Inter-destination multimedia synchronization: A contemporary survey

Dimitris Kanellopoulos

Abstract — **The advent of social networking applications, media streaming technologies, and synchronous communications has created an evolution towards dynamic shared media experiences. In this new model, geographically distributed groups of users can be immersed in a common virtual networked environment in which they can interact and collaborate in real-time within the context of simultaneous media content consumption. In this environment, intra-stream and inter-stream synchronization techniques are used inside the consumers' playout devices, while synchronization of media streams across multiple separated locations is required. This synchronization is known as multipoint, group or Inter-Destination Multimedia Synchronization (IDMS) and is needed in many applications such as social TV and synchronous e-learning. This survey paper discusses intra- and inter-stream synchronization issues, but it mainly focuses on the most well-known IDMS techniques that can be used in emerging distributed multimedia applications. In addition, it provides some research directions for future work.**

Index Terms — **Multimedia synchronization, IDMS, multipoint synchronization, RTP/RTCP**

Abbreviations

AMP	Adaptive media playout
DCS	Distributed control scheme
ETSI	European Telecommunications Standards Institute for Advanced Networking
IDMS	Inter-destination multimedia synchronization
IETF	Internet Engineering Task Force
M/S	Master/slave receiver scheme
MU	Media unit
QoE	Quality of experience
QoS	Quality of service
RTP	Real-Time Transport Protocol
RTCP	RTP Control Protocol
SMS	Synchronization maestro scheme
TISPAN	Telecoms & Internet Converged Services and Protocols
VTR	Virtual-time rendering synchronization algorithm

I. Introduction

Nowadays, novel media consumption paradigms such as social TV and synchronous e-learning are enabling users to consume multiple media streams at multiple devices together and having dynamic shared media experiences [1]. In order to provide an enjoyable dynamic shared media experience, various technical challenges must be faced. Examples are synchronization, Quality of Service (QoS), Quality of Experience (QoE), scalability, user mobility, intelligent media adaptation and delivery, social networking integration, privacy concerns, and user preferences management [2]. This survey focuses on the synchronization of media streams across multiple separated locations/consumers. This synchronization is known as *multipoint, group* or *Inter-Destination Multimedia Synchronization* (IDMS) and is required in many use cases such as social TV, synchronous e-learning, networked quiz shows, networked real-time multiplayer games, multimedia multi-point to multi-point communications, distributed tele-orchestra, multi-party multimedia conferencing, presence-based games, conferencing sound reinforcement systems, networked stereo loudspeakers, game-show participation, shared service control, networked video wall, and synchronous groupware [3]. These use cases require media synchronization as there are significant delay differences between the various delivery routes for multimedia services (e.g., media streaming). Meanwhile, broadcasters have started using proprietary solutions for over-the-top media synchronization such as media fingerprinting or media watermarking technologies. Given the commercial interest in media synchronization and the disadvantages of proprietary technologies, consumer-equipment manufacturers, broadcasters, and telecom and cable operators have started developing new standards for multimedia synchronization.

An important feature of multimedia applications is the integration of multiple media streams that have to be presented in a synchronized fashion [4]. *Multimedia synchronization* is the preservation of the temporal constraints within and among multimedia data streams at the time of playout. Temporal relations define the temporal dependencies between media objects [5]. An example of a temporal relation is the relation between a video and an audio object which are recorded during a concert. If these objects are presented, the temporal relation during the presentations of the two media objects must correspond to the temporal relation at the time of recording. *Discrete* media like text, graphics, and images are time-independent media objects, while the semantic of their content does not depend upon a presentation to the time domain. A discrete media object is frequently presented using one presentation unit. Conversely, a time-dependent media object

Manuscript received August 20, 2018, revised December 21, 2018.
D. Kanellopoulos is with the Department of Mathematics, University of Patras, GR 26500 Greece (e-mail: d_kan2006@yahoo.gr)

DOI: 10.36244/ICJ.2019.1.2

is presented as a *continuous* media stream in which the presentation durations of all *Media Units* (MUs) are equal [4]. For example, a video consists of a number of ordered frames, where each of these frames has a fixed presentation duration. Most of the components of a multimedia system support and address temporal synchronization. These components may include the operating system, communication subsystem, databases, documents, and even applications. In distributed multimedia systems, networks introduce random delays in the delivery of multimedia information. Actually, there are some sources of *asynchrony* that can disrupt synchronization [3],[6]:

- *Network Jitter*. This is an inherent characteristic of best-effort networks like the Internet.
- *Local Clock Drift* arises when clocks at users run at different rates. Without a synchronization mechanism, the asynchrony will gradually become more and more serious.
- *Different Initial Collection Times*. Let us consider two media sources, one providing voice and the other video. If these sources start to collect their MUs at different times, the playback of the MUs of voice and video at the receiver loses semantic meaning.
- *Different Initial Playback Times*. If the initial playback times are different for each user, then asynchrony will arise.
- *Network topology changes* and *unpredictable delays*. In mobile ad hoc networks (MANETs), the preservation of temporal dependencies among the exchanged real-time data is mainly affected: (1) by the asynchronous transmissions; (2) by constant topology changes; and (3) by unpredictable delays.
- *The encoding used*. If media streams are encoded differently, the decoding times at receiver may vary considerably.

Delay is a simple constraint when users are consuming non-time sensitive content from content-on-demand networks. However, delay and jitter (variation of end-to-end delay) become serious constraints when an interaction between the user and the media content (or interaction between different users) is needed. In those applications, delay and jitter could be harmful to the QoE and may prevent the inclusion of higher forms of interactivity in various group-shared services. Consequently, many multimedia synchronization techniques have been proposed to ensure synchronous sharing of content among users temporarily collocated, either being spatially distributed or even sharing a physical space.

This paper presents the basic control schemes for IDMS and discusses IDMS solutions and IDMS standardization efforts for emerging distributed multimedia applications. The structure of the paper is organized as follows. Section II discusses intra-stream and inter-stream synchronization issues. Section III reviews well-known schemes for IDMS, while Section IV presents standardization efforts on IDMS as well as effective IDMS solutions. Finally, Section V concludes the paper and gives directions for future work.

II. BACKGROUND

A. Intra-stream Synchronization

Intra-stream (also known as *intra-media* or *serial*) synchronization is the reconstruction of temporal relations between the MUs of the same stream. An example is the reconstruction of the temporal relations between the single frames of a video stream. The spacing between subsequent frames is dictated by the frame production rate. For instance, for a video with a rate of 40 frames per second, each of these frames must be displayed for 25 ms. Jitter may destroy the temporal relationships between periodically transmitted MUs that constitute a real-time stream, thus hindering the comprehension of the stream. *Playout adaptation algorithms* undertake the labor of the temporal reconstruction of the stream. This reconstruction is referred to as the '*restoration of its intra-stream synchronization quality*' [7]. *Adaptive Media Playout* (AMP) improves the media synchronization quality of streaming applications by regulating the playout time interval among MUs at a receiver. To mitigate the effect of the jitter, MUs have to be delayed at the receiver in order a continuous synchronized presentation to be achieved. Therefore, MUs have to be stored in a buffer and the size of this buffer may correspond to the amount of jitter in the network. As the synchronization requirements can vary according to the application on hand, we must control the individual sync requirements (i.e., delay sensitivity, error tolerance etc.) for each media separately. To this direction, Park and Choi [7] investigated an efficient and flexible multimedia synchronization method that can be applied at intra-media synchronization in a consistent manner. They proposed an adaptive synchronization scheme based on: (1) the delay offset; and (2) the playout rate adjustment that can match the application's varying sync requirements effectively. Park and Kim [8] introduced an AMP scheme based on a discontinuity model for intra-media synchronization of video applications over best-effort networks. They analyzed the temporal distortion (i.e., discontinuity) cases such as playout pause and skip, to define a unified discontinuity model. Finally, Laoutaris and Stavrakakis [9] surveyed the work in the area of playout adaptation. Actually, the problem of intra-stream synchronization has been solved efficiently as many intra-stream synchronization techniques in the literature achieved to avoid receiver buffer underflow and overflow problems.

