Multimedia Synchronization for Live Presentation Using the N-Buffer Approach*

Chung-Ming Huang and Ruey-Yang Lee

Institute of Information Engineering
National Cheng Kung University
Tainan, Taiwan 70101
R.O.C.
Correspondence: huangcm@locust.iie.ncku.edu.tw

Abstract

The demand of bringing multimedia information systems into distributed environments makes multimedia synchronization more difficult. In order to eliminate the side effects result from delay jitters, we propose a bounded buffer allocation scheme, in which the audio stream adopts the *blocking* synchronization scheme and the video stream adopts the *non-blocking* synchronization scheme, for live audio and video presentations in this paper. The forward synchronization schemes are performed to overcome the asynchrony anomalies. Once some anomalies of presentations are detected, a forward re-synchronization scheme is triggered to eliminate the asynchrony anomalies. Neither a global clock nor a feedback mechanism is needed using the proposed method. Based on the proposed method, trade-offs between the presentation qualities and networking resources are mathematically calculated. According to these calculable trade-offs, users can derive their own (acceptable) presentation qualities of live video and live audio media based on their available networking resources.

Keywords: Multimedia Synchronization, Blocking and Non-blocking Schemes, Live Presentation.

1 Introduction

Traditional networks provide window-based schemes for smooth flow controls and provide re-transmission schemes for error-free traffic controls. Since the human perception is not very sensitive, some synchronization discrepancies can be tolerated in a lot of applications. Thus, in order to meet real-time requirements, the traditional flow and error control schemes are not suitable for multimedia networking. For example, a video stream (i) can tolerate some errors that result from corruption or loss, but (ii) cannot tolerate the presentation discontinuity that result from window-based flow control and re-transmission-based error control schemes. In other words, to meet real-time delivery requirements or to achieve synchronization, some packets can be discarded without harm in multimedia networking [3]. Thus, one of the key considerations in multimedia networking is to maintain real-time intra-media and inter-media synchronizations without perceivable discrepancy.

Intra-medium synchronization deals with the displaying schedule (rate) of one medium's composed data units. Inter-media synchronization maintains the requirement of temporal relationship between two or more media, such as lip-synchronization. In the past few years, many synchronization protocols/methods have been proposed [1, 5, 7, 8, 9, 10, 11, 14, 15]. Most of previously proposed synchronization protocols/methods are suitable for stored multimedia information. For stored multimedia information, the transmission process at the server site can cyclically or acyclically retrieve media objects during a multimedia presentation: The transmission process can slow down the retrieval and transmission of media units, or can skip some media units for synchronization maintenance; the transmission process can follow a pre-orchestrated schedule, which is designed according to the characteristics of networks and available computing resources at the destination site, to satisfy presentation requirements. However, these synchronization protocols/methods are not suitable for live presentations containing live audio and live video streams, because the transmission rate is constant and is equal to the presentation rate.

In [13], Stone and Jeffay have proposed delay jitter management schemes to deal with longer delay media frames. According to the queue length, the monitoring policy can skip some media frames in order to decrease the display latency. But their skipping policy only considers the balance of the intra-medium synchronization discrepancy and the display latency. The inter-media synchronization, which is also very important for smooth presentations, is not considered.

*The research is supported by the National Science Council of the Republic of China under the grant NSC 84-2213-E-006-019

Additionally, the derivation of the mathematical relationship between buffer size and delay jitter, and the corresponding mathematical asynchrony anomalies are not identified. As a result, users cannot have concrete sense, which can be derived based on some mathematical formulae, about the presentation quality based on the available computing and networking resources. For example, (1) given a jitter J_{max}, how many buffer units should be allocated? What are the incurred maximum intra-medium and inter-media asynchrony anomalies? (2) Given a maximum intra-medium or a maximum inter-media asynchrony anomaly, how many buffer units should be allocated and what is the allowed jitter J_{max}? These relationships should be mathematically derived. Otherwise, the computing&networking system will have no guidance when a connection is setting up.

