A Performance Comparison of Single-Stream and Multi-Stream Approaches to Live Media Synchronization

IEICE TRANS. COMMUN., VOL. E81–B, NO. 11 NOVEMBER 1998

PAPER Special Issue on Performance and Quality of Service (QoS) of Multimedia Networks

A Performance Comparison of Single-Stream and Multi-Stream Approaches to Live Media Synchronization*

Shuji TASAKA† *and* Yutaka ISHIBASHI†, *Members*

SUMMARY This paper presents a performance comparison between the single-stream and the multi-stream approaches to lip synchronization of live media (voice and video). The former transmits a single transport stream of interleaved voice and video, while the latter treats the two media as separate transport streams. Each approach has an option not to exert the synchronization control at the destination, which leads to four basic schemes. On an interconnected ATM-wireless LAN, we implemented the four basic schemes with RTP/RTCP on top of UDP and two variants which exercise dynamic resolution control of JPEG video. Making the performance measurement of the six schemes, we compare them to identify and evaluate advantages and disadvantages of each approach. We then show that the performance difference between the two approaches is small and that the dynamic resolution control improves the synchronization quality.

key words: *media synchronization, live media, interconnected ATM-wireless network, performance comparison, experiment*

1. Introduction

Media synchronization is one of the most important functions in multimedia communications, where various kinds of media are transmitted and exchanged together so that their individual and relative structures can be preserved. It is concerned with the temporal relations between information units such as video frames and voice packets[1]. Synchronization between spoken voice and the movement of the speaker's lips (i.e., *lip synchronization*) is a typical example of media synchronization. In order to achieve lip synchronization, we first need *intra-stream synchronization*, which refers to the temporal constraints within a single stream, e.g., the preservation of time intervals between two successive video frames of a video stream. In addition, *inter-stream synchronization* is necessary; this is synchronization among plural media streams, e.g., between a voice stream and a video stream. Media capturing process and network delay jitters disturb the temporal relations. Therefore, the destination end system must have the capability of reproducing the same temporal relations as those at the source end system.

Manuscript received April 1, 1998.
Manuscript revised June 12, 1998.

†The authors are with the Department of Electrical and Computer Engineering, Nagoya Institute of Technology, Nagoya-shi, 466–8555 Japan.

*This paper was presented in part at IEEE ICC'98, Atlanta, U.S.A., June 1998.

The delay characteristics depend on the underlying communication network; for instance, ATM networks can achieve excellent performance, while wireless packet networks may produce large delay variation. Consequently, we need media synchronization mechanisms suitable for the underlying communication network.

A set of lip synchronization mechanisms studied in [2] and [3] aims at coping with various network environments. The set comprises two approaches: *single-stream* and *multi-stream*. The single-stream approach interleaves individual media streams (typically, audio and video) so that a single transport stream can be formed. On the other hand, the multi-stream approach treats them as separate transport streams. Each approach has an alternative of whether media synchronization control at the destination is carried out or not. Thus, we have four basic schemes, which are referred to as *types 0* through *3* in [2] and [3].

The two approaches we have taken were motivated by two major standardization activities in multimedia communications. One is the international standards or recommendations activity of multimedia multiplexing methods, which often takes the single-stream approach. This includes ISO/IEC 11172-1[4] for MPEG 1, ISO/IEC 13818-1[5] for the Transport Stream and the Program Stream of MPEG 2 and ITU-T H.223[6] for low bit rate multimedia communication. The other is the Internet standardization activity of multimedia transmission over the Internet, which often takes the multi-stream approach. For instance, RFC 1889 specifies *RTP/RTCP* (*Real-time Transport Protocol/Real-time Transport Control Protocol*), which provides end-to-end delivery services for real-time data such as audio and video[7]. It indicates that in a multimedia session, each medium should be carried in a separate RTP session with its own destination transport address, which implies a separate transport stream. The MBone audio and video transmission with vat[8] and vic[9] is a typical example of this approach. However, RTP/RTCP is also adopted by ITU-T H.225.0, which describes a method for combining audio, video, data and control information on a non-guaranteed *quality of service* (*QOS*) LAN to provide conversational services[10].

Advantages and disadvantages of each approach

have not yet been identified and evaluated sufficiently. Performance comparison of the two approaches is a necessary step for the progress in multimedia communications; this enables us to select the most suitable scheme for a given network environment.

To reach this goal, the authors implemented the four schemes on three kinds of networks for stored *JPEG* (*Joint Photographic Coding Experts Group*) video and voice [2] and for stored *MPEG* (*Moving Picture Experts Group*) video and voice [3]. The first network was an ATM LAN which comprises two ATM switches; it offered the *constant bit rate* (*CBR*) service, which can guarantee an end-to-end QOS. The second one was an ATM LAN with the same configuration, but it supported only the *unspecified bit rate* (*UBR*) service, which cannot necessarily guarantee an end-to-end QOS†. Thirdly, we examined an interconnected ATM-wireless LAN which consists of the ATM LAN with the CBR service and a shared media wireless LAN with a best effort service. The TCP protocol was chosen for the sake of reliability and for simplicity of implementation. The media synchronization algorithm adopted [11] makes use of the stored media's property that its real-time equirement is not so critical as live media; this is consistent with the choice of TCP.

On these experimental systems, performance measurement of each scheme was made, and their features and application environments suitable for them were clarified. As a result, we confirmed that every scheme achieves good quality of media synchronization in the ATM LAN with the CBR service over a wide range of network loads, and in the ATM LAN with the UBR service up to a certain amount of network loads. We also found that the media synchronization capability is indispensable in the interconnected ATM-wireless LAN, which produces large delay jitters.

