitter* (b) under differen congestion rates.

C. Performance Observations

Fig. 3 (a) shows the *average playback delay* performances of different buffer schemes. In (a), the *average playback delay* of MINB is always the lowest one. It is because of the minimum buffer delay that MINB incurs. APB achieves lower *average playback delay* performance than DBA and FVB. Since the communication with the *end-to-end delay* under 250ms is real-time, the simulation data in (a) show that the multimedia communications are real-time when the congestion rate is not above 65% in MINB and APB and the congestion rate is not above 50% in DBA. The communication with FVB is not real-time for its large buffer delay. Fig. 3 (b) illustrates the *average playback delay jitter* performances of different buffer schemes. In (b), the *average playback delay jitter* of FVB is always the lowest one. It is because that FVB buffers enough packets to avoid the delay fluctuation incurred by the network transmission. APB achieves a much lower *average playback*

TABLE I
THE COMPARISON OF TIMES THAT THE DELAY JITTER EXCEEDS THE DELAY JITTER BOUND J.

	FVB	APB	DBA	MINB
5%	5	3	9	36
10%	5	4	9	32
15%	5	5	8	33
20%	5	7	13	31
25%	5	7	14	27
30%	5	5	7	27
35%	5	5	11	16
40%	6	7	17	42
45%	6	5	11	38
50%	6	5	11	29
55%	6	6	10	15
60%	6	7	10	13
65%	5	7	11	15
70%	15	16	17	30

delay jitter performance than DBA. This is because that the adjustment of playback buffer in APB is based on the instantaneous network situations, however, DBA is based on the prediction of the current buffer delay which cannot deal with the "burst" of the network. MINB cannot achieve continuous communications because its *average playback delay jitter* is larger than 10ms in most situations. The simulation data show that the *average playback delay jitter* is acceptable when the congestion rate is not above 75% with FVB and APB and the congestion rate is not above 70% with DBA.

Table I gives the comparison of times that the *delay jitter* exceeds the *delay jitter bound J* under different congestion rates for these four playback buffer schemes. The data show that APB is better than DBA and a little worse than FVB in terms of the frequency of *delay jitter*.

Altogether, the simulation results show that the communications with APB is continuous and real-time when the congestion rate is not above 65%, however, the communications with DBA is acceptable when the congestion rate is not above 50%. It is because of the accurate adjustment based on the instantaneous network situations to the playback buffer in APB. As for FVB, although it can smooth out the delay fluctuation, its delay is not acceptable by the users. MINB has a good delay performance, but it causes objectionable disruption or stalling in the playback. Hence, in these four schemes, APB is the best one in terms of the real-time and continuous wireless streaming communications.

V. CONCLUSION

In this paper, we have studied the *adaptive playback buffer*, APB, to guarantee the real-time and continuous wireless streaming communications in the compensatory way. We propose a novel scheme—sending probing packets to achieve the instantaneous network situations to adaptively adjust the playback buffer with the *delay margin* and the *delay jitter margin*. The design analysis and the computer simulations show that our *adaptive playback buffer* can achieve the real-time and continuous streaming communications in wireless networks indeed. We believe that APB is efficient and practical for interactive wireless streaming media communications.

ACKNOWLEDGMENT

The work is supported by Research grant council (RGC) Hong Kong, SAR China under nos CityU 1055/01E and CityU 1039/02E.

REFERENCES

[1] Anni Cai and Jinao Shuen, "Multimedia Communications", Electronics and Industry Publishing Company, Aug.2000.
[2] Wenfei Xu, Lihua Tang and Zheng Tan, "Design and Implementation of Delay Balance Algorithm for Mutual Audion in the Local Ethernet", The 7-th International Conference on Multimeida Technology, Oct.1998.
[3] Qinghua Zheng and Renhou Li, "Performance Parameter Calculation in the Distributed Multimedia Synchronization", Journal of China Institute of Communications, Oct.1999.
[4] ITU-T G.114: One-way Transmission Time. May.2000.
[5] E.W.Zegura, M.H.Ammar, Z.Fei, S.Bhattacharjee, "Application-layer anycasting," *IEEE/ACM Transactions on Networking(TON)*, August 2000, Volume 8 Issue 4.
[6] L.Breslau, E.Knightly, S.Schenker, I.Stoica and H.Zhang, "Endpoint admission control: Architecture issues and performance", In *Proc. of SIGCOM 2000*, pp.57-69.
[7] Weijia Jia, Dong Xuan, Wanqing Tu, Lidong Lin and W.Zhao, "Distributed admission control for anycast flows", *IEEE Transaction on Parallel and Distributed Systems*, vol.15, no.6, June 2004.
[8] Wanqing Tu and Haiming Qiu, "Research on Guarantee Technology of Video Stream QoS in the Sink over IP Network Based on RTP", ACTA SCIENTIARUM NATURALIUM UNIVERSITATIS SUNYATSENI, Vol.42 No.5 Sep.2003.
[9] Mengshun Zhang, "Multimedia Communications and Relative Technologies", The 7-th International Conference on Multimedia Technology, Oct.1998.
[10] Ingo Busse, Bernd Deffner, Henning Schulzrinne. Dynamic QoS control of multimedia applications based on RTP. Computer Communications 19(1996)49-58.
[11] UC Berkeley,LBL,USC/ISI,and Xerox PARC, "*ns* Notes and Documentation," October 20, 1999.