In this paper, we propose a synchronization method for presenting live continuous media, i.e., video and audio media, on distributed environments. Our proposed synchronization method not only considers the intra-medium synchronization but also guarantees the inter-media synchronization. The presentation scheme is as follows: When a live multimedia application is established, the transport protocols create an audio connection and a video connection that are based on their negotiated QOS parameters. Each medium connection (1) has the same transmission characteristics, i.e., has the same maximum and minimum delays, and (2) allocates enough buffer units for intra-medium synchronization and inter-media synchronization at the client site. At the client site, each medium stream has its dedicated control process(es) to perform its presentation. Media units of each connection are not pre-fetched. Once the control process at the client site receives the very first medium unit from the network connection, the dedicated control process(es) starts to display at the moment. Thus, presentations of these media streams perhaps are not synchronized at the commencement point due to the lack of global clock and random network delays, but the asynchronies are in the allowed ranges through QOS control. That is, based on the given maximum and minimum communication delays and the media transmission/presentation rate, the corresponding buffer size, which can achieve an acceptable asynchrony anomaly, can be derived using our method. Processes at the server site transmit live media, and some synchronization schemes are adopted at the client site. In our proposed method, the *non-blocking synchronization scheme* is adopted for video streams and the *blocking synchronization scheme* is adopted for audio streams. In order to lower the asynchrony anomalies, we also propose some forward synchronization schemes that can be applied at the destination site.

The rest of this paper is organized as follows. In Section 2, some preliminaries, including synchronization models, and related definitions and notations that are used in this paper, are introduced. In Section 3, analysis of intra-video-stream synchronization using the non-blocking scheme is presented. In Section 4, analysis of intra-audio-stream synchronization using the blocking scheme is presented. In Section 5, the inter-stream synchronization is given. In Section 6, some re-synchronization schemes are introduced. Finally, we have conclusion remarks in Section 6.

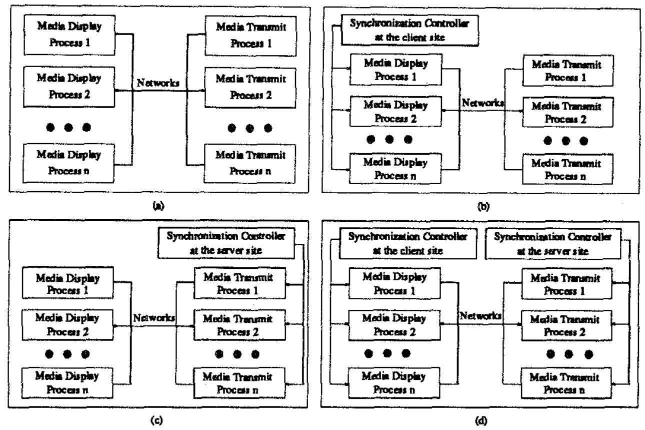

Figure 1: The synchronization model.

2 Preliminaries

For distributed multimedia applications, displaying devices at the local client site receive media units that are retrieved/transmitted from media bases located at the remote server site. The transmission server processes are responsible for transmitting media units from all media generators. The receiving client processes display received media units. Essentially, there are four synchronization models on distributed environments. Figure 1 shows the abstract architectures of these four synchronization models. The first model, which is depicted in Figure 1-(a), has no synchronization controller at both the server and client sites. The second (third) model, which is depicted in Figure 1-(b) (Figure 1-(c)) has a synchronization controller at the client (server) site but has no synchronization controller at the server (client) site. The fourth model has a synchronization controller at both server and client sites respectively. A client (server) synchronization controller can have some coordinations among media streams during media displaying (transmission) at the client (server) site. To achieve multimedia synchronization, (1) some actions can be invoked to discard some media units, to re-display some media units, and so on, at the client site; (2) the transmission rate can be adjusted if there are stored media bases, or adopting other schemes if the transmission server is a live video/audio media generator, i.e., the transmission rate cannot be changed.

In our paper, we use the third synchronization model, which is depicted in Figure 1-(c) for live multimedia presentation. There is no media displaying control process and each medium has its own multimedia virtual circuit. Once the client receives a medium unit from the virtual circuit, its dedicated playback process displays this medium unit or this medium unit is stored in the buffer temporarily. Since our goal is for live presentation, e.g., video/audio conference, the media transmission rate and the media presentation rate are equal. That is, the transmission server periodically transmits media information, and the client displays each medium unit using the same rate. Let θ denote the nominal period of every medium unit being generated at the server site and displayed at the client site. That is, the server transmits $1/\theta$ media units per second and the client displays $1/\theta$ media units per second. The server transmits media units in terms of their physical producing time. Synchronization schemes are implemented at the client site. The synchronization controller at the server site is only responsible for coordinating the transmission commencement for all media streams.