B. Inter-stream Synchronization

Inter-stream (also known as *inter-media* or *parallel*) *synchronization* is the problem of synchronizing different but related streams. Precisely, it is the preservation of the temporal dependencies between playout processes of different, but correlated, media streams involved in a multimedia session. An example of inter-stream synchronization is the *Lip synchronization* that refers to the temporal relationship between an audio and a video stream for the particular case of human speaking [10]. Fig. 1 shows an example of the temporal relations in inter-stream synchronization.

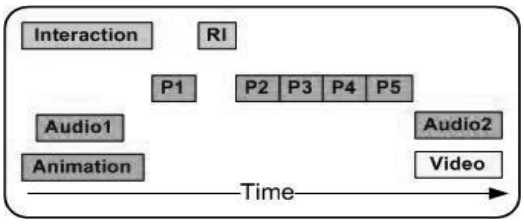

Fig. 1. An inter-media synchronization example

It starts with an animation (Animation) which is partially commented using an audio sequence (Audio1). Starting the animation presentation, a multiple-choice question is presented to the user (Interaction). If the user has made a selection, a final picture (P1) is shown. Then, the replay of a recorded user interaction (RI) follows with a slide sequence (P2-P5), and a lip-synchronized audio/video sequence (Audio2 and Video). Blakowski and Steinmetz [5] illustrated the main specification methods that can describe synchronization scenarios. These methods are *interval-based specification, control flow-based specification, axes-based synchronization, event-based synchronization, scripts,* and *comments*. Among them, *Scripts* is one of the most powerful methods that describe the majority of synchronization scenarios. Scripts often become full programming languages extended by timing operations. Such language is SMIL (*Synchronized Multimedia Integration Language*) that became a standard (W3C SMIL 3.0, Dec. 2008). Scripts may rely on different specification methods. A typical script is a script that is based on a hierarchical method and supports three main operations: serial presentation, parallel presentation, and the repeated presentation of a media object. Below, we write a script for the application example depicted in Fig. 1.

```
activity Picture            Picture1("picture1.jpeg");
activity DigAudio           Audio1("animation.au");
activity RTAnima            Animation("animation.ani");
activity Xrecorder          Recorder("window.rec");
activity StartInteraction   Selection;
activity Picture            Picture2("picture2.jpeg");
activity Picture            Picture3("picture3.jpeg");
activity Picture            Picture4("picture4.jpeg");
activity Picture            Picture5("picture5.jpeg");

activity DigAudio           Audio2("audio2.au");
activity SMP                Video("video.smp");

script AniComment CC = Animation &
Audio1.Translate(GR);
script Picture_sequence
     4Pictures =          Picture2.Duration(3)>>
                          Picture3.Duration(3)>>
                          Picture4.Duration(3)>>
                          Picture5.Duration(3);
script Lipsynch  AV = Audio2 & Video;
script Multimedia
  Application_example {
                          ((Selection Picture1) & CC >>
                          Record.UI >>
                          4Pictures>>
                          AV
                 }
```

Fig. 2. A script for the example depicted in Fig. 1

The symbols & and >> denote parallel and serial presentation correspondingly. Note that *activities* and *subscripts* compose the *script*. A *synchronization specification* of a multimedia object describes all temporal dependencies of the objects included in this object. It is comprised of:

- Intra-object synchronization specifications for the media objects of the presentation.
- QoS descriptions for intra-object synchronization.
- Inter-object synchronization specifications for media objects of the presentation.
- QoS descriptions for inter-object synchronization.

To achieve inter-stream synchronization, various algorithms have been applied. Also, there are several types of intra-stream synchronization control such as *Skipping* [11], *Buffering* [11], *Adaptive Buffer Control* (ABC) [12], *Queue Monitoring* (QM) [13], *Virtual-Time Rendering* (VTR) [14], and *Media Adaptive Buffering* [15]. Boronat et al. [16] have reviewed and compared the most powerful inter-stream synchronization algorithms. The building blocks of these algorithms are the synchronization techniques utilized both at the sender and the receiver sides. These algorithms can use multiple synchronization techniques to achieve synchronization aim even from different categories [17].

C. Classification of Inter-Media Techniques

Boronat et al. [16] categorized synchronization techniques according to the '*location*', '*content*', '*sync information used*' and '*purpose*'.

- **Location of synchronization technique:** The synchronization control can be performed either by source or receiver. If control is performed by the source, most of the time it will require some feedback information from the receiver. The receiver will tell the source about the degree of asynchrony at the current instance.
- **Live vs. Synthetic synchronization (Type of Media):** In *live* media, the temporal relations are exactly reproduced at a presentation as they existed during the capture process. Synthetic synchronization techniques are used for *stored* media.
- **Information used for synchronization technique:** The information included in the MU for the synchronization purpose can be different like *timestamp*, *sequence number*. Some techniques use either sequence number or timestamp, while others may use both. For example, the Real-Time Transport Protocol (RTP) provides timestamps to synchronize different media streams.
- **Purpose of synchronization technique:** The techniques can be divided into four subcategories with respect to their purpose:
  1. The *basic control* techniques are required in almost all synchronization algorithms. Examples are adding synchronization information in MUs at the source and buffering of MUs at the receiver.
  2. The *common control* techniques can be applied in both ways.
  3. The *preventive control* techniques are used to prevent the asynchrony in the streams. Preventive mechanisms minimize latencies and jitters and may involve disk-reading scheduling algorithms, network

transport protocols, operating systems, and synchronization schedules.

4. The *reactive control* (or *corrective*) techniques are designed to recover synchronization in the presence of synchronization errors. An example of corrective mechanisms is included in the Stream Synchronization Protocol (SSP).

Based on these criteria, Table I shows a classification of inter-media techniques.

TABLE I.
CLASSIFICATION OF INTER-MEDIA TECHNIQUES

Technique	Location	Description
Basic Control	Source control	• Add information useful for synchronization: timestamps, sequence numbers (identifiers), event information and/or source identifiers.
	Receiver control	• Buffering techniques
Common Control	Source control	• Skip or pause MUs in the transmission process. • Advance the transmission timing dynamically. • Adjust the input rate. • Media Scaling.
	Receiver control	• Adjust the playout rate. • Data Interpolation.
Preventive Control	Source control	• Initial playout instant calculation. • Deadline-based transmission scheduling. • Interleave MUs of different media streams in only one transport stream.
	Receiver control	• Preventive skips of MUs (e.g., discardings) and/or preventive pauses of MUs (repetitions, insertions or stops). • Change the buffering waiting time of MUs. • Enlarge or shorten the silence periods of the streams.
Reactive Control	Source control	• Adjust the transmission timing. • Decrease the media streams transmitted. • Drop low-priority MUs.
	Receiver control	• Reactive skips (eliminations or discardings) and/or reactive pauses (repetitions, insertions or stops). • Make playout duration extensions or reductions (playout rate adjustments). • Use of virtual time with contractions or expansions. • Master/slave scheme. • Late event discarding (Event-based). • Rollback techniques (Event-based)

III. INTER-DESTINATION MULTIMEDIA SYNCHRONIZATION

Inter-Destination (also known as *Inter-Receiver* or *group* or *multipoint*) *Multimedia Synchronization* (IDMS) has been gaining popularity due to the rise of social networking applications. IDMS involves the simultaneous synchronization of one or more playout receivers of one or several media streams at geographically distributed receivers to achieve fairness among them. Fairness implies that during a multimedia session, all the receivers must play the same MU at each media stream. For example, in a networked video wall scenario, wherein users are watching an on-line football match, all users should experience the goal event almost simultaneously (to have a fair shared experience).

