Chapter 2
Evolution of Temporal Multimedia Synchronization Principles

Zixia Huang, Klara Nahrstedt and Ralf Steinmetz

Abstract Ever since the invention of the world's first telephone in the nineteenth century, the evolution of multimedia applications has drastically changed human life and behaviors, and has introduced new demands for multimedia synchronization. In this chapter, we present a historical view of temporal synchronization efforts with a focus on continuous multimedia (i.e., sequences of time-correlated multimedia data). We demonstrate how the development of multimedia systems has advanced the research on synchronization, and what additional challenges have been imposed by next-generation multimedia technologies. We conclude with a new application-dependent multilocation multi-demand synchronization framework to address these new challenges.

Keywords Continuous multimedia · Temporal synchronization · Evolution

2.1 Introduction

The past century has witnessed generations of multimedia applications. The maturity of storage and transmission technologies enables transition from analog modulation to digital media. The emergence of low-cost sensory devices contributes to growing popularity of multimodal multichannel data. The advancement of high-speed wireline and wireless Internet allows transmission and sharing of these multimedia information at a large scale.

Z. Huang (✉)
Google Inc., Mountain View, USA
e-mail: zixia@google.com

K. Nahrstedt
University of Illinois at Urbana-Champaign, Champaign, USA
e-mail: klara@illinois.edu

R. Steinmetz
Technische Universitat Darmstadt, Darmstadt, Germany
e-mail: ralf.steinmetz@kom.tu-darmstadt.de

M. Montagud et al. (eds.), *MediaSync*,
https://doi.org/10.1007/978-3-319-65840-7_2

Multimedia synchronization is needed to preserve original correlations among diverse multimedia data, so that they are **synchronous** (or **in-sync**) before their final presentation. There are two major synchronization categories in multimedia systems. On the one hand, **temporal synchronization** [13, 55] requires presentation of multimedia data based on their original time attributes. For example, a motion picture and an audio sample which are captured by a camera and microphone at the same time must be presented at the corresponding output devices synchronously. On the other hand, **spatial synchronization** [51] demands layout alignment of media data based on their contextual correlations at each time point. For instance, two images must show up on the left and right side of a presentation slide in the first two seconds. Existing synchronization studies mostly focus on temporal synchronization because of the time-sensitive nature of multimedia applications, which demand strict time correlations among multimedia data.

Temporal synchronization is demanded for both **continuous** and **discrete** multimedia data [13, 50]. Continuous multimedia are defined as sequences of time-correlated media packets, which are generated by one or multiple sensory devices over time. Video, audio, and haptic data are all continuous multimedia. On the contrary, discrete multimedia are the set of static media data like single images and texts, or standalone media **events** (e.g., pop-up of an image, or movement of a text). Synchronization of discrete multimedia may come with a coarse granularity where only their temporal order needs to be preserved. Hence, it is also called **event synchronization**. There have been numerous synchronization research works for both continuous and discrete multimedia. This chapter only focuses on continuous multimedia.

The configuration of a continuous multimedia system can be represented in multiple forms of media components (Sect. 2.2), where each component demands individual temporal synchronization. However, their original time dependencies at the source sensory devices can lose track in multiple locations during media computation and distribution, because of variations of Internet latencies and computation demands. The problem is called **mis-synchronization**. A mis-synchronization in one location of a multimedia system can be propagated to future locations, and multimedia data become **asynchronous** (or **out-of-sync**) during their final presentation.

Mis-synchronization of multimedia data can negatively impact human perception. Depending on application functionalities, a single multimedia system may exhibit heterogeneous demands in terms of synchronization and affects user experience at different levels.

Multimedia synchronization has already been a well-known issue in traditional multimedia applications (e.g., 2D video conferencing, on-demand video, video broadcast, etc.). **Next-generation multimedia systems** (NG-MS), like 3D tele-immersion, virtual reality, and Internet of Things (IoT), utilize multimodal multi-channel sensors to provide users an immersive and realistic experience. They are becoming more complex in terms of hardware configurations, more diverse in terms of application functionalities, and more expensive in terms of consumptions of computation and network resources. Hence, preserving the time correlations of media data in these systems becomes an even larger challenge. A systematic framework is needed to integrate application-dependent multilocation multi-demand synchronization problems, in order to achieve in-sync multimedia presentation at their final outputs. We will show that such a framework is unfortunately missing in existing studies.