The present work is continuation of the research efforts about the set of lip synchronization schemes. The purpose of this paper is twofold. The first one is a performance comparison of the schemes in the set for *live media*, which are very important in many multimedia applications such as teleconferencing. The second purpose is the implementation of the schemes with RTP/RTCP on top of UDP, which is expected to be more suitable for live media than TCP. In this paper, we deal with live JPEG video and voice. We implement the four basic schemes utilizing no control function of RTCP and two variants with a function of RTCP, where dynamic resolution control of video [12], [13] is exercised. This paper makes a performance comparison of the six schemes.

As in [2] and [3], we constructed the three kinds of experimental systems for the live media. As a result of the experiment, we have found as before that in the ATM LAN there is no substantial performance difference among all the schemes, while the interconnected ATM-wireless LAN exhibits the difference. Therefore, this paper describes only the case of the interconnected ATM-wireless LAN.

The rest of the paper is organized as follows. Section 2 defines the four basic types of synchronization mechanisms and two variants. Section 3 gives an outline of the media synchronization algorithm used in this paper. Section 4 illustrates the configuration of the experimental system and a method of the performance measurement. Section 5 presents experimental results.

2. Basic Types of Synchronization Mechanism and Two Variants

We assume that the *synchronization layer*, which offers the media synchronization services, is located on top of the transport layer [11].

We now define the four basic types of the synchronization mechanism [2], [3]. Each type is characterized by two factors: the number of transport streams treated (single or multi) and the existence or nonexistence of the media synchronization capability at the destination. The *single-stream approach* [14] forms a single *composite stream* by interleaving voice and video *media units* (*MUs*)†† in the order of their timestamps and transmits them according to the timestamps. *Types 0* and *1* belong to this category. The *multi-stream approach* [11], [15]–[17] comprises *types 2* and *3*, which treat voice and video streams as separate transport streams. Types 0 and 2 do not carry out synchronization control at the destination; therefore, they are intended for the use in networks with guaranteed QOS of transport and lower layers such as ATM networks. On the other hand, types 1 and 3 exert the control so that they can be employed in any network. Note that ISO/IEC 11172-1, ISO/IEC 13818-1 and ITU-T H.223 correspond to either type 0 or type 1. The combination of vat and vic for the MBone is classified into type 2 except for intra-stream control of vat, and the scheme in [18] is a kind of type 3. The features of types 0 through 3 are summarized in Table 1.

We can consider some hybrid approach which consists of both interleaved and non-interleaved streams in the case where there are three or more media sources, e.g., text and graphic data, pointer data, as well as more than one audio/video stream. We are conducting experiments on a few cases of this kind of media sources.

In this paper, we also consider variants of types 1 and 3, which exert dynamic resolution control of video using RTCP in addition to media synchronization control; these enhanced versions of types 1 and 3 are called *type 1+* and *type 3+*, respectively. The dynamic resolution control dynamically changes the spatial or temporal resolution of video according to the network load.

† Experimental results of the UBR case are presented only in [3].

†† We use this term to represent the information unit for media synchronization.

Table 1 Features of the four basic types of lip synchronization mechanism.

item	single-stream approach		multi-stream approach	
	no control	with control	no control	with control
	type 0	type 1	type 2	type 3
networks intended for use	networks with small delay jitters (e.g., QOS guaranteed networks)	any network	networks with small delay jitters (e.g., QOS guaranteed networks)	any network
intra-stream synchronization	—	necessary	—	necessary
inter-stream synchronization	—	not necessarily required	—	necessary
location of media sources	the same location		any location	
QOS control of each medium	—	controlled together	—	separately controllable

The change of the resolution implies the change of the video source coding rate, and consequently it can be effective in controlling network traffic. We will compare the performance of the six schemes: types 0 through 3, 1+ and 3+.

3. Virtual-Time Rendering Algorithm for Media Synchronization

3.1 Principle

We can employ any algorithm for media synchronization in the schemes defined in the previous section. In this paper, we adopt the algorithm proposed in [11], which has been used in our experiments of media synchronization [2], [3], [12]–[17]. It is referred to as the *virtual-time rendering* (*VTR*) algorithm here.

The VTR algorithm attaches a timestamp to each MU at the *transport service access point* (*TSAP*) of the source when it is generated. The basic idea of the algorithm is quite simple; in addition to the actual time, we introduce a virtual-time which extends or shrinks according to the amount of delay jitters of MUs received at the destination, and media are rendered along the virtual-time axis. The time extension and shrinkage are realized in a form of modification of the *target output time*, which is the time when the destination should output an MU†. The application form of the modification depends on the kind of media treated, i.e., stored or live. In the case of stored media such as those in video-on-demand, the easiest way of the application is only time extension; that is, the target output time is postponed only (see [14] for its behavior). This implies that the resulting playback time becomes longer than the original recording time. However, if the amount of the time extension is not perceptible, the subjective quality of an output media stream can be improved. On the other hand, live media need both time extension and time shrinkage [15], [17], since the real-time property must be preserved. In addition to the modification of the target output time, the VTR algorithm also adopts skipping and pausing of MUs in order to recover from the media asynchrony as in many of other synchronization algorithm.

For inter-stream synchronization, the VTR algorithm selects a media stream as the *master stream* and the others as *slave streams*.