Existing distribution technologies do not handle the IDMS problem in an optimal way. Thus, additional adaptive techniques must be provided to meet the IDMS synchronization requirements in practical content delivery networks. The levels of required synchrony among the receivers depend on the application on hand. However, the exact ranges of asynchrony levels (which could be tolerated by users for emerging distributed applications) have not sufficiently determined yet [3].

Akyildiz and Yen [6] introduced group synchronization protocols for real-time multimedia applications including teleconference, tele-orchestration, and multimedia on demand services. Their protocols achieve synchronization for all configurations (one-to-one, one-to-many, many-to-one, and many-to-many), and do so without prior knowledge of the end-to-end delay distribution, or the distribution of the clock drift. The only a-priori knowledge the protocols require is an upper bound on the end-to-end delay. Boronat et al. [16] reviewed the most-known multimedia group and inter-stream synchronization approaches. Group synchronization techniques can be classified at three schemes (discussed later). These schemes are based on the *Virtual-Time Rendering* (VTR) *media synchronization algorithm* to determine the output timing of each MU so that the timing can be the same at all the destinations. VTR algorithm is applicable to networks with unknown delay bounds. It makes use of globally synchronized clocks. VTR consists of the dynamic adjustment of the MUs rendering-time according to the network condition. For a better understanding of these schemes, let us consider that M sources and N destinations/receivers are connected through a network. MUs of M different streams have been stored with timestamps in M sources, and they are broadcasted to all the receivers. The timestamp contained in a MU indicates its generation time. The streams often fall into a *Master stream* and *Slave streams*. At each receiver, the slave streams are synchronized with the Master stream by using an inter-media synchronization mechanism.

A. Master/Slave (M/S) Receiver Scheme

In M/S scheme [18], the receivers are categorized into one *Master* receiver and *Slave* receivers. Multiple streams are received at each receiver and one of these streams acts as Master stream in order inter-media synchronization to be achieved at each receiver (Fig. 3).

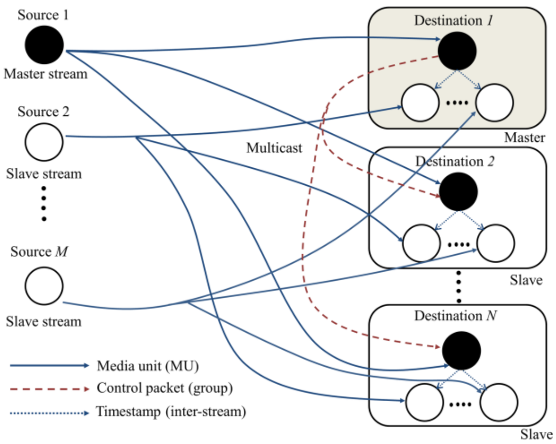

Fig. 3. Master/Slave Receiver Scheme [19].

None of the slave receivers send any feedback information about the timing of the playout processes. It only adjusts the playout timing of MUs to that of the Master receiver. Only the Master receiver sends (multicasts) its playout timing to all the other (slave) receivers. The Master receiver controls and computes the presentation time of the MUs according to its own state of the received stream data. Group synchronization is achieved by adjusting the presentation time of the MUs of master stream at the slave receivers to that of the Master receiver. Therefore, the slave receivers should present MUs at the same timing as the Master receiver. The synchronization of the slave receivers is achieved as follows:

- The Master receiver multicasts a control packet to all slave receivers. This control packet includes the presentation time of its first MU of the master stream. This process is called "*initial presentation adjustment*".
- When the target presentation time of the Master receiver changes, the Master receiver notifies all the slaves about this modification by multicasting a control packet. This control packet contains the amount of time that is modified and the sequence number of the MU for which the target presentation time has been changed.
- The Master receiver periodically multicasts proper control packets to accommodate the newly joined slave receivers.

Boronat et al. [20] presented the M/S scheme by extending the RTP/RTCP (Real-time Transport Protocol/ RTP Control Protocol) messages for containing the synchronization information. Fig. 3 presents the different type of message exchanges in the basic M/S scheme. The advantage of the M/S technique is its simplicity and the decreased amount of information exchange (i.e., control packets) to support group synchronization. However, the selection of the Master receiver can influence the performance of the scheme because slave receivers must present MUs at the same timing as the Master receiver. If the fastest (more advanced) receiver is selected as the master, the playout point of this receiver is selected as the IDMS reference. This will result to poor presentation quality at slower (or more lagged) receivers. On the contrary, if the slowest receiver is selected as master, this will result in high packet drops at faster slave receiver(s). It is noteworthy that synchronization can also be based on the mean playout point (i.e., the IDMS reference is calculated by averaging the playout timing reported from all the distributed receivers). A problem with the M/S technique is that the master can act as a bottleneck in the system. A second problem deals with the associated degree of unfairness with the slave receivers. Boronat et al. [21] discussed possible options with pros and cons for the master selection in this scheme.

B. Synchronization Maestro Scheme (SMS)

In SMS scheme [22], all the receivers are handled fairly as master and slaves do not exist. SMS involves a *Synchronization Manager* (SM) which can be performed by one of the source or receiver. For example, in Fig. 4, one receiver (destination) performs the role of SM.

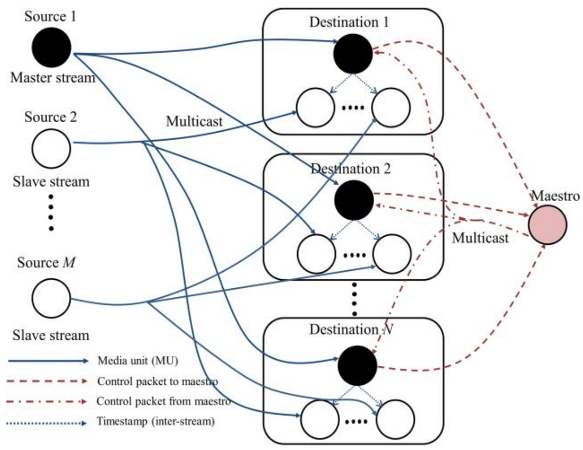

Fig. 4. Synchronization Maestro Scheme [19].

Each receiver estimates the network delay and uses the estimates to determine the local presentation time of the MU. Then, each receiver sends this estimated presentation time of MU to the SM. After that, the SM gathers the estimates from the receivers and adjusts the presentation timing among the receivers by multicasting control packets to receivers. The SMS scheme assumes that the clock speed at the sources and receivers is the same and that the current local times are also the same (i.e., globally synchronized clocks). Figure 4 depicts the basic principle of the SMS technique. Boronat et al. [16] presented the RTCP-based schemes which follow the same basic principle. The SMS scheme (like the M/S) is a centralized solution, and thus it can confront the bottleneck problem. The advantage of the SMS scheme over M/S is its fairness to the receivers because the feedback information of all the receivers is accounted for determining the presentation time of the MU. However, this fairness costs more communication overhead among the receiver and the Synchronization Manager (SM).