It is expected that ATM-based networks are embedded in the underlied multimedia information networking platforms, because ATM-based networks are connection-oriented and can guarantee the maximum network delay bound via resource reservation, admission control and cell multiplexing [2, 6]. Therefore, we assume that the communication delay time of each medium unit from the server site to the client site can be bounded between D_{max} and D_{min}, i.e., $[D_{min}, D_{max}]$. The communication delay time that a medium unit experiences through networks is from the commencement of the medium unit being generated at the server site to being received in the display buffer at the client site. Thus, the communication delay time includes propagation delay, transmission delay, queuing delay, and so on. Additionally, it is assumed that no medium unit, i.e., a complete video frame or an audio segment, will be lost. When some cells are lost or garbled, these cells within a medium unit M are filled with some dummy data. Thus, medium unit M perhaps has some distortion but M is still available. In this way, all media units are ensured to be received and available. In this paper, we will calculate the minimum buffer size for guaranteeing intra/inter media synchronization and for avoiding buffer overflow. Let N be the required buffer size for each medium connection. When multimedia connections are established, the transport protocols can thus allocate sufficient buffers to guarantee presentation continuity according to the expected QOS, e.g., acceptable asynchrony anomaly Δ.

In mathematical calculation, if the latest arrival time of medium unit k is later than the earliest arrival time of medium unit k+1, the bursty arrival phenomenon may happen at the receiver site. Since ATM-based networks, which are connection-oriented virtual circuit networks, are used in our method, all media units will arrive in sequence [4]. Thus, if the front media units experience the longer delay, the following media units may be queued so that bursty arrival occurs.

Let the communication delay be bounded, i.e., $[D_{min}, D_{max}]$, and let the transmission time of medium unit k be t_0. The latest arrival time of medium unit k is $t_0 + D_{max}$, and the earliest arrival time of medium unit k+1 is $t_0 + \theta + D_{min}$. The bursty arrival

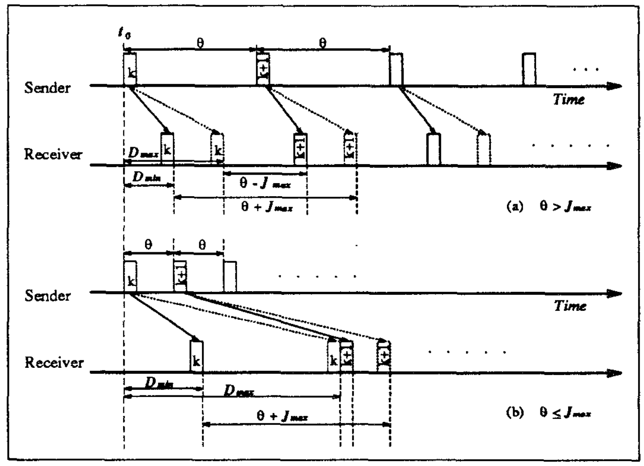

Figure 2: The transmission-receiving time diagrams of two adjacent media units.

can occur when the following equation is satisfied:

$$t_0 + \theta + D_{min} \le t_0 + D_{max}$$

$$\theta \le D_{max} - D_{min} \tag{1}$$

Let the maximum delay jitter J_{max} be equal to $D_{max} - D_{min}$. The bursty arrival can occur when the following equation is satisfied:

$$\theta \le J_{max} \tag{2}$$

The corresponding transmission-receiving time diagrams are depicted in Figure 2.

In the following sections, we propose N-buffer derivation schemes to counteract the network jitters that belong to the $\theta \le J_{max}$ situation.

3 Intra-Video-Stream Synchronization Using the Non-Blocking Scheme

In this section, the buffer requirement used for dealing with the bursty arrival, and the asynchrony anomalies between the nominal and physical presentation schedules for video streams are presented.

3.1 Buffer Requirement in the Video Stream

In order to maintain the continuity, video streams can adopt the *non-blocking* scheme. Using the non-blocking scheme, the process immediately displays the most recently arrived medium object if the expected medium object does not arrive on time. Some bursty arrival situations may occur in the video stream that adopts the *non-blocking synchronization scheme.* In the bursty arrival situations, the receiver site should allocate enough buffer units to avoid buffer overflow.