We can find similar ideas to the VTR algorithm in the literature. In [19], Anderson and Homsy present a concept of LTS (Logical Time System), which performs time-expansion and shrinkage only by skipping and pausing of MUs. An adaptive buffering scheme proposed by Xie et al. [20] defines the set-back and advance operations of the PlayOut Clock, which correspond to time-expansion and shrinkage, respectively, in the same way as the VTR algorithm.

3.2 Algorithm

Consider a media stream (say stream j), which can be either the composite stream (say $j = 0$), the voice stream ($j = 1$), or the video stream ($j = 2$). The single-stream approach deals only with stream 0, where streams 1 and 2 are substreams; in this paper, we adopt the *nondistinctive* scheme [14], which applies only intra-stream control to stream 0. In the multi-stream approach, we select voice (stream 1) as the master stream and video (stream 2) as the slave stream, since voice is more sensitive to intra-stream synchronization error than video. Thus, the multi-stream approach applies the intra-stream control to both stream 1 and stream 2, and the inter-stream control to stream 2.

Also, for voice, we employ the *graceful recovery policy* [11], which recovers from asynchrony gradually, and the *quick recovery policy* [16] for video. This is be-

†If there were no network delay jitter, it would be the arrival time of the MU, which is equal to its departure time at the source plus the network propagation delay. In reality, however, the jitters exist. Therefore, the target output time is modified in order to adapt to the delay variation.

cause we have found in our experiment that the graceful recovery policy reproduces live voice smoother than the quick one, while the latter realizes exact and quick synchronization of video.

The graceful-recovery VTR algorithm used in this paper is a revised one of that in [15], which is a special case (unicast case) of the algorithm presented in [17]. The quick recovery version of the algorithm is easily derived from the graceful recovery version; its derivation is straightforward. The reader is referred to the references for details of the algorithm.

4. Experimental System

On the three kinds of networks, we have implemented the schemes of types 0 through 3 with RTP/RTCP on top of UDP, utilizing no control function of RTCP, as well as types 1+ and 3+, which carry out dynamic resolution control of video using RTCP. We now describe the case of the interconnected ATM-wireless LAN.

4.1 System Configuration

Figure 1 illustrates the configuration of the experimental system, which is divided into the ATM-LAN part and the wireless-LAN part. The two parts are interconnected by a router (Cisco 4700-M).

The ATM-LAN part comprises two ATM switches (Cisco System's LightStream 1010 [21]), which are connected to each other through a 155 Mbps SONET multimode fiber-optic cable. In the experiment, however, we limit the transmission capacity between the two switches to 15 Mbps by means of the *traffic pacing mechanism* of the LightStream 1010 as in [2] and [3]†.

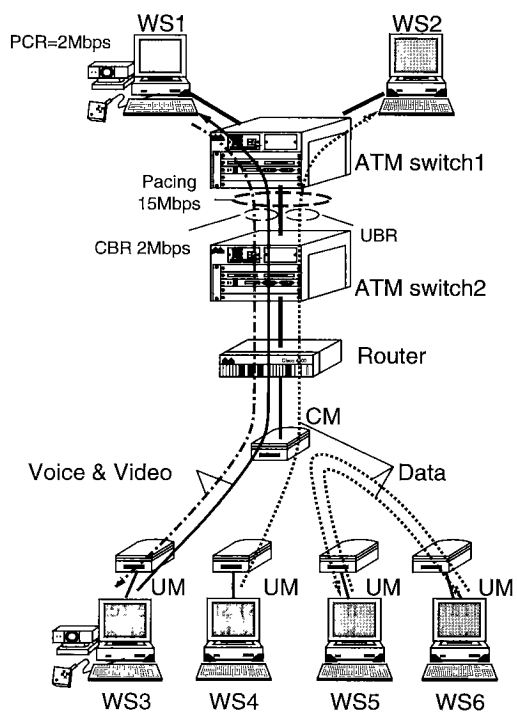


Fig. 1 Configuration of the experimental system.

The wireless-LAN part is a microcellular wireless LAN of Motorola's ALTAIR [22], which is a shared media LAN with a best effort service. It consists of a single *Control Module* (*CM*), which is connected to the router, and four *User Modules* (*UMs*). The CM is essentially the base station of the microcell; it acts as the relay point between UMs and between a UM and the router. The CM and UMs transmit and receive on the same frequency of 19 GHz, and the transmitter power is 80 mW. They share a transmission capacity of 15 Mbps in a way of TDD (Time Division Duplex). The MAC protocol for the UM-to-CM transmission is a reservation protocol with a slotted ALOHA reservation channel.

Two workstations (WS1 and WS2) are connected directly to an ATM switch, and four workstations (WS3 through WS6) are attached to the UMs. These workstations are SUN Ultra 1 except for WS3, which is Ultra 2. Each Ultra 1 workstation has the main memory of 64 Mbytes, but that of the Ultra 2 is 128 Mbytes. Both Ultra 1 and Ultra 2 run under Solaris 2.5 with OpenWindow 3.5. WS1 and WS2 each have a 155 Mbps ATM board (ZeitNet Z-ATM SBus Adapter/OC-3), in which AAL type 5 is used. WS1, as well as WS3, is also equipped with a video board (PowerVideo from Parallax Graphics, Inc.) handling a continuous bit stream of NTSC-standard video encoded by the JPEG compression scheme.

A *permanent virtual channel* (*PVC*) was set up between WS1 and the router as well as between WS2 and the router. We have adopted the method of *classical IP over ATM* in this environment. The UNI 3.0 signaling is used. The NNI signaling protocol we selected is the *Interim Interswitch Signaling Protocol* (*IISP*).