C. Distributed Control Scheme (DCS)

Figure 5 illustrates the DCS scheme [23]. Each receiver estimates the network delay, and then determines the presentation time of the MU. Then, it sends (multicasts) this presentation time to all the receivers. After that, every receiver will have the entire view of the estimated time of MU. Each receiver has the flexibility to decide the reference playout time among the timing of all the receivers. The DCS scheme provides higher flexibility to each receiver to decide the

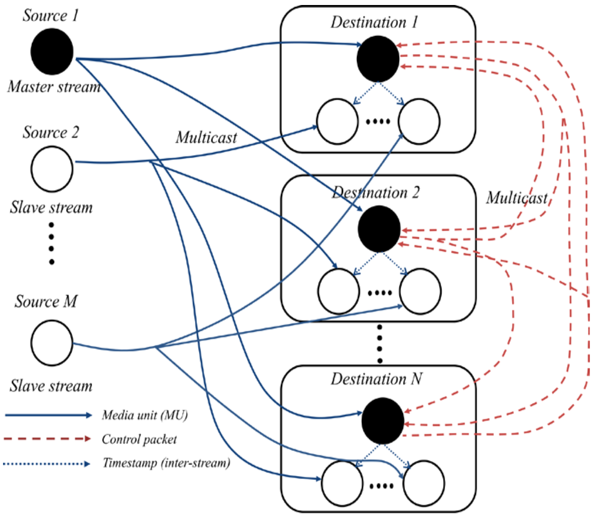

Fig. 5. The Distributed Control Scheme [19]

presentation time of MU. For example, it is possible that by selecting the presentation time of other receiver, it can achieve higher group synchronization quality, but it may cause the inter-media or intra-media synchronization degradation. In this case, the receiver has the flexibility to choose between the types of synchronization depending upon the nature of application on hand. If the application on hand requires the higher inter-media or intra-media synchronization and can sacrifice on the group synchronization to certain limit, then the receiver can select its own determined presentation time and vice versa. DCS is a distributed scheme by nature and does not suffer from the bottleneck problem. If one or more receivers leave the system, it will not disturb the overall scheme. This greater flexibility and the distributed nature of DCS make it complex in terms of processing. This happens because the receiver does more calculations and comparisons before deciding the presentation time of MU. Finally, DCS has a higher message complexity, because each receiver multicasts the estimated presentation time.

D. Comparison of Control Schemes - Lessons Learned

The following factors affect IDMS performance (i.e., the level of synchronicity among receivers). These factors can be used as evaluation criteria for the comparison of IDMS control schemes [3][16]:

Robustness: Disconnections and failures of some receivers/participants may affect the ability to perform the IDMS control. In a distributed control architecture (DCS), the failure of any of the participant has a slight effect on the other participants because each one of them is independent and has locally all the required information to compute the overall synchronization status at any time. In SMS, if the Maestro cannot communicate with the other terminals owing to some trouble, no destination can carry out the IDMS control. Generally, a centralized scheme (SMS or M/S) is *less robust* than distributed schemes. A distributed architecture (DCS) is more robust because it can simplify the deployment and maintenance of a distributed multimedia application.

Scalability: This is the ability to handle multiple concurrent participants/receivers in an IDMS session. SMS requires the maintenance of a dedicated server (Maestro) to which all the control information converges. Thus, SMS may present higher scalability constraints. For example, multiple receivers may send control packets almost simultaneously, thus originating a feedback-implosion problem because of the IDMS control. As the number of the receivers/participants increases, bursty traffic due to control packets can overwhelm the synchronization manager and may degrade the output quality of the media streams.

Traffic overhead: It is generated by two factors: (1) the distribution of the playout timing messages from the participants to the synchronization manager; and (2) the transmission of playout setting instructions. Generally, traffic overhead may be higher in DCS than in SMS.

Interactivity (*low delays*): Each slave destination can compute the detected playout asynchrony when it receives the control messages from the master destination. Consequently, the lowest delays may be achieved using the M/S scheme. In DCS, each participant must gather the overall status from all the other active participants. As a result, delays in DCS are bit larger. In SMS, the Maestro must gather the playout timing of all the receivers, and then send back to them new control messages including IDMS setting instructions. Therefore, the highest delay (smallest interactivity) occurs in SMS, but this delay depends on the network topology and on the routing tree structure.

Location of control nodes: The location of the multimedia source and the location of the synchronization manager affect the IDMS performance of the schemes. Centralized control schemes are more sensitive to these locations. Under heavily loaded network conditions, the IDMS performance with SMS can be slightly larger than the one with M/S and DCS schemes, if the media source is selected as the Maestro. This is due to the fact that IDMS control packets sent by the Maestro are sent through the same path as the MUs (e.g., video frames, encapsulated in data packets). In SMS scheme, IDMS control messages scarcely increase the network load. But, if the bandwidth availability is limited, some (data or control) packets may be dropped. If a control packet is dropped (lost), the destination cannot get the reference output timing until receiving the next control packet. On the other hand, in M/S scheme, if the most heavily loaded destination is selected as the master, the data packets are less likely dropped on the intermediate links because it does not need to receive control packets and their own sent control packets may be transmitted in the opposite direction to the media data packets.

Consistency: In media-sharing applications, consistency is required to guarantee concurrently synchronized playout states in all the distributed participants. In centralized schemes, inconsistency between receivers' states occurs less likely, since all of them always receive the same control information about IDMS timing from the Maestro (in SMS) or the Master receiver (in M/S scheme). On the contrary, in a DCS scheme, there is no guarantee that the same reference IDMS timing, from among all the collected IDMS control reports, will be selected in all the distributed receivers since each one takes its own decisions locally. This leads to a more probable potential inter-receivers inconsistency.

Security: Centralized architectures provide higher security than distributed architectures. In DCS architectures, we have lack of control because each participant has the responsibility of what is doing, and some participants may be malicious. Synchronization entities (Maestro in SMS, or each destination in DCS and in M/S) must consider inconsistent playout information (exceeding configuration limits) as a malfunctioning service and reject that information in the calculation of the necessary playout adjustments (synchronization actions).

Coherence: This is the ability to synchronously and simultaneously coordinate the media playout timing according to a reference timing for IDMS. For this reason, the maximum playout asynchrony (between the most lagged and the most advanced receiver) must be estimated. And this is easy in DCS and SMS schemes. But, in M/S scheme, each receiver can only know the asynchrony between its local playout process and that of the Master. Using M/S scheme, the reactive synchronization actions will not be performed simultaneously because slave receivers adjust their playout timing when they detect an asynchrony value (regarding the playout state of the master) exceeding an allowable threshold and this situation may not be detected at the same time in all the slave receivers. Consequently, SMS outperforms the M/S and DCS in terms of coherence.

Fairness: M/S scheme is appropriate for applications in which a single receiver has a certain priority level over the others. For example, in multi-point video conferencing (e.g., synchronous e-learning), the teacher's terminal can be selected as the Master receiver, which directs to the students' devices the required playout adjustments to get in sync. However, M/S scheme cannot treat all the receivers fairly. This problem is minimized when SMS or DCS are employed because the reference output timing is selected after a comparison among the output timing of all the receivers.

Flexibility: Using M/S scheme, there is no option for selecting the reference output timing since it is taken from the one reported by the master destination. Conversely, the Maestro, in SMS, and the distributed receivers, in DCS, can employ several dynamic policies for selecting an IDMS reference from the collected output timings.

Conclusively, M/S scheme can provide the best performance in terms of scalability, traffic overhead, and interactivity. Moreover, M/S scheme can be proper in those scenarios in which the bandwidth availability is limited, and also in those use cases in which a single participant (e.g., a teacher in a synchronous e-learning scenario) has a certain priority level over the others. However, the M/S scheme presents serious drawbacks, if some features such as robustness, coherence, flexibility, and fairness are required. Finally, M/S and SMS control schemes are the most appropriate in terms of consistency. Centralized schemes (M/S and SMS) have larger network delays (low interactivity), lower robustness with poorer flexibility and scalability.

E. Classification of Group Synchronization Solutions

In Table II, we summarize the most well-known synchronization solutions by presenting the above schemes and other features of interest such as the following ones:

- *Group synchronization schemes*: The control schemes (M/S receiver scheme, SMS, and/or DCS) included in the solutions are indicated.
- *Synchronization information*: The information used for synchronization (included in the transmitted MUs) is indicated.
- *Location of the synchronization techniques*: The synchronization control is made by the source(s) or by the receiver(s) or both.
- *Synchronization techniques*: The most representative techniques included in each solution have been indicated in Table II.