The general bursty arrival situation is depicted in Figure 3, in which a medium unit k+1 experiences the maximum delay. Let x be the latest arrived medium unit that experiences the minimum delay before presenting medium unit k+2. There are also two situations, i.e., $\frac{J_{max}}{\theta}$ equals or does not equal an integer. When $\frac{J_{max}}{\theta}$ is not an integer, i.e., medium unit x arrives before medium unit k+2 commencing to be displaying (Figure 3-(a)). The value of x is derived as follows:

$$t_0 + (x-k)\,\theta + D_{min} < t_0 + \theta + D_{max} + 2\,\theta$$

$$(x-k)\,\theta < 3\theta + J_{max}$$

$$x = k + 3 + \lfloor \frac{J_{max}}{\theta} \rfloor \tag{3}$$

Since media units k+2 to x need to be stored temporarily when medium unit k+1 is displaying, the buffer requirement is x −(k+1) buffer units. In other words, the buffer size N is as follows:

$$N = x - (k+1) = 2 + \lfloor \frac{J_{max}}{\theta} \rfloor \tag{4}$$

When $\frac{J_{max}}{\theta}$ is an integer, it implies that medium unit x arrives at $t_0 + \theta + D_{max} + \theta$, and the arrival time of medium unit x is the end time of presenting medium unit k+1. The transmission-receiving time diagram is depicted in Figure 3-(b). The value of x is derived as follows:

$$t_0 + (x-k)\,\theta + D_{min} \leq t_0 + \theta + D_{max} + \theta$$

$$(x-k)\,\theta \leq 2\,\theta + J_{max}$$

$$x = k + 2 + \frac{J_{max}}{\theta} \tag{5}$$

Since media units k+2 to x need to be stored temporarily when medium unit k+1 is displaying, the buffer requirement is also x −(k+1). The buffer size N is as follows:

$$N = x - (k+1) = 1 + \frac{J_{max}}{\theta} \tag{6}$$

From Equations (4) and (6), the buffer requirement N is as follows:

$$N = 1 + \lceil \frac{J_{max}}{\theta} \rceil \tag{7}$$

Therefore, in order to guarantee the quality of synchronizing live video streams using the intra-medium non-blocking scheme, the transport protocol must allocate $1 + \lceil \frac{J_{max}}{\theta} \rceil$ buffer units to each live video stream.

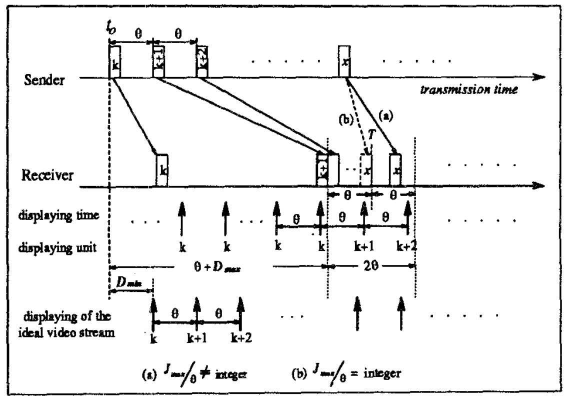

Figure 3: The transmission-receiving time diagram for a video stream that adopts the non-blocking scheme.

3.2 Asynchrony Anomalies between Nominal and Physical Presentation Schedules for Intra-Video Streams

It is assumed that the ideal live video stream displays continuously such that all media units always experience the minimum delay, i.e., there is no delay jitter between any two adjacent media units. Based on Figure 3, the maximum asynchrony anomaly occurs as the following condition: When the commencement time of the final presentation of medium unit k T_p(k) is just before medium unit (k+1)'s arrival time T_a(k+1), i.e., $\epsilon = T_a$(k+1) $-T_p$(k) and ϵ is not zero but almost zero. Let time T be the arrival time of medium unit k+1 plus θ time units. At time T, (1) the delayed video stream starts the presentation of medium unit k+1, and (2) the total number of displayed media units in the video stream is k. Thus, the anomaly of the physical presentation time and the nominal presentation time of medium unit k+1 is $\theta + J_{max}$, i.e., the maximum adjacent arriving period of medium unit k and medium unit k+1. The anomaly can also be derived as follows: Let t_{init} be the transmitting time of medium unit 1. Thus, the nominal presentation time of medium unit k+1 is $t_{init} + k\theta + D_{min}$. But the physical presentation time of medium unit k+1 is $t_{init} + k\theta + D_{max} + \theta$. Therefore, the anomaly is $(t_{init} + k\theta + D_{max} + \theta) - (t_{init} + k\theta + D_{min}) = \theta + (D_{max} - D_{min}) = \theta + J_{max}$.

Bursty arrivals are still able to occur on the following media units after medium unit k+1, i.e., some media units still may experience the maximum delay. Using the proposed bounded buffer scheme, it is guaranteed that, after the presentation of medium unit k+1 that has experienced the maximum delay, the presentations of the following media units after medium unit k+1 are smooth without any distortion.