4.2 Method of the Experiment

As illustrated in Fig. 1, we used WS1 and WS3 as the video and voice source. Each of the two workstations input a video stream of a lady's head view and her voice stream from a video cassette recorder in order to generate the video and voice traffic of the same amount in each experimental run. Voice was encoded at a constant bit rate according to ITU-T G.711 μ-law. We define a single video frame as a video MU, whose size is variable because of JPEG, and a constant number of voice samples as a voice MU. The two workstations transferred the voice and video streams to each other

† As mentioned earlier, we constructed the three kinds of experimental systems. However, the capacity of 155 Mbps was too large for our experiment of the ATM LAN with the CBR service to examine the QOS guarantee capability of the switch. Therefore, we limited the capacity of the ATM-LAN part in all the experimental systems.

with the RTP/RTCP. Note that the RTP does not suppose the single-stream approach but the multi-stream one [7]. Thus, the use of the protocol may favor the latter; however, the influence of the favor seems small since the amount of voice traffic is much smaller than that of video traffic. The socket buffer capacity at each workstation in the single-stream approach is 51200 bytes. A workstation with the multi-stream approach assigns a capacity of 25600 bytes to each stream so that the buffer condition can be the same in the two approaches[†].

In types 1+ and 3+, we dynamically change the spatial resolution of video according to the network loads as follows[††]. WS1 and WS3 employ the value of the "fraction lost" field of sender report (SR)/receiver report (RR) RTCP packets [7] to identify the degree of congestion[†††]. This fraction is defined to be the number of RTP packets lost divided by the number of RTP packets expected since the previous SR or RR RTCP packet was sent. As a matter of fact, no RR RTCP packet was transmitted since both workstations were active senders [7]. In our experimental system, an RTCP packet is transmitted at intervals of 5 seconds. Thus, we can consider the network congested when the fraction is not zero. It should be noted that the fraction takes account of the loss of both voice and video packets in the single-stream approach, but that of only video packets in the multi-stream approach; that is, the single-stream approach may have somewhat larger fraction than the multi-stream approach.

With the *Q factor*, which corresponds to "Quantization factor" of the JPEG scheme, each of the two workstations (i.e., PowerVideo) changes the spatial resolution of video in types 1+ and 3+. As the Q factor increases, the spatial resolution becomes coarser, and so the average bit rate decreases. Therefore, we can alleviate the network congestion by increasing the Q factor. We set the Q factor at a value chosen from among the following three: 150, 300 and 450. The workstation starts with a Q factor of 150. When the workstation receives an SR/RR RTCP packet, it increases the Q factor by 150 at a time up to 450 if the value of "fraction lost" is larger than 0; otherwise, it decreases the Q factor by 150 at a time down to 150.

For types 0 through 3, on the other hand, we always set the Q factor at 150. The specifications of voice and video are shown in Table 2.

In order to guarantee QOS of the voice and video transmission in the ATM-LAN part, we selected the *constant bit rate* (*CBR*) service class for simplicity of experiment. The source workstation WS1 performed traffic shaping by setting the *peak cell rate* (*PCR*) to 2 Mbps, and the two ATM switches used the same value of PCR. For the traffic generated at WS3, the router carried out traffic shaping in the same way as WS1.

WS2 and WS4 through WS6 were utilized to generate a traffic flow of interference. We selected the *unspecified bit rate* (*UBR*) service class for the interference traffic between WS2 and WS4. WS4 sent fixed-size data messages of 1472 bytes each to WS2 under the UDP protocol at exponentially distributed intervals, and WS5 and WS6 also transmitted data messages to each other in the same way as WS4. The amount of the interference traffic was adjusted by changing the average of the interval by the same value at the three workstations at the same time.

In the experiment, we employed the same values of the thresholds for the synchronization algorithm as those in [17].

In the performance comparison between the single-stream and the multi-stream approaches, the conditions for the comparison should be identical in the two approaches. Regarding this matter, we first found in the experiment that the performance depends on the capture timing of MUs. This is because the difference in capture timing of MUs leads to the difference in transmission timing of MUs, which usually affects the performance. Since we have selected a single video frame as a video MU, the video capture timing is determined by the video program development tool (Xlib in our experiment). On the other hand, the capture timing of voice MUs is determined by the size of a voice MU, which we can change as a design parameter. Thus, the size of a voice MU has been chosen so that its average inter-MU time is the same as that of video (see Table 2). This implies almost the same capture timing in the two approaches. In addition to the capture timing, a function of the network delay estimation of the VTR algorithm [17] also introduces the difference in condition; therefore, the function was disabled in the experiment.

4.3 Performance Measures

In order to evaluate the performance of the media synchronization schemes, we should examine two aspects of the performance: the quality of media synchronization and the efficiency of information transfer.

As a performance measure of inter-stream synchronization quality, we adopt *mean square error*, which is defined as the average square of the difference between the output time of each slave MU (excluding skipped MUs) and its *derived output time* [16]. The derived output time of each slave MU is defined as the output time

[†]We also measured the performance of different assignment under the condition that the total capacity is 51200 bytes. We then obtained almost the same results unless the capacity assigned to one stream was too small.

[††]We also tried to change the temporal resolution instead of the spatial one. However, we found that we should perform the control along with traffic shaping (that is, control of packet transmission intervals) since the temporal resolution control hardly changes the burstiness of video traffic. We need further study on this matter.

[†††]We can also utilize other fields such as "interarrival jitter" [7] in SR/RR RTCP packets to this end; this is for further study.