In the first column (Table II), the *Name* of the group synchronization solution and the corresponding cited work are included. Several solutions use RTP/RTCP protocols [32]. Particularly, they use feedback and time information (timestamps) included in the RTP/RTCP. These solutions exploit the use of control RTCP report packets for including feedback information for multimedia synchronization purposes. The VTR media synchronization algorithm [18] has been used in media synchronization between voice and movement of avatars in networked virtual environments. The synchronization maestro scheme (SMS) for group synchronization, employed together with the VTR media synchronization algorithm, has been enhanced so that the SMS scheme can be used efficiently in a networked real-time game with collaborative work [31], and in a P2P-based system [28].

TABLE II:
CLASSIFICATION OF SOME GROUP SYNCHRONIZATION SOLUTIONS

Name	Scheme	Sync information	Location	Synchronization techniques
VTR [18]	Master/Slave receiver scheme	Timestamps Seq. number	Source and Receiver	Change of the buffering time according to the delay estimation. Decreasing the number of media streams. Preventive pauses. Reactive skips and pauses. Skips at the source side. Playout duration extensions or reductions. Virtual local time expansions or contractions.
[6]	DCS	Timestamp in 1st packet	--	Initial transmission and playout instant. Playout rate adjustments (receiver's clock). Master/slave receiver switching (chairman).
BS [24]	DCS	Timestamps Seq. number	Receiver	Skips (discarding) and pauses (duplicates). Late events are dropped.
LL-TW [23]	DCS	Timestamps	Receiver	Event-based synchronization control. Playout duration extension. Rollback-based techniques.
DCS [25]	DCS	Timestamps	Receiver	VTR techniques
TSS [26]	DCS	Timestamps	Receiver	Event-based synchronization control. Playout duration extension. Rollback-based techniques.
ILA [27]	DCS	Timestamps	Receiver	Event-based synchronization control. Preventive MDU/event discarding. Reactive events discarding.
ESMS	SMS	Timestamps	Source	Skipping and VTR

[28]		Seq. number	and Recei-ver	techniques for intra- stream synchronization.
RTP-FGP [29]	SMS	Timestamps Source id.	Source and Recei-ver	Initial playout instant. Reactive skips and pauses at the receiver side. Playout rate adjustment. Virtual time expansion. Master/slave receiver switching (group synchronization).
SMS [11], [18], [30], [31], [22]	SMS	Timestamps Seq. number	Recei-ver	Initial transmission instant (only in [22])

BS: bucket synchronization; DSC: distributed control scheme;
ILA: interactivity-loss avoidance;
LL-TW: local-lag and time warp algorithms;
RTP-FGP: RTP-based feedback-global protocol;
SMS: synchronization maestro scheme; TSS: trailing state synchronization;
VTR: virtual time rendering algorithm

Moreover, many different reactive techniques have been proposed. For example, receivers can discard late events in [24]. In addition, receivers in [25] can use rollback techniques, such as maintaining late events and using them to compensate for inconsistency at the receiving end. In the *Timewarp* algorithm [23], this can cause an extra overhead in terms of memory space and computation for inconsistency compensation. To re-establish the consistency of the game state, rollback-based techniques were developed in [23]. Copies of the states are maintained after command executions and events received after their playout time are stored locally instead of being dropped and used to compensate for the inconsistency among receivers' views. Then, visual rendering of significant events can be delayed (to avoid inconsistencies if corrections occur). In this case, the difficulty is that the use of these realignment techniques may further impact on the responsiveness of the system. The *Trailing State Synchronization* (TSS) algorithm [26] uses dynamically changing states as the source of rollbacks as opposed to static snapshots, which is the fundamental difference between it and *Timewarp* [23]. TSS preserves more than a few instances of the applications running with different synchronization delays. In TSS, inconsistencies are noticed by detecting when the leading state and the correct state diverge, and at that point are corrected. From another perspective, a proactive event discarding mechanism is used in [27]. This mechanism is based on the discrimination of obsolete events. In particular, obsolete events are discarded with a probability depending on the level of interactivity.

In the next section, we present IDMS standardization efforts and some state-of-the-art IDMS solutions.

IV. Recommendations and Solutions

A. Standardization efforts

ETSI (European Telecommunications Standards Institute) TISPAN (Telecoms & Internet converged Services & Protocols for Advanced Networking) has been carried out standardization efforts of IDMS. This standardization is also a highlight for the IETF AVTCORE WG (Internet Engineering Task Force - Audio/Video Transport Core Maintenance Working Group). The specification [33] does pose IDMS and the synchronization of media streams from different sources as a requirement for providing synchronization-sensitive interactive services. These use cases are mostly in the categories of 'low' or 'medium' synchronization, and not very high requirements are posed to delay differences between various user equipments. However, Montagud et al. [3] presented up to 19 use cases for IDMS, each one having its own (very high) synchronization requirements. The most of these use cases are not supported by the protocol specification, which gives a delay difference of between 150 and 400 ms as a guideline for achieving transparent interactivity, based on ITU guidelines for interactivity in person-to-person communication.

ETSI TISPAN has done the first work on standardizing RTCP usage for IDMS. The ETSI proposal is a dedicated solution for use in large scale IPTV deployments with 'low' to 'medium' level synchronization requirements. The ETSI solution [34] is an evolved version of an RTCP-based IDMS approach including an AMP scheme that adjusts the playout timing of each one of the geographically distributed consumers in a specific cluster if an allowable asynchrony threshold between their playout states is exceeded. Still, there are use cases [3] that require higher levels of synchronization and are not supported efficiently by the ETSI solution.

Within the Internet Engineering Task Force (IETF), the AVTCORE working group [35] carries out standardization of the RTCP-based IDMS protocol. This is the core group that is responsible for the RTP and accompanying RTCP protocol. Actually, most RTCP extensions are developed within the IETF. van Deventer et al. [36] provided an overview of recently published standards for media synchronization from the most relevant bodies: IETF, ETSI, MPEG, DVB, HbbTV, and W3C.

B. Solutions

Boronat et al. [16] described most IDMS solutions that define new proprietary protocols with specific control messages which increase the network load. Montagud et al. [37] reviewed the existing sync reference models by examining the involved features, components, and layers in each one of them. Their study reflects the need for a new modular and extensible theoretical framework to efficiently comprehend the overall media sync research area. From another perspective, Huang et al. [38] presented a historical view of temporal synchronization studies focusing on continuous multimedia. They demonstrated how the development of multimedia systems has created new challenges for synchronization technologies. They concluded with a new application dependent, multi-location, multi-requirement synchronization framework to address these new challenges.

The realization of synchronous shared experiences requires that users feel that they are coherently communicating with each other. Vaishnavi et al. [1] analyzed challenges that need to be tackled to achieve coherence: QoS, mobility, and distributed media synchronization. They presented their solution to distributed media synchronization. Their design uses the local lag mechanism over a distributed control or master–slave signaling architecture. Montagud et al. [39]

presented an IDMS solution based on extending the capabilities of RTP/RTCP protocols. To enable an adaptive, highly accurate, and standard compliant IDMS solution, they specified RTCP extensions in combination with several control algorithms and adjustment techniques.

Focused on the TV area, Costa and Santos [40] surveyed the existing media sync solutions, classifying them in terms of types of involved devices, types of media content, types of sync techniques, targeted applications or scenarios, and evaluation methodologies. The following sync specific aspects were considered to classify the existing solutions: protocols, algorithms, delivery channels, specification methods, architectural schemes, allowable asynchrony levels, and evaluation metrics. Marfil et al. [41] presented an adaptive, accurate and standard-compliant IDMS solution for hybrid broadcast and broadband delivery. Their solution can accomplish synchronization when different formats/versions of the same (or even related) contents are being played out in a shared session. It can also independently manage the playout processes of different groups of users. Their IDMS solution has been integrated within an end-to-end platform, which is compatible with the *Hybrid broadcast broadband TV* (HbbTV) standard. It has been applied to digital video broadcasting-terrestrial technology and tested for a social TV scenario, by also including an ad-hoc chat tool as an interaction channel.