Theorem 1: *When medium unit k+1 has experienced the maximum delay, which is depicted in Figure 3, for a live video stream presentation adopting the*

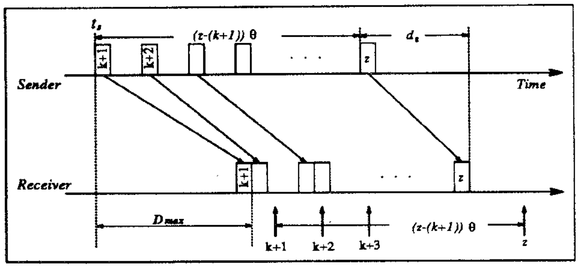

Figure 4: Guarantee the smooth presentation after a medium unit having experienced the maximum delay.

non-blocking scheme, the presentations of the following media units after medium unit k+1 are smooth without any distortion.

Proof: Let the medium unit k+1 experience the maximum delay and $z >$ k+1. Figure 4 depicts the transmission-receiving time relationship of medium units k+1 and z. Assume that medium unit z misses its presentation deadline, i.e., medium unit z does not arrive in time. In this condition, it implies that the earliest arrival time of medium unit z is greater (later) than the presentation time of medium unit z, when medium unit k+1 has experienced the maximum delay. It can be proved that medium unit z does not exist. Let the transmission time of medium unit k+1 be t_s. Since the presentation time of medium unit k+1 is certainly greater than or equal to $t_s + D_{max}$, the presentation time of medium unit z is the presentation time of medium unit k+1 plus $(z-(k+1))\,\theta$, i.e., z's presentation time is greater than or equal to $T = t_s + D_{max} + (z-(k+1))\,\theta$. Let medium unit z experience delay d_z, i.e., the arrival time of medium frame z be $T_a = t_s + (z-(k+1))\,\theta + d_z$. Since $d_z \le D_{max}$, $T_a \le T$, i.e., the arrival time of medium unit z is certainly smaller than or equal to the presentation time of medium frame z. Thus, it implies that when medium unit k+1 has experienced the maximum delay, there is at least one medium unit in the buffer at any moment after medium unit (k+1)'s presentation. In other words, the following presentations after presenting medium unit k+1 are smooth without invoking the non-blocking scheme.

The transport protocol entities negotiate QOS parameters when a multimedia network connection is constructing. One of the key parameters is the acceptable synchronization discrepancy, which is specified as Δ for convenience, during presentation. Based on the Δ parameter and related information, the transport protocol allocates the required bandwidth to establish the connection with the expected $[D_{min}, D_{max}]$. Let A denote the maximum asynchrony anomaly. Δ must be greater than or equal to A. According to the discussions presented in Section 3.1, A is $\theta + J_{max}$ in the video stream, which adopts the non-blocking scheme, using our method. Thus, the transport protocol can guarantee Quality Of Presentation (QOP) of the live video stream presentation within the acceptable synchronization discrepancy if $\theta + J_{max} \le \Delta$.

4 Intra-Audio-Stream Synchronization Using the Blocking Scheme

In this section, the buffer requirement used for dealing with the bursty arrival, and the asynchrony anomalies between the nominal and physical presentation schedules for audio streams are presented.

4.1 Buffer Requirement in the Audio Stream

In the presentation of an audio stream, it is nonsense to repeatedly play a small segment of audio information. Thus, the *blocking scheme* is adopted for the audio stream presentation. The blocking scheme is explained as follows: If there are some audio media units in the buffer, then the buffered audio media units are presented; otherwise the presentation is suspended until the following audio medium unit arrives. Some bursty arrival situations may occur in the audio stream that adopts the blocking scheme.

Figure 5 depicts the general bursty arrival situation, in which the (k+1)th medium unit experiences the maximum delay, medium unit k experiences the random delay that is in $[D_{min}, D_{max}]$, and the buffer becomes empty before the arrival of (k+1)th medium unit. Let medium unit x be the latest arrived medium unit that experiences the minimum delay before medium unit k+2's displaying. Based on the transmission-receiving time diagram depicted in Figure 5, the value of x can be derived as follows:

$$t_0 + (x-k)\,\theta + D_{min} \le t_0 + \theta + D_{max} + \theta$$

$$(x-k)\,\theta \le 2\theta + D_{max} - D_{min}$$

$$x = k + 2 + \lfloor \frac{J_{max}}{\theta} \rfloor \tag{8}$$

Since media units k+2 to x should be stored temporarily during the presentation of medium unit k+1, the buffer requirement is $x -$ (k+1). Therefore, the buffer size N is as follows:

$$N = x - (k+1) = 1 + \lfloor \frac{J_{max}}{\theta} \rfloor \tag{9}$$

Thus, the transport protocol must allocate $1 + \lfloor \frac{J_{max}}{\theta} \rfloor$ buffer units to each live audio stream in order to guarantee the quality of synchronization and avoid data loss