Table 2 Specifications of voice and video.

item	voice	video (Q factor)		
		150	300	450
coding scheme	ITU-T G.711 μ-law	JPEG		
image size [pixels]	—	320×240		
average MU size [bytes]	420	5567	3712	3020
original average MU rate [MU/s]	17.0			
original average inter-MU time [ms]	58.8			
original average bit rate [kbps]	57	757	505	411

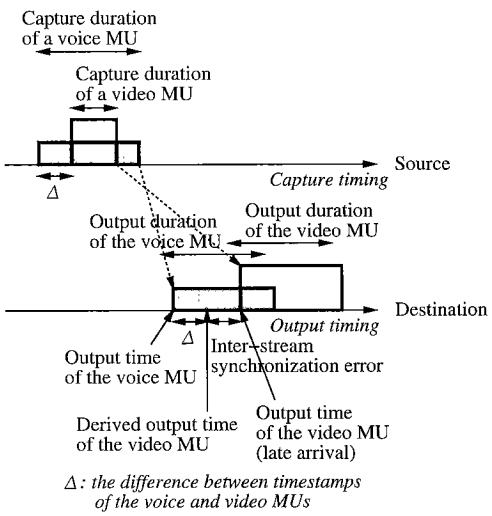


Fig. 2 Relation between capture timing and output timing.

of the corresponding master MU plus the difference Δ between the timestamps of the two MUs (see Fig. 2). For intra-stream synchronization quality, we here use the *coefficient of variation of output interval*[17]. The coefficient of variation of output interval denotes the smoothness of output. To evaluate the smoothness, we could use mean square error of intra-stream synchronization [2], [16]. However, the error cannot reflect the variation of output intervals in our experiment owing to the modification of the target output time. We should take the modification into account; we can do so with the coefficient of variation.

As the efficiency measures, we use the *throughput* and *average MU rate*. The throughput and the average MU rate are defined as the average number of information (both voice and video) bits and the average number of (either voice or video) MUs, respectively, output in a second at the destination.

Since we handle live media in this paper, another key measure is the *average MU delay*, which is the average time in seconds from the moment an MU is generated (i.e., from the beginning of capturing an MU) until the instant the MU is output.

In addition, we make subjective assessment of the output, when necessary.

5. Experimental Results

In this section, we examine measurement results to find out which type achieves the best performance and how the dynamic resolution control improves the quality of media synchronization. We measured the performance of each experimental run for 240 seconds (the total number of MUs of voice was equal to that of video, and it was 4077). In the following figures, we present the performance measured only at WS1.

5.1 Measurement Results

Figures 3 and 4 show the coefficient of variation of output interval for voice and video, respectively, as a function of data load for the six types, namely, types 0 through 3, and types 1+ and 3+. The data load is defined as the amount of the interference traffic generated by a single workstation†. Figure 5 depicts the mean square error of inter-stream synchronization. Furthermore, Figs. 6 through 10 display the throughput, the average MU rates of voice and video, and the average MU delays of voice and video, respectively, versus the data load.

In Figs. 3 and 4, we see that when the data load is smaller than about 0.55 Mbps, types 0 and 2 provide the largest coefficient of variation of output interval and the second largest, respectively, among all the types. In this area, we note that type 1 has somewhat larger values of the coefficient of variation than type 3. The coefficient of variation of type 1+ is also slightly larger than that of type 3+ in the whole range of the data load considered here. This is because a voice/video MU in types 1 and 1+ may wait for a short while before the MU is put into the composite stream; that is, this is caused by the interleaving.

We also observe in Figs. 3 and 4 that when the data load is lighter than about 0.46 Mbps, the difference in the coefficient of variation between types 1 and 1+ and that between types 3 and 3+ are not so large.

† The amount of the interference traffic in the ATM LAN part is equal to the data load itself, and that in the UM-to-CM transmission of the wireless LAN part is three times the data load.

Furthermore, we see that when the data load exceeds around 0.55 Mbps, the coefficients of variation of types 0 through 3 start to increase largely and the difference among the types becomes smaller. This is because the number of lost MUs increases; we can confirm this in Figs. 7 and 8. For the same heavy data loads, however, types 3+ and 1+ achieve the smallest and the second smallest coefficient of variation, respectively. This is one of the effects of the dynamic resolution control.

We also made the subjective assessment of the smoothness. However, we perceived no substantial difference in the smoothness of output among the six types for the data loads up to about 0.55 Mbps. When the data load exceeds the value, the voice quality of types 0 through 3 started to degrade, and we noticed a little jerky video of the types; in types 1+ and 3+, the output quality of voice and video was not so largely damaged.

We can confirm in Fig. 5 that the schemes with the control (i.e., types 1, 3, 1+ and 3+) provide smaller mean square error of inter-stream synchronization than those without any control (types 0 and 2) for the data loads less than about 0.66 Mbps. However, even types 0 and 2 have at most about 600 ms^2, which is much

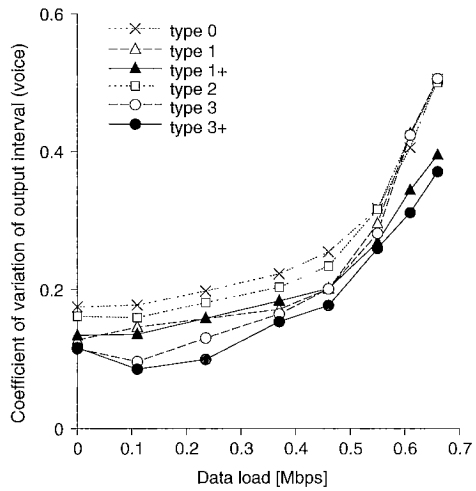


Fig. 3 Coefficient of variation of output interval versus data load for voice.