Ishibashi et al. [42] carried out QoE assessment of fairness between players in a networked game with olfaction. They investigated the influence of the time it takes for a smell to reach a player on fairness. They illustrated that fairness is hardly damaged when the constant delays are smaller than about 500 ms. The used media synchronization algorithm considers the human perception of intra-stream and inter-stream synchronization errors. Ghinea and Ademoye [43] conducted a perceptual measurement of the impact of a synchronization error between smell sensory data and audiovisual content, assuming the audiovisual lip skew is zero. Their results showed a synchronization threshold of 30 s, when olfaction is ahead of audiovisual data, and of 20 s when olfaction is behind. In joint musical performance, multiple users play their respective same or different types of musical instruments together. However, the media synchronization quality and interactivity may seriously be deteriorated owing to the network delay. Sithu and Ishibashi [19] proposed a new media synchronization control called the '*dynamic local lag control*'. By QoE assessment, they demonstrated that this new control can achieve a high quality of media synchronization and keep the interactivity high in joint musical performance.

Bello et al. [44] presented a distributed multimedia synchronization protocol oriented to satisfy logical and temporal dependencies in the exchange of real-time data in mobile distributed systems by using logical mapping, avoiding the use of global references. Two main aspects of their protocol include: (1) the computation of the deadline for messages by using only relative time points, and (2) by dividing the processing stage to achieve synchronization with an asymmetric principle of design. Simulations results showed that their protocol is effective in diminishing the synchronization error. Furthermore, their protocol is efficient as regards processing and storage costs at the mobile hosts, and in the overhead attached per message with a reduced usage of bandwidth across the wired and wireless channels in comparison with the RTP.

Internet-based video services can also benefit from IDMS. We can achieve a smooth multiple-stream distributed multimedia presentation over the Internet if we apply presentation adaptation and flow control. Huang et al. [45] proposed the *Pause-And-Run* approach for *k*-stream (PARK) multimedia presentations over the Internet to achieve reliable transmission of continuous media. They evaluated the application of the PARK approach over the Internet. The evaluation results revealed a suitable buffering control policy for the audio and video media respectively. The characteristics of the PARK approach are:

- PARK adopts TCP to achieve reliable transmission for continuous media.
- A novel flow adaptation scheme reduces the overhead of the network and end-hosts because the slow-start scheme is embedded in TCP. The server adapts its transmission rates to the buffer situation of the client and prevents the client's buffers from overflow and underflow as much as possible.
- With the provision of multiple-stream synchronization and the multi-level adaptation control, the client achieves smooth multimedia presentations and graceful presentation degradation.

From another perspective, *Wersync* [46] is a novel web-based platform that enables distributed media synchronization and social interaction across remote users. By using Wersync, users can create or join on-going sessions for concurrently consuming the same media content with other remote users in a synchronized manner.

Rainer et al. [47] presented *Merge and Forward*, an IDMS scheme for adaptive HTTP streaming as a distributed control scheme and adopting the MPEG-DASH standard [48] as a representation format. They introduced so-called IDMS sessions and described how an unstructured peer-to-peer overlay can be created using the session information and using the MPEG-DASH. They assessed the performance of *Merge and Forward* with respect to convergence time (time needed until all clients hold the same reference time stamp) and scalability. After the negotiation on a reference time stamp, the clients have to synchronize their multimedia playback to the agreed reference time stamp. In order to achieve this, the authors proposed a new AMP approach minimizing the impact of playback synchronization on the QoE. The proposed AMP was assessed subjectively using crowdsourcing.

Kwon et al. [49] proposed a media sharing scheme (named *PlaySharing*) for scalable media streaming and precise group synchronization services. PlaySharing combines event-based synchronization, local adjustment of playback position errors, and rare and periodic synchronization. It achieves sustained precise synchronization by minimizing synchronization control packets during network congestion. Event-based synchronization manages the synchronization between a media source device and client devices using event messages from the source device. To reduce the number of synchronization control packets, the local adjustment of playback position errors corrects those on the media client devices according to the expected playback position for the time elapsed since the last synchronization. Rare and periodic synchronization is applied in preparation for control packet loss, correcting the local adjustment errors when no event occurs. To evaluate the performance of PlaySharing, the average synchronization errors (between the source device and the client devices) were measured in an IEEE 802.11 infrastructure network configuration and a hierarchical wireless media streaming network (HSN) configuration. For these measurements, two protocols were used: the user datagram protocol unicast and broadcast for control packet transmission. The experimental results showed that PlaySharing, with user datagram protocol unicast transmission of control messages in the HSN, has the lowest synchronization errors in the experiments.

Last but not least, an additional challenge in IDMS is securing group communication that involves *Multicast Group Key Management*. Such management is the management of the keys in a group communication. Developing group key management faces additional challenges in wireless mobile networks (e.g., MANETs) due to their inherent complexities. The constraints of wireless devices in terms of resources scarcity and the mobility of group members increase the complexity of designing a group key management scheme. Daghighi et al. [50] surveyed existing group key management schemes that consider the host mobility issue in secure group communications in wireless mobile environments.

V. CONCLUSION AND FUTURE WORK

This paper has illustrated various issues on multimedia synchronization. It has presented the basic control schemes for IDMS and has focused on IDMS solutions and standardization efforts for emerging distributed multimedia applications.

Lessons Learned

IDMS is essential in various emerging distributed multimedia applications such as social TV, hybrid broadcast/broadband services, networked quiz shows, networked video wall, multi-party multimedia conferencing, and interactive 3D tele-immersive applications. 3D tele-immersive applications provide geographically distributed users with a realistic and immersive multimedia experience [51]. The protocol software developer must take into account: (1) the context and space in which the IDMS solution is going to be deployed; and (2) the multimedia application requirements that must be satisfied. The key-point in IDMS is to minimize the delay differences among different receivers by introducing proper buffering mechanisms. The primary latency in IDMS scenario is the playout delay that consists of the sending buffer delay, packet transfer delay, and receiving buffer delay. The transfer delay (which includes packet transmission and path propagation delay) of the same (media) video packet to different destinations often differs significantly because of the variations in available bandwidth and path propagation delays. These packet transfer delay differences are the main barrier for IDMS because they affect the receivers' synchronous playout possibility substantially.

Existing control schemes (i.e., M/S, SMS, DCS) for IDMS have their own strengths and weaknesses. However, the choice between these schemes is largely application-dependent. For their evaluation, certain metrics must be used such as robustness, fairness, scalability, traffic overhead, interactivity (low delay), location of control nodes, consistency, coherence, security, and flexibility.

Precise group synchronization schemes can be deployed by using *event-based synchronization*. This kind of synchronization implies that the synchronization controller can transfer a synchronization control message to the media client devices when an event (e.g., Play, Pause, Resume, Stop, and Seek) in a media source device occurs. The control message may include an event time, an event type, a playback position, etc. Then, media client devices could synchronize their playback states with the media source device after correcting errors, based on the received control message.

The current industry pushes for new IDMS services, both at the IP media stream level (IETF RTCP, ETSI TISPAN) and the MPEG-2 transport stream level (DVB CSS, MPEG TEMI). It also includes more fundamental standards [(W3C SMIL and ITU - NCL (Nested Context Language)] that can serve as models for future and more general synchronization primitives. The standardization of IDMS will facilitate the uptake of implementations and of the interoperability between different implementations. Such standardization will ensure a more extensive use of IDMS.