In this chapter, we present a historical view of the temporal synchronization studies for continuous multimedia over the past 30 years. Based on synchronization definitions and formulations (Sect. 2.2), we demonstrate how the development of multimedia technologies has advanced the research on multimedia synchronization, and what additional issues have been imposed by NG-MS (Sect. 2.3). We conclude with a multidimensional synchronization framework to address these issues at the end of the chapter (Sect. 2.4).

2.2 Synchronization Formulation

Before we discuss existing research studies on multimedia synchronization, we formulate the term "synchronization". We present a mathematical model which will be used throughout this chapter. The mathematical symbols and their denotations are also listed in Table 2.1 in Appendix I.

2.2.1 Continuous Multimedia Data Model

The overall architecture of continuous multimedia data can be described in five categories in a hierarchical fashion.

- **Session**. A session indicates a status of multimedia communications between two or more *sites* (end systems). In this chapter, we use $\{n^1, \dots, n^N\}$ to denote N sites within the same session.
- **Bundle**. A bundle is a set of time-correlated multimedia data outputted from heterogeneous sensors of the same sender site. We denote the bundle of site n^x as u^x.
- **Media Modality**. To provide users with a realistic and immersive experience, each site may be equipped with multiple multimedia sensory devices with different modalities: 2D/3D videos, audios, haptics, etc. We let m_i^x be the i-th media modality of site n^x, i.e., $\{m_1^x, m_2^x, \dots\}$. For example, we can use $i = 1$ or "V" to represent the video modality, $i = 2$ or "A" for the audio modality, $i = 3$ or "H" for the haptic modality, etc.
- **Sensory Stream**. To preserve directionality and spatiality of the physical room environment, multiple media sensors of the same modality (e.g., a microphone array or a multi-camera array) can capture a scene at the same time, but from different angles. Each sensor produces a sensory stream $s_{i,j}^x$ (j is the stream index).

- **Media Frame**. A sensory stream is composed of a sequence of media frames (i.e., motion images and audio samples), captured by the same sensor over time. We denote the k-th media frame of $s^x_{i,j}$ as $f^x_{i,j}(k)$.

Hence, a site n^x outputs a multimedia bundle data that can include multiple media modalities, i.e., $u^x = \{m^x_1, m^x_2, \ldots\}$. Each media modality can produce multiple media streams: $m^x_i = \{s^x_{i,1}, s^x_{i,2}, \ldots\}$. For example, $m^x_{\mathrm{A}} = \{s^x_{\mathrm{A},1}, s^x_{\mathrm{A},2}, s^x_{\mathrm{A},3}\}$ represents the audio modality at site n^x with three audio streams captured by a microphone array. Each sensory stream is further composed of a sequence of media frames: $s^x_{i,j} = \{f^x_{i,j}(1), f^x_{i,j}(2), \ldots\}$. For example, $s^x_{\mathrm{V},2} = \{f^x_{\mathrm{V},2}(1), f^x_{\mathrm{V},2}(2), \ldots, f^x_{\mathrm{V},2}(k)\}$ represents the second video stream at site n^x with k media frames.

2.2.2 Layers of Synchronization Demands

Due to the hierarchical multisite multisensory nature of multimedia data, four layers of synchronization relations are demanded, where each *synchronization layer* from bottom-up is depicted in Fig. 2.1.

Intra-stream synchronization prescribes synchronous presentation of media frames within each sensory stream at receivers, according to their original captured timeline at the multimedia sensors. A mis-synchronization in this layer can cause temporal media distortion (e.g., image jerkiness or audio pitch).

Intra-media synchronization refers to synchronization of sensory streams from multiple media devices of the same media modality within a media bundle. Mis-synchronization in this layer can violate their spatial and temporal correlations during media presentation (e.g., a visual mismatch between two multi-view images).