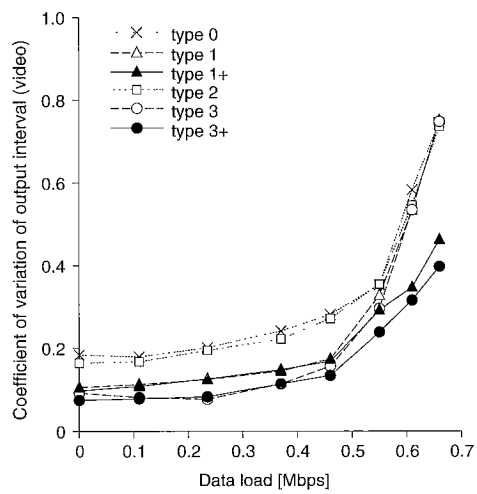


Fig. 4 Coefficient of variation of output interval versus data load for video.

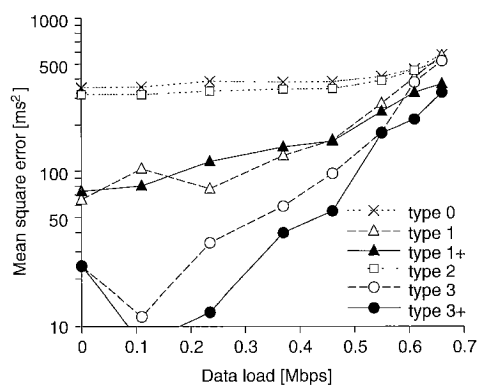


Fig. 5 Mean square error of inter-stream synchronization versus data load.

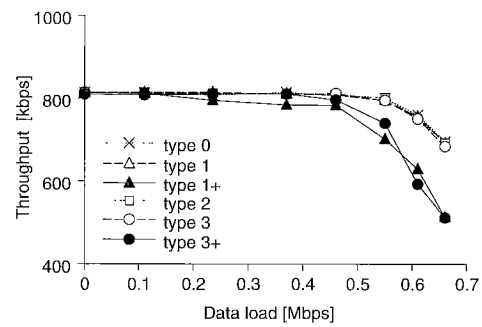


Fig. 6 Throughput versus data load.

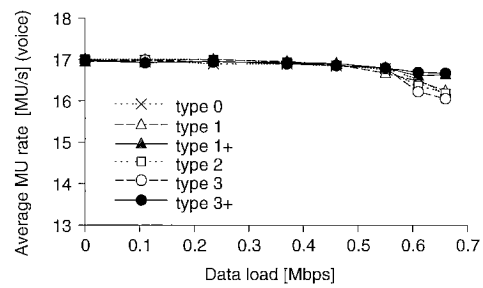


Fig. 7 Average MU rate of voice versus data load.

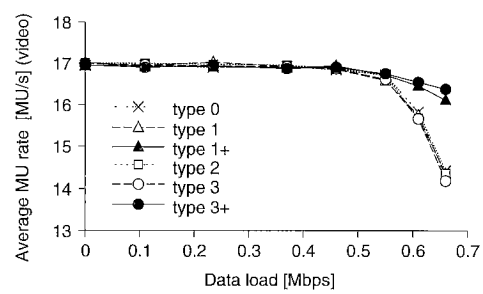


Fig. 8 Average MU rate of video versus data load.

smaller than the maximum allowable mean square error of inter-stream synchronization, that is, 6400 ms^2 (= 80^2 ms^2) [1], [16]. The reason why the difference between no-control and control schemes is small is that the VTR algorithm for live media stores only few MUs before output so as not to increase MU delay unnecessarily [11], [15], [17]; for stored media, we show in [2] that the difference in inter-stream synchronization error among the four basic types becomes large. The small difference is one of the features of the VTR algorithm for live media; all the types attain good quality of inter-stream synchronization. In fact, we hardly perceived the difference among the six types.

In Fig. 6, we find that the throughput of all the schemes starts to decrease largely when the data load exceeds about 0.46 Mbps. This is due to loss of packets; we found that no MU was skipped. We notice in Fig. 6 that types 1+ and 3+ have the smallest or the second smallest throughput among all the types for the data loads larger than about 0.24 Mbps. However, we observe in Figs. 7 and 8 that the average MU rates of types 3+ and 1+ are larger than those of types 0 through 3 for the data loads heavier than 0.55 Mbps. To judge which of the two measures is more important for video and voice, we have to rely on subjective assessment. We subjectively found that the average MU rate of video is more important than the video throughput in this case. Note that the resolution of voice was not changed in our experiment. Thus, a higher average MU rate of voice means higher throughput.

In Figs. 9 and 10, we see that types 0 and 2 have smaller average MU delay than types 1 and 3 when the data load is lighter than about 0.46 Mbps. We can confirm that types 1+ and 3+ outperform the other types for the data loads heavier than around 0.55 Mbps.

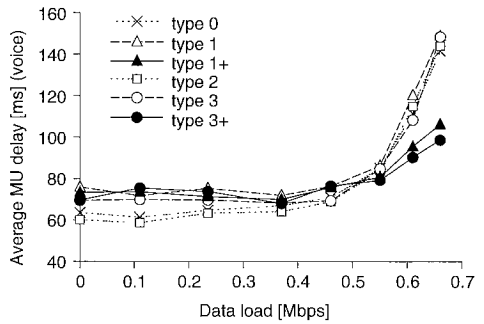


Fig. 9 Average MU delay of voice versus data load.