Future Work

- The basic control schemes for IDMS must be compared and evaluated under various types of wireless networks (e.g., MANETs, VANETs). The evaluation metrics must cover many aspects such as robustness, interactivity, etc.
- Future IDMS techniques could benefit from cross-layer optimization. Such optimization allows communication between OSI-RM layers by permitting one layer to access the data of another layer to exchange information and enable interaction [52]. It contributes to an improvement of QoS under various operational conditions. The cross-layer control mechanism can provide feedback on concurrent quality information for the adaptive setting of control parameters of a multimedia system. As a result, it could help to the utilization of synchronization techniques such as preventive control. A comprehensive multimedia synchronization subsystem will integrate preventive and reactive methods and will use a cross-layer optimization method and other components (e.g., the IP Multimedia Subsystem).
- In RTP-based multimedia streaming services, client-driven media synchronization mechanisms must be developed to provide accurate media synchronization such as to reduce: (1) the initial synchronization delay; (2) the processing complexity at the client device; (3) the number of required user datagram protocol ports; and (4) the amount of control traffic injected into the network. Such a synchronization mechanism was recently proposed in [53]. In this mechanism, the server does not need to send any RTCP sender report packets for synchronization.

Instead, the client device derives the precise normal play time for each video and audio stream from the received RTP packets containing an RTP timestamp.

- Intelligent distributed control schemes are required to develop IDMS for pull-based streaming. Such schemes must negotiate a reference playback timestamp among the peers participating in an IDMS session. The MPEG-DASH standard can be used to incorporate these IDMS sessions in the Media Presentation Description (MPD). In this way, the proposed solutions will remain compliant to the MPEG-DASH because non-IDMS peers will ignore the additional session description when parsing the MPD.
- Finally, we must introduce and evaluate transmission schemes that will minimize the transmission loss rate, while still ensuring the synchronous arrival of video packets. The main principle of their design will be to leverage the packet transfer delay differences among different destinations for spreading the departures of video/audio packets. The integration of such transmission schemes with dynamic AMP solutions will be a challenge.

REFERENCES

[1] Vaishnavi, I., Cesar, P., Bulterman, D., Friedrich, O., Gunkel, S., and Geerts, D. "From IPTV to synchronous shared experiences challenges in design: Distributed media synchronization," *Signal Processing: Image Communication,* vol. 26, no. 7, pp.370-377, 2011.

[2] Kernchen, R., Meissner, S., Moessner, K., Cesar, P., Vaishnavi, I., Boussard, M., and Hesselman, C. "Intelligent multimedia presentation in ubiquitous multidevice scenarios," *IEEE Multimedia*, vol. 17, no. 2, pp. 56-63, 2010.

[3] Montagud, M., Boronat, F., Stokking, H., and van Brandenburg, R. "Inter-destination multimedia synchronization: Schemes, use cases and standardization," *Multimedia Systems,* vol. 18, no. 6, pp. 59-482, 2012.

[4] Li, M., Sun, Y., and Sheng, H. "Temporal relations in multimedia systems," *Computers & Graphics*, vol. 21, no. 3, pp. 315-320, 1997.

[5] Blakowski, G., and Steinmetz, R. "A media synchronization survey: Reference model, specification, and case studies," *IEEE J. Sel. Areas Commun.,* vol. 14, no. 1, pp. 5-35, 1996.

[6] Akyildiz, I.F., and Yen, W. "Multimedia group synchronization protocols for integrated services networks," *IEEE J. Sel. Areas Commun.,* vol. 14, no. 1, pp. 162-173, 1996.

[7] Park, S., and Choi, Y. "Real-time multimedia synchronization based on delay offset and playout rate adjustment," *Real-Time Imaging*, vol. 2, no. 3, pp.163-170, 1996.

[8] Park, S., and Kim, J. "An adaptive media playout for intra-media synchronization of networked-video applications," *Journal of Visual Commun. and Image Representation*, vol. 19, no. 2, pp.106-120, 2008.

[9] Laoutaris, N., and Stavrakakis, I. "Intrastream synchronization for continuous media streams: A survey of playout schedulers," *IEEE Network*, vol. 16, no. 3, pp. 30-40, 2002.

[10] Aggarwal, S. and Jindal, A. "Comprehensive overview of various lip synchronization techniques," *Int. Symposium on Biometrics and Security Technologies*, ISBAST 2008, pp.1-6, 23-24 April 2008.

[11] Ishibashi, Y., Tasaka, S., and Miyamoto, H. "Joint synchronization between stored media with interactive control and live media in multicast communications," *IEICE Trans. on Communications*, vol. E85-B, no. 4, pp.812–822, 2002.

[12] Wongwirat, O., and Ohara, S. "Haptic media synchronization for remote surgery through simulation," *IEEE MultiMedia*, vol. 13, no. 3, pp.62–69, 2006.

[13] Hikichi, K., Morino, H., Arimoto, I., Sezaki, K., and Yasuda, Y. "The evaluation of delay jitter for haptic collaboration over the Internet," In *Proc. of IEEE Global Commun. Conference (GLOBECOM)* (pp. 1492-1496), 2002.

[14] Ishibashi, Y., Tasaka, S., and Hasegawa, T. "The Virtual-Time Rendering algorithm for haptic media synchronization in networked virtual environments," In *Proc. of the 16th Int. Workshop on Communications Quality and Reliability* (pp. 213-217), 2002.

[15] Isomura, E., Tasaka, S., and Nunome, T. "QoE enhancement in audiovisual and haptic interactive IP communications by media adaptive intra-stream synchronization," In *Proc. of IEEE TENCON* (pp. 1085-1089), 2011.

[16] Boronat, F., Lloret, J., and García, M. "Multimedia group and inter-stream synchronization techniques: A comparative study," *Information Systems*, vol. 34, no. 1, pp. 108-131, 2009.

[17] Din, S., and Bulterman, D. "Synchronization techniques in distributed multimedia presentation," *MMEDIA 2012: The Fourth Int. Conferences on Advances in Multimedia* (pp.1-9), 2012.

[18] Ishibashi, Y., Tsuji, A., and Tasaka, S. "A group synchronization mechanism for stored media in multicast communications," In *Proc. of the Sixth Annual Joint Conference of the IEEE Computer and Communications Societies* (INFOCOM), (vol. 2, pp. 692–700). Kobe, Japan: IEEE Press, 1997.

[19] Sithu, M., and Ishibashi, Y. "Media synchronization control in multimedia communication," In D. Kanellopoulos (Ed.), *Emerging Research on Networked Multimedia Communication Systems* (pp.25-61). Hershey, PA: Information Science Publishing, 2015.

[20] Boronat, F., Guerri, J.C., and Lloret, J. "An RTP/RTCP based approach for multimedia group and inter-stream synchronization," *Multimedia Tools Appl J*, vol. 40 no. 2, pp. 285-319, 2008.

[21] Boronat, F., Montagud, M., and Vidal, V. "Master selection policies for inter-destination multimedia synchronization in distributed applications," In *Proc. of the IEEE 19th Int. Symposium on MASCOTS*, (pp. 269-277), 2011.

[22] Ishibashi, Y., and Tasaka, S. "A group synchronization mechanism for live media in multicast communications," In *Proc. of the IEEE GLOBECOM' 97*, November 1997, (pp.746–752), 1997.

[23] Mauve, M., Vogel, J., Hilt, V., and Effelsberg, W. "Local-lag and timewarp: Providing consistency for replicated continuous app," *IEEE Trans. Multimedia*, vol. 6, no. 1, pp. 47-57, 2004.

[24] Diot, C., and Gautier, L. "A distributed architecture for multiplayer interactive applications on the Internet", *IEEE Network*, vol. 13, no. 4, pp. 6-15, 1999.

[25] Ishibashi, Y., and Tasaka, S. "A distributed control scheme for causality and media synchronization in networked multimedia games", in *Proc. of the 11th Int. Conference on Computer Communications and Networks*, Miami, USA, October 2002, pp. 144–149.