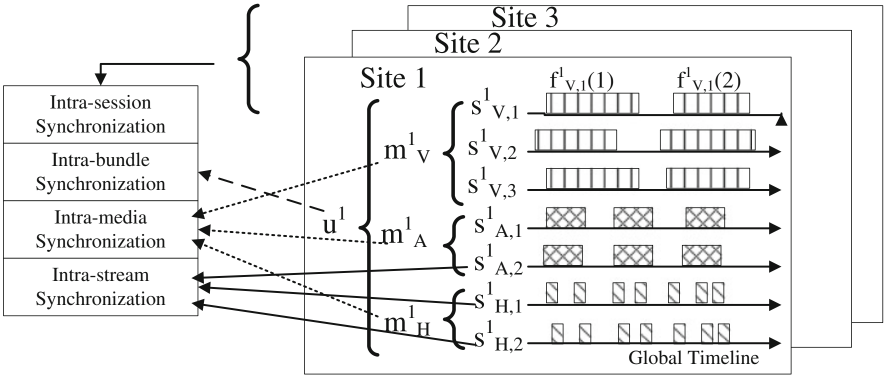


Fig. 2.1 Four layers of synchronization relations. $f^x_{i,j}(k)$ denotes the frame k in stream $s^x_{i,j}$; $s^x_{i,j}$ denotes the j-th sensory stream in media modality i="V", "A", and "H"; and m^x_i denotes the media modality in bundle u^x at site n^x ($x = 1, 2, 3$)

Intra-bundle synchronization is defined as synchronization of multiple media modalities within a bundle. This layer evaluates timing consistency across different media modalities. The most studied example would be audiovisual synchronization (e.g., lip synchronization). Note that previous studies (e.g., [13, 51, 55]) usually combine intra-media and intra-bundle synchronization demands into a single layer called **inter-stream synchronization** or **inter-media synchronization**, i.e., synchronization of multiple multimodal streams within a media bundle. In NG-MS, the scalability of media devices of each media modality and the diversity of new media devices of heterogeneous media modalities are increasing, posing very different requirements on intra-media and intra-bundle synchronization demands. Therefore, these two synchronization relations should be addressed separately.

Intra-session synchronization represents ***inter-receiver*** and/or ***inter-sender*** synchronization within a multimedia session. The inter-receiver synchronization, also named **group synchronization** or **inter-destination synchronization**, has been extensively studied by the community (e.g., [13, 17]). It refers to synchronization of media bundles from the same sender site (or media server) to multiple receivers (e.g., synchronous video playback during TV broadcast). An out-of-sync presentation can cause inconsistent interactions when multiple users at different receiver sites get a timing privilege to conduct an activity. The inter-sender synchronization, also named **inter-source synchronization**, is a new demand imposed by interactive and immersive activities. It represents in-sync presentation of media bundles from multiple senders at the same receiver (e.g., synchronous playout of 3D video streams of multiple scenes in a joint virtual space). A mis-synchronization may lead to confusion of the users when they are watching senders conducting a highly collaborative activity.

2.2.3 *Definition of Synchronization Skews*

The **synchronization skew** in a continuous multimedia setting is defined as the delay difference of two time-correlated *media objects* (media frame, sensory stream, media modality, and participating site), traveling from the media sources to the current location. One of the objects is usually the **synchronization reference**, i.e., the (most important) media object that other objects need to be synchronized against. Because of the multilayer synchronization hierarchy, a media object can be represented in multiple forms, meaning that the synchronization references must change accordingly at different layers. Thus, it is not possible to use a single skew to describe the whole multimedia session, but rather, multiple skew definitions for different layers will be more reasonable. Please note that a synchronization reference at each layer can be dynamically changed throughout a media session because of possible activity changes in a multimedia application.