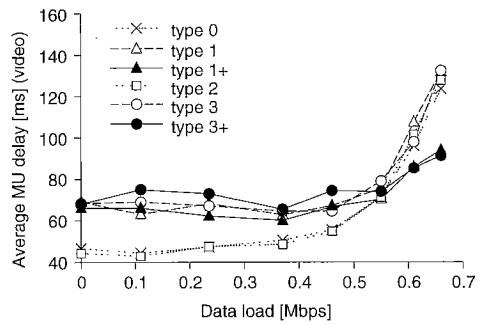


Fig. 10 Average MU delay of video versus data load.

5.2 Discussions and Remarks

From the observation in the previous subsection, we can say that the multi-stream approach is somewhat superior to the single-stream one in our experiment. For stored media, we show in [2], [3] and [12] that the multi-stream approach is better than the single-stream approach in many cases; the performance difference for heavy traffic is larger than that in the case of live media. Regarding inter-stream synchronization of stored media for heavy traffic, on the other hand, type 1 provides better quality than type 3 because of its interleaving mechanism. For live media, however, all the types attain high quality of inter-stream synchronization owing to the real-time property.

In addition, we see that the dynamic resolution control improves the performance of live media transmission at heavy loads as well as stored media transmission [12].

Furthermore, we carried out the experiment in another environment; we changed the capture timing of video by employing another video program development tool (XView Toolkit) instead of that (Xlib) in the above experiment†. We examined the influence of the capture timing of MUs on the performance for various values of the voice MU size. As a result, we found that the quantitative relation between the single-stream and multi-stream approaches changes according to the capture timing of MUs. However, the difference is small from a subjective assessment point of view.

Note that media synchronization control can be effective for heavily loaded networks. Under the heavy traffic condition, however, the use of RTP on top of UDP incurs frequent loss of MUs. This implies not only the bad quality of media synchronization but also the overall performance degradation. This deficiency can be fixed partly by means of MU retransmission within a limited time interval so as to preserve the real-time property. It is reasonable that a lost MU is retransmitted before the output deadline of the MU. In [23] and [24], the necessity of such a retransmission function is asserted. The retransmission method is particularly important for interframe-coded video such as MPEG. For stored media, on the other hand, we can resort to the retransmission method more extensively, since the real-time requirement is not so severe††. Whether the MU retransmission is carried out or not affects the per-

† It takes about 10 ms to capture each video MU when we use Xlib, while around 44 ms for XView Toolkit.

†† On the basis of this idea, we utilize the retransmission capability of TCP for stored media in [2], [3] and [12].

formance relations among the four basic types, as we have seen; that is, since the retransmission increases delay jitters, the relations may change. For heavy traffic, RTP/UDP achieves small delay jitters at the expense of MU loss, whereas TCP attains no loss of MU at the expense of delay jitters. Thus, we should select the retransmission capability according to the requirement of target applications. We need further study on this issue.

6. Conclusions

We made a performance comparison of the six schemes (types 0 through 3, and types 1+ and 3+) for live voice and video by experiment. We first confirmed that when network load is light, all the schemes can provide sufficient performance as in the case of stored media [2], [3]; however, the effectiveness of the media synchronization control was demonstrated. For heavy loads, the performance of types 0 through 3 degrades owing to MU loss.

We also found that the multi-stream approach is somewhat superior to the single-stream approach; however, the performance difference is smaller than that in the case of stored media [2], [3]. This is mainly due to the difference between RTP/UDP and TCP. In addition, we noticed that the quantitative relation between the single-stream and multi-stream approaches changes according to the capture timing of MUs. However, the difference between the two was small from a subjective point of view. In order to clarify the relation in more detail, we need to carry out experiments under more diverse traffic conditions for a variety of the capture timing of MUs.

Furthermore, we noticed that the dynamic resolution control can improve the smoothness of output as well as average MU rate and average MU delay.

In our experiment, we handled only UDP traffic. Sawashima et al. examine the influence of TCP flow control on the UDP packet loss by simulation in [25]. They conclude that reducing the UDP transmission rate does not result in reducing the UDP packet loss. Therefore, we plan to carry out an experiment in the case where UDP and TCP traffic is conveyed simultaneously.

In addition, as one of the next subjects in our research, we will compare the six schemes for live MPEG video.

It should be noted that in the multi-stream approach, we can select a transport protocol suitable for each stream separately; this is one of the advantages of the approach. Thus, we need to make an experiment by employing a transport protocol suitable for each stream in the case where we deal with media sources such as text and pointer data as well as voice and video streams.

We are also planning to study how to change the video resolution dynamically according to network loads and how to utilize parameters in RTCP packets in further detail.

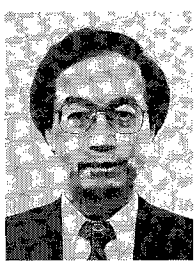
Acknowledgment

This work was supported by Telecommunications Advancement Organization of Japan.