4.2 Asynchrony Anomalies between Nominal and Physical Presentation Schedules for Intra-Audio Stream

Based on Figure 5, the maximum asynchrony anomaly in the audio stream that adopts the blocking scheme is as follows: The maximum asynchrony

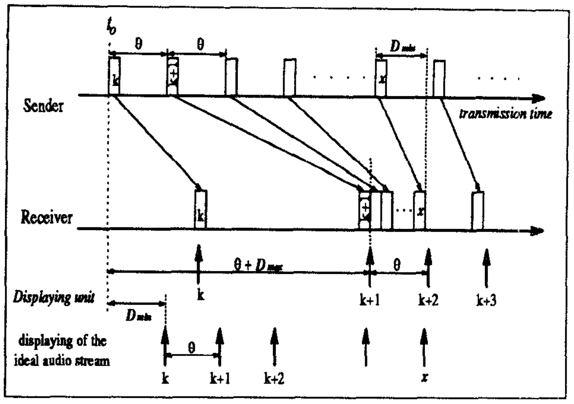

Figure 5: The transmission-receiving time diagram for an audio stream that adopts the blocking scheme.

anomaly occurs between the physical presentation time and the nominal presentation time of medium unit k+1. Let t_{init} be the transmitting time of the 1*st* medium unit. The nominal presentation time of medium unit k+1 is $t_{init} + k\,\theta + D_{min}$. The physical presentation time of medium unit k+1, which experiences the maximum delay, is $t_{init} + k\,\theta + D_{max}$. Thus, the maximum asynchrony anomaly A is as follows:

$$A = (t_{init} + k\,\theta + D_{max}) - (t_{init} + k\,\theta + D_{min})$$

$$A = D_{max} - D_{min} = J_{max} \qquad (10)$$

A must be smaller than or equal to the acceptable maximum synchronization discrepancy Δ, i.e., $J_{max} \leq \Delta$.

Theorem 2: *When medium unit k+1 has experienced the maximum delay, which is depicted in Figure 5, in a live audio stream presentation that adopts the blocking scheme, the presentations of the following media units after medium unit k+1 are smooth without blocking the audio stream.*

proof: The proof is the same as that for the video stream in **Theorem 1**.

Once a medium unit in the audio stream suffers the maximum delay, the following presentation will be smooth without waiting for the arrival of media units, i.e., the presentation will not be suspended any more.

5 Inter-Stream Synchronization Presentation

Inter-stream synchronization ensures concurrent presentations of multiple media streams. Our proposed buffer allocation scheme can (1) support intra-stream continuity, and (2) maintain the inter-stream synchronization. To be within human beings' perception allowance, inter-media asynchrony anomalies between/among streams must be restricted to some levels. Let X denote the video stream and Y denote the audio stream, the asynchrony anomaly of X be ${}_t\Delta_X$, and the asynchrony anomaly of Y be ${}_t\Delta_Y$, at any moment t. At any moment t, (1) if the physical presentation of Y is on schedule, then the inter-media asynchrony anomaly is ${}_t\Delta_X$; (2) if the physical presentation of X is on schedule, then the inter-media asynchrony anomaly is ${}_t\Delta_Y$; (3) if both X and Y experience jitter, the inter-media asynchrony anomaly is $|{}_t\Delta_X - {}_t\Delta_Y|$. The range of ${}_t\Delta_X$ (${}_t\Delta_Y$) is from zero to its maximum asynchrony value, that is, ${}_t\Delta_X \in [0,\, \Delta_{M_X}]$, where $\Delta_{M_X} = \theta + J_{max}$, and ${}_t\Delta_Y \in [0,\, \Delta_{M_Y}]$, where $\Delta_{M_Y} = J_{max}$. Thus, the maximum inter-media asynchrony anomaly is the maximum value of Δ_{M_X} and Δ_{M_Y}. The inter-media difference cannot be perceived if the maximum value of Δ_{M_X} and Δ_{M_Y} is less than the maximum perceivable level of human's perception. From the experiments conducted by Steinmetz in IBM Heidelberg, 120msec mismatching in lip-synchronization, i.e., mismatch between the audio stream and the video stream, cannot be perceived by users [12].