Intra-stream synchronization skew. The skew within a sensory stream $s^x_{i,j}$ is evaluated by computing the delay difference of a media frame $f^x_{i,j}(k)$ w.r.t. the *reference frame* $f^x_{i,j}(*)$. We denote $D(f^x_{i,j}(k), n^y)$ as the experienced latency of $f^x_{i,j}(k)$ from its capturing time, when it is being delivered to the receiver site n^y. Thus, the intra-stream synchronization skew is defined by Eq. 2.1 as

$$\forall x, y, i, j,\ f^x_{i,j}(k) \in s^x_{i,j} \ :\ \ \Delta D(f^x_{i,j}(k), n^y) = D(f^x_{i,j}(k), n^y) - D(f^x_{i,j}(*), n^y) \qquad (2.1)$$

Intra-media synchronization skew. We denote $D(s^x_{i,j}, n^y)$ as the experienced latency of $s^x_{i,j}$ when delivered to n^y. Note that due to potential computation and Internet jitter across media frames within the sensory stream, we use the latency of the reference frame to represent that of the stream, i.e., $D(s^x_{i,j}, n^y) = D(f^x_{i,j}(*), n^y)$. Hence, the intra-media synchronization skew $\Delta D(s^x_{i,j}, n^y)$ w.r.t. the *reference stream* $s^x_{i,*}$ is defined by Eq. 2.2 as

$$\forall x, y, i, j,\ s^x_{i,j} \in m^x_i \ :\ \ \Delta D(s^x_{i,j}, n^y) = D(s^x_{i,j}, n^y) - D(s^x_{i,*}, n^y) \qquad (2.2)$$

Intra-bundle synchronization skew. Because sensory streams within a media modality can experience heterogeneous latencies, we prescribe that the latency of a media modality is equivalent to that of the intra-media synchronization reference (i.e., reference stream) within this modality, in order to best match human perceptual interests, i.e., $D(m^x_i, n^y) = D(s^x_{i,*}, n^y)$. Thus, the intra-bundle synchronization skew of m^x_i w.r.t. the *reference modality* m^x_* is defined by Eq. 2.3 as

$$\forall x, y, i,\ m^x_i \in u^x \ :\ \ \Delta D(m^x_i, n^y) = D(m^x_i, n^y) - D(m^x_*, n^y) \qquad (2.3)$$

Note that the **inter-stream synchronization skew** studied in multiple studies (e.g., [13, 51, 55]) is defined regardless of media modalities. In other words, it uses a single reference stream (denoted as s^x_*) for all other streams of different media modalities within the same bundle. The skew in these studies can be defined by Eq. 2.4 as

$$\forall x, y, i, j \ :\ \ \Delta D(s^x_{i,j}, n^y) = D(s^x_{i,j}, n^y) - D(s^x_*, n^y) \qquad (2.4)$$

There is no skew constraint between two non-reference streams in inter-stream synchronization. For example, we are unable to bound the skew between two video streams (from a multi-camera system) which use the same audio stream as the reference. This is why we propose the intra-media and intra-bundle synchronization layers separately. The issue has been neglected even in the work finished when camera/microphone arrays were being deployed [16], mainly because of the community's stereotyped view of synchronizing a single video and a single audio stream in the most common on-demand or conferencing multimedia systems.

Intra-session synchronization skew. Similar to the intra-bundle layer, we prescribe that the latency of a bundle is equivalent to that of the intra-bundle synchronization reference within the bundle, i.e., $D(u^x, n^y) = D(m^x_*, n^y)$. Given the *reference* *site* n^*, the inter-sender synchronization skew as to a receiver site n^{y_0} is defined by Eq. 2.5 as

$$\forall x: \quad \Delta D(u^x, n^{y_0}) = D(u^x, n^{y_0}) - D(u^*, n^{y_0}) \tag{2.5}$$

Accordingly, the inter-receiver synchronization (group synchronization) skew as to a sender site n^{x_0} is defined by Eq. 2.6 as

$$\forall y: \quad \Delta D(u^{x_0}, n^y) = D(u^{x_0}, n^y) - D(u^{x_0}, n^*) \tag{2.6}$$

In continuous multimedia, the synchronization skews are usually evaluated at specific time points, called ***synchronization points***. Multiple studies utilize the concept of synchronization points to evaluate synchronization skews and perform synchronization controls [76].

Based on the above formulation, we will review existing research works on multimedia synchronization. Each work addresses one or multiple layers of synchronization demands. For consistency, we will use the same set of mathematical symbols throughout the chapter.