References

[1] G. Blakowski and R. Steinmetz, "A media synchronization survey: Reference model, specification, and case studies," IEEE J. Select. Areas Commun., vol.14, no.1, pp.5–35, Jan. 1996.
[2] S. Tasaka and Y. Ishibashi, "Stored media synchronization schemes in ATM and wireless networks: A performance comparison," Proc. IEEE ICUPC'97, pp.766–772, Oct. 1997.
[3] S. Tasaka and Y. Ishibashi, "Media synchronization in heterogeneous networks: Stored media case," IEICE Trans. Commun., vol.E81–B, no.8, pp.1624–1636, Aug. 1998.
[4] ISO/IEC 11172-1, "Information technology—Coding of moving pictures and associated audio for digital storage media at up to about 1.5 Mbps, Part 1: Systems," International Standard, 1993.
[5] ISO/IEC 13818-1, "Information technology—Generic coding of moving pictures and associated audio information: Systems," International Standard, Nov. 1994 (also, ITU-T Recommendation H.222.0).
[6] ITU–T Recommendation H.223, "Multiplexing protocol for low bit rate multimedia communication," March 1996.
[7] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, "RTP: A transport protocol for real–time applications," RFC 1889, Feb. 1996.
[8] V. Jacobson and S. McCanne, "Visual Audio Tool," Lawrence Berkeley Laboratory, Software on–line, (ftp://ftp.ee.lbl.gov/conferencing/vat).
[9] S. McCanne and V. Jacobson, "vic: A flexible framework for packet video," Proc. ACM Multimedia'95, pp.511–522, Nov. 1995.
[10] ITU–T Recommendation H.225.0, "Media stream packetization and synchronization for visual telephone systems on non–guaranteed quality of service LANs," Nov. 1996.
[11] Y. Ishibashi and S. Tasaka, "A synchronization mechanism for continuous media in multimedia communications," Proc. INFOCOM'95, pp.1010–1019, April 1995.
[12] S. Tasaka, H. Nakanishi, and Y. Ishibashi, "Dynamic resolution control and media synchronization of MPEG in wireless LANs," Conf. Rec. GLOBECOM'97, pp.138–144, Nov. 1997.
[13] S. Tasaka and H. Imura, "Dynamic resolution control of stored video traffic in a wireless LAN," Proc. IEEE PIMRC'96, pp.153–157, Oct. 1996.
[14] S. Tasaka, Y. Ishibashi, and H. Imura, "Stored media synchronization in wireless LANs," Conf. Rec. GLOBECOM'96, pp.1904–1910, Nov. 1996.
[15] Y. Ishibashi, S. Tasaka, and A. Tsuji, "Performance measurement of a live media synchronization mechanism in an ATM network," Conf. Rec. ICC'96, pp.1348–1354, June 1996.
[16] Y. Ishibashi, S. Tasaka, and E. Minami, "Performance measurement of a stored media synchronization mechanism: Quick recovery scheme," Conf. Rec. GLOBECOM'95, pp.811–817, Nov. 1995.
[17] Y. Ishibashi and S. Tasaka, "A group synchronization mechanism for live media in multicast communications," Conf. Rec. GLOBECOM'97, pp.746–752, Nov. 1997.
[18] I. Kouvelas, V. Hardman, and A. Watson, "Lip synchronisation for use over the Internet: Analysis and implementa-

tion," Conf. Rec. GLOBECOM'96, pp.893–898, Nov. 1996.
[19] D.P. Anderson and G. Homsy, "A continuous media I/O server and its synchronization mechanism," IEEE Computer, vol.24, no.10, pp.51–57, Oct. 1991.
[20] Y. Xie, C. Liu, M.J. Lee, and T.N. Saadawi, "Adaptive multimedia synchronization in a teleconference system," Conf. Rec. ICC'96, pp.1355–1359, June 1996.
[21] "LightStream 1010 ATM Switch User Guide," Cisco Systems, Inc., 1996.
[22] D. Buchholz, P. Odlyzko, M. Taylor, and R. White, "Wireless in-building network architecture and protocols," IEEE Network, vol.5, pp.31–38, Nov. 1991.
[23] H. Xie, P. Narasimhan, R. Yuan, and D. Raychaudhuri, "Data link control protocols for wireless ATM access channels," Proc. IEEE ICUPC'95, pp.753–757, Oct. 1995.
[24] T. Hasegawa, T. Hasegawa, T. Kato, and K. Suzuki, "Applying reliable data transfer protocol to real time video retrieval system," IEICE Trans. Commun., vol.E80–B, no.10, pp.1482–1492, Oct. 1997.
[25] H. Sawashima, Y. Hori, H. Sunahara, and Y. Oie, "Characteristics of UDP packet loss: Effect of TCP traffic," Proc. INET'97, Engineering 3–1, June 1997.



Shuji Tasaka received the B.S. degree in electrical engineering from Nagoya Institute of Technology, Nagoya, Japan, in 1971, and the M.S. and Ph.D. degrees in electronic engineering from the University of Tokyo, Tokyo, Japan, in 1973 and 1976, respectively. Since April 1976, he has been with Nagoya Institute of Technology, where he is now a Professor in the Department of Electrical and Computer Engineering. In the 1984–1985 academic year, he was a Visiting Scholar in the Department of Electrical Engineering at the University of California, Los Angeles. His current research interests include wireless networks, high–speed networks and multimedia communication protocols. He is the author of a book entitled *Performance Analysis of Multiple Access Protocols* (Cambridge, MA: The MIT Press, 1986). Dr. Tasaka is a member of the IEEE, ACM and Information Processing Society of Japan.

Yutaka Ishibashi received the B.S., M.S. and Ph.D. degrees from Nagoya Institute of Technology, Nagoya, Japan, in 1981, 1983 and 1990, respectively. From 1983 to 1993, he was with NTT Laboratories. In 1993, he joined Nagoya Institute of Technology, where he is currently an Associate Professor in the Department of Electrical and Computer Engineering. His research interests include media synchronization mechanisms, multimedia communication protocols, and high–speed network architecture. Dr. Ishibashi is a member of the IEEE, ACM, Information Processing Society of Japan, and the Institute of Image Information and Television Engineers.