[26] Cronin, E. Filstrup, B. Jamin, S. and Kurc, A.R. "An efficient synchronization mechanism for mirrored game architectures", *Multimedia Tools Appl.*, vol. 23 no. l, pp. 7-30, 2004.

[27] Palazzi, C.E., Ferretti, S., Cacciaguerra, S., and Roccetti, M. "On maintaining interactivity in event delivery synchronization for mirrored game architectures", in *IEEE Global Telecommunications Conference Workshops*, Dallas, TX, USA, November/December 2004, pp. 157–165.

[28] Hashimoto, T. and Ishibashi, Y. "Group synchronization control over haptic media in a networked real-time game with collaborative work", in *Proc. of the Fifth ACM SIGCOMM workshop on Network and System Support for Games,* Singapore, October 2006.

[29] Boronat, F., Guerri, J.C., Esteve, M., & Murillo, J.M. "RTP-based feedback global protocol integration in Mbone tools", EUROMEDIA 2002, Modena, Italy (Best Paper Award) April 2002.

[30] Kaneoka, H., and Ishibashi, Y. "Effects of group synchronization control over haptic media in collaborative work", in: *Proc. of the 14th Int. Conference on Artificial Reality and Telexistence* (ICAT'04), Coex, Korea, November/December 2004, pp. 138–145.

[31] Kurokawa, Y., Ishibashi, Y., and Asano, T. "Group synchronization control in a remote haptic drawing system", in: *Proc. of the IEEE International Conference on Multimedia and Expo*, Beijing, China, July 2007, pp. 572–575.

[32] Schulzrinne, H., Casner, S., Frederick, R., and Jacobson, V. "RTP: a transport protocol for real-time applications", RFC–3550, July 2003.

[33] ETSI TS 181 016 V3.3.1 (2009-07). *Telecommunications and Internet converged Services and Protocols for Advanced Networking (TISPAN); Service Layer Requirements to integrate NGN Services and IPTV*.

[34] Montagud, M., and Boronat, F. "Enhanced adaptive RTCP-based inter-destination multimedia synchronization approach for distributed applications," *Computer Networks*, vol. 56, no. 12, pp. 2912-2933, 2012.

[35] van Brandenburg, R.H.O.F.M.K., Stokking, H., van Deventer, O., Boronat, F., Montagud, M., and Gross, K. "Inter-Destination Media Synchronization (IDMS) using the RTP Control Protocol (RTCP)", RFC–7272, 2014.

[36] van Deventer, M. O., Stokking, H., Hammond, M., Le Feuvre, J., and Cesar, P. "Standards for multi-stream and multi-device media synchronization," *IEEE Communications Magazine*, vol. 54, no. 3, pp.16-21, 2016.
[37] Montagud, M., Jansen, J., Cesar, P., and Boronat, F. "Review of media sync reference models: Advances and open issues," *MediaSync2015 Workshop*, ISBN: 978-90-5968-463-4, 2015.
[38] Huang, Z., Nahrstedt, K., and Steinmetz, R. "Evolution of temporal multimedia synchronization principles: A historical viewpoint," *ACM Trans. on Multimedia Computing, Commun., and Applications*, vol. 9, no.1. Article 34, 23 pages, 2013.
[39] Montagud, M., Boronat, F., Stokking, H., and Cesal, P. "Design, development and assessment of control schemes for IDMS in a standardized RTCP-based solution," *Computer Networks*, vol. 70, no. 9, pp.240-259, 2014.
[40] Costa, R., and Santos, C.A.S. "Systematic review of multiple contents synchronization in interactive television scenario," *ISRN Commun. and Networking*, Volume 2014, Article ID 127142, pp. 1-17, 2014.
[41] Marfil, D., Boronat, F., Montagud, M., and Sapena, A. "IDMS solution for hybrid broadcast broadband delivery within the context of HbbTV standard". *IEEE Transactions on Broadcasting*, 2018.
[42] Ishibashi, Y., Hoshino, S., Zeng, Q., Fukushima, N., and Sugawara, S. "QoE assessment of fairness in networked game with olfaction: Influence of time it takes for smell to reach player," *Multimedia Systems*, vol. 20, no. 5, pp. 621-631, 2014.
[43] Ghinea, G., and Ademoye, O. A. "Perceived synchronization of olfactory multimedia," *IEEE Trans. on Systems, Man and Cybernetics*, vol. 40, no. 4, pp. 657-663, 2010.
[44] Bello, M. A. O., Dominguez, E. L., Hernandez, S. E. P., and Cruz, J. R. "Synchronization protocol for real time multimedia in mobile distributed systems," *IEEE Access*, vol. 6, pp.15926-15940, 2018.
[45] Huang, C.-M., Kung, H.-Y., and Yang, J.-L. "Synchronization and flow adaptation schemes for reliable multiple-stream transmission in multimedia presentations," *Journal of Systems and Software*, vol. 56, no. 2, pp. 133-151, 2001.
[46] Belda, J., Montagud, M., Boronat, F., Martinez, M., and Pastor, J. "Wersync: A web-based platform for distributed media synchronization and social interaction," In *Proc. ACM Int. Conf. on Interactive Experiences for Television and online Video (TVX 2015)*, 2015.
[47] Rainer, B., Petscharnig, S., and Timmerer, C. "Merge and Forward: A self-organized inter-destination media synchronization scheme for adaptive media streaming over HTTP," In *MediaSyn*c (pp. 593-627). Springer, Cham., 2018.
[48] Sodagar. I. "The MPEG-DASH standard for multimedia streaming over the Internet," *IEEE MultiMedia*, vol. 18 no.4, pp.62-67, 2011.
[49] Kwon, D., Kim, H., and Ju, H. (2018). "PlaySharing: A group synchronization scheme for media streaming services in hierarchical WLANs," *Int. J. of Netw. Management*, vol. 28 no. 6, e2024, 2018.
[50] Daghighi, B., Kiah, M.L.M., Shamshirband, S., Iqbal, S., and Asghari, P. "Key management paradigm for mobile secure group communications: Issues, solutions, and challenges," *Computer Communications*, vol. 72, pp. 1-16, 2015.
[51] Huang, Z., Wu, W., Nahrstedt, K., Rivas, R., and Arefin, A. "SyncCast: synchronized dissemination in multi-site interactive 3D tele-immersion". In *Proc. of the second annual ACM conference on Multimedia systems* (pp. 69-80). ACM. 2011.
[52] Bin-Salem, A., and Wan, T. "Survey of cross-layer designs for video transmission over wireless networks," *IETE Technical Review*, vol. 29, no. 3, pp. 229-247, 2012.
[53] Jung, T., and Seo, K. "A client-driven media synchronization mechanism for RTP packet-based video streaming," *Journal of Real-Time Image Processing*, vol. 12 no. 2, pp. 455-464, 2016.

Dimitris N. Kanellopoulos is a member of the Educational Software Development Laboratory in the Department of Mathematics at the University of Patras, Greece. He received a Diploma in Electrical Engineering and a Ph.D. in Electrical and Computer Engineering from the University of Patras. Since 1990, he was a research assistant in the Department of Electrical and Computer Engineering at the University of Patras and involved in several EU R&D projects. He is a member of the IEEE Technical Committee on Multimedia Communications. He serves as a reviewer for highly-respected journals such as: *J. Netw. Comput. Appl.* (Elsevier), *Int. J. of Commun. Systems* (Wiley), *J. of Systems and Software (Elsevier), Information Sciences* (Elsevier), *IETE Technical Review, Electronics* (MDPI), etc. He has served as a technical program committee member to many international conferences. His current research interests are multimedia networking and wireless ad hoc networks. He has many publications to his credit in international journals and conferences at these areas. He has edited two books on Multimedia Networking, while he serves as an editorial board member in some refereed journals.