6 Discussions and Conclusion

Using the proposed method, the commencements of presentations of streams perhaps may not be at the same time due to different delays. The commencements of presentations of our proposed method can be more smooth, if another presentation commencement scheme is adopted. The modified scheme is as follows: The commencement of presentation can be invoked when each medium stream has buffered at least one medium unit. The modified scheme would not result in buffer overrun. Because the extreme case is when medium unit 1 of stream X experiences the maximum delay, and the front media units of stream Y experience the minimum delays. The transmission-receiving time relationship is depicted in Figure 6. In this extreme case, the latest presentation commencement time is $t_{init} + D_{max}$, where t_{init} is the transmission time of medium unit 1. Let p denote the latest arrived medium unit of stream Y before the presentation commencement time. The value of p is derived as follows:

$$t_{init} + (p-1)\,\theta + D_{min} \leq t_{init} + D_{max}$$

$$(p-1)\,\theta \leq D_{max} - D_{min}$$

$$p = 1 + \lfloor \frac{J_{max}}{\theta} \rfloor \qquad (11)$$

In other words, the quicker stream Y requires $1 + \lfloor \frac{J_{max}}{\theta} \rfloor$ buffer units. Referring to Equations (7) and (11), the modification of presentation commencement does not influence our buffer allocation scheme. If the modified scheme is adopted, the synchronization model is changed to the fourth type, which is depicted in Figure 1-(d). That is, there is one additional process to control the presentation commencement at the client site.

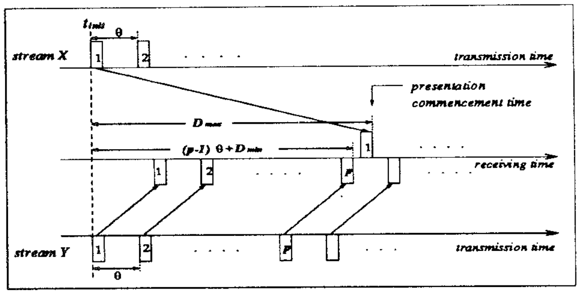

Figure 6: The transmission-receiving time relationship for the modified presentation commencement scheme.

When a medium unit w experiences a longer delay, e.g., the maximum delay, the buffer may be filled with some media units. The client site can allocate a control process to execute some re-synchronization schemes. For example, the presentation process can skip some buffered media units and directly display $(w+n_{skip}+1)$th medium unit, where n_{skip} is the number of discarded media units, in order to shorten or smooth the inter-media asynchrony anomaly. An example is as follows: Let the synchronization model be the fourth one, which is depicted in Figure 1-(d). That is, there is an additional process that can monitor the presentation and buffering status of media streams at the displaying client site. The monitoring process keeps a counter for each medium stream, such that which stream is quicker can be judged. In the following paragraphs, we propose two re-synchronization schemes, one is called the *aggressive forward scheme* and the other is called the *conservative forward scheme*, to adjust the asynchrony anomalies.

The principle of the *aggressive forward re-synchronization scheme* is as follows: At the slower stream, skip to the buffered medium unit which is most adjacent to the displaying medium unit of the quicker stream. For example, let Y's ith medium unit have just presented, Y's buffer have stored k_1 media units, and X's next presentation medium unit is $i+k$. When the aggressive forward re-synchronization scheme is adopted, the next medium unit to be presented in Y is (1) $i+k$, i.e., media units $(i+1, i+2, \ldots, i+k-1)$ are skipped, if $k_1 > k$; or (2) $i+k_1$, i.e., media units $(i+1, i+2, \ldots, i+k_1-1)$ are skipped, if $k_1 \leq k$. In this way, the asynchrony anomaly can be decreased or ceased very soon using the aggressive forward re-synchronization scheme. The side effect of adopting the aggressive forward re-synchronization scheme is as follows: The network congestions are nondeterministic. When the network congestions are present after skipping some media units, and the phenomenon occurs very often or even periodically, an oscillation presentation is generated in the extreme case. Based on the previous example, if the following media units after medium unit $i+k_1$ experience the maximum delay, the presentation becomes discontinuous and is divided into segments.

The oscillation can be decreased if the skipping procedure is step by step, i.e., stepwise skipping, such that the anomalies is ceased gradually. The modified principle results in the *conservative forward re-synchronization scheme.* The principle of the conservative forward re-synchronization scheme is as follows: Once after $I \times \theta$ time units, which is called the control interval, the number of buffered media units of Y, i.e., the slower stream, does not decrease, the presentation of Y can skip one medium unit. After M control intervals, i.e., $M \times I \times \theta$ time units, if there are more than two media units stored at the buffer of the slower stream, then the presentation process can adopt the *aggressive forward re-synchronization* scheme to keep pace with the quicker stream. In this situation, it implies that the media units of Y are arriving steadily. That is, the network traffic becomes stable. Using this hybrid method, i.e., the conservative scheme, the asynchrony anomalies can be ceased more smoothly.

Most of previously proposed multimedia synchronization methods focus on stored-data presentations, i.e., pre-orchestrated presentations. In this paper, we have proposed a bounded buffer allocation scheme to synchronize live video and live audio presentations, in which the *non-blocking* synchronization scheme is performed on the video stream and the *blocking* synchronization scheme is performed on the audio stream. A bounded buffer allocation scheme, which can avoid buffer starvation and overrun, is always adopted to guarantee the acceptable asynchrony of multimedia presentations. If the service quality can tolerate a little more asynchrony, which is within the toleration of human perception sensation, presentation systems can have allocated fewer buffer units. That is, the buffer requirement can be reduced and the presentation quality still reaches the users' required QOSs. Based on the principle, this paper analyzes the relationship between the required buffer size N and the tolerable asynchrony anomaly Δ. The proposed N-buffer allocation scheme can guarantee the bounded asynchrony anomaly and avoid buffer overflow. To have more smooth presentation, (1) an adjusted presentation commencement scheme is proposed, and (2) an aggressive forward and a conservative forward re-synchronization scheme, which won't increase the buffer requirement, are proposed to decrease the asynchrony anomalies during presentation.

References

[1] Z. Ali, M. Woo, and A. Ghafoor, "Distributed Synchronization Protocols for Multimedia Services on Internet," Proc. of IEEE International Conference on Network Protocols, pp. 204-211, 1994.

[2] D. Ferrari and D. C. Verma, "A Scheme for Real-Time Channel Establishment in Wide-Area Networks," IEEE Journal on Selected Areas in Communications, VOL. 8, NO. 3, pp. 368-379, 1990.

[3] B. Furht, "Multimedia Systems : An Overview," IEEE Multimedia, VOL. 1, NO. 1, pp. 47-59, 1994.

[4] R. Händel, M. N. Huber, and S. Schröder, "ATM Networks Concepts, Protocols, Applications," Addison-Wesley, 1994.

[5] C. M. Huang and C. M. Lo, "An EFSM-based Multimedia Synchronization Model and Authoring System," accepted by IEEE Journal on Selected Areas in Communications, 1995.

[6] J. Kurose, "Open Issues and Challenges in Providing Quality of Service Guarantees in High-Speed Networks," Proc. ACM SIGCOMM, VOL. 23, NO. 1, 1993.

[7] T. D. C. Little, and A. Ghafoor, "Network Consideration for Distributed Multimedia Object Composition and Communication," IEEE Network, VOL. 4, NO. 6, pp. 32-49, 1990.

[8] T. D. C. Little, and A. Ghafoor, "Multimedia Synchronization Protocols for Broadband Integrated Services," IEEE Journal on Selected Areas in Communications, VOL. 9, NO. 9, pp. 1368-1382, 1991.

[9] S. Ramanathan, and P. V. Rangan, "Adaptive Feedback Techniques for Synchronized Multimedia Retrieval over Integrated Networks," IEEE/ACM Trans. on Networking, VOL. 1, NO. 2, pp. 246-260, 1993.

[10] S. Ramanathan and P. V. Rangan, "Feedback Techniques for Intra-Media Continuity and Inter-Media Synchronization in Distributed Multimedia Systems," The Computer Journal, VOL. 36, NO. 1, pp. 19-31, 1993.

[11] K. Ravindran and V. Bansal, "Delay Compensation Protocols for Synchronization of Multimedia Data Streams," IEEE Trans. Knowledge and Data Engineering, VOL. 5, NO. 4, pp. 574-589, 1993.

[12] R. Steinmetz and C. Engler, "Human Perception of Media Synchronization," IBM European Networking Center, Technical Report 43.9310, 1993.

[13] D. L. Stone and K. Jeffay, "An Empirical Study of Delay Jitter Management Policies," ACM Multimedia Systems, VOL. 2, NO. 6. pp. 267-279, 1995.

[14] C. J. Wang, L. S. Koh, C. H. Wu, and M. T. Liu, "A Multimedia Synchronization Protocol for ATM Networks," Proc. of IEEE International Conference on Distributed Computing Systems, pp. 476-483, 1994.

[15] M. Woo, N. U. Qazi, and A. Ghafoor, "A Synchronization Framework for Communication of Pre-orchestrated Multimedia Information," IEEE Networks, VOL. 8, NO. 1, pp. 52-61, 1994.