2.3 A Historical View of Multimedia Synchronization Studies

The multimedia technologies have experienced multiple generations of evolution, with different synchronization focuses in each generation. In this chapter, we roughly divide them into four stages. In each stage, we discuss the advancement of multimedia technologies and its impact on synchronization. Figure 2.2 shows a timeline of the four stages.

2.3.1 Years of Birth: In and Before 1980s

The rise of electronic technologies had given birth to a number of analog and digital multimedia applications in early years. The rapid deployment of digital computing and communication technologies, such as PCs and Internet, and their unreliable characteristics brought people's attention to the problem of digital multimedia synchronization. However, the synchronization concept was mainly concerned with the fidelity or intelligibility of multimedia signals.

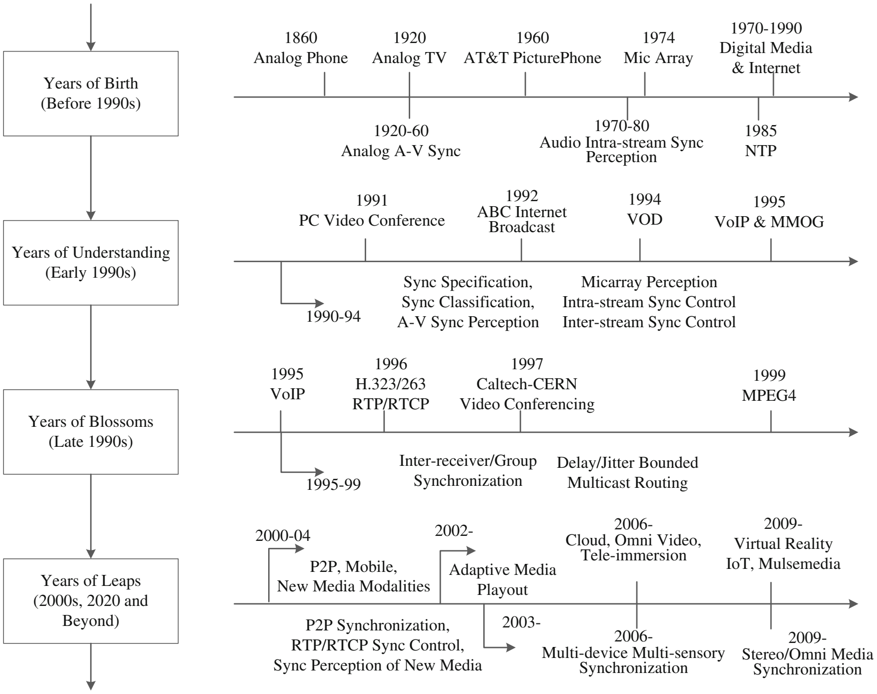

Years of Birth (Before 1990s)
1860 Analog Phone
1920 Analog TV
1960 AT&T PicturePhone
1974 Mic Array
1970-1990 Digital Media & Internet
1920-60 Analog A-V Sync
1970-80 Audio Intra-stream Sync Perception
1985 NTP
Years of Understanding (Early 1990s)
1991 PC Video Conference
1992 ABC Internet Broadcast
1994 VOD
1995 VoIP & MMOG
1990-94
Sync Specification, Sync Classification, A-V Sync Perception
Micarray Perception Intra-stream Sync Control Inter-stream Sync Control
Years of Blossoms (Late 1990s)
1995 VoIP
1996 H.323/263 RTP/RTCP
1997 Caltech-CERN Video Conferencing
1999 MPEG4
1995-99
Inter-receiver/Group Synchronization
Delay/Jitter Bounded Multicast Routing
Years of Leaps (2000s, 2020 and Beyond)
2000-04
P2P, Mobile, New Media Modalities
2002-
Adaptive Media Playout
2006- Cloud, Omni Video, Tele-immersion
2009- Virtual Reality IoT, Mulsemedia
P2P Synchronization, RTP/RTCP Sync Control, Sync Perception of New Media
2003-
2006- Multi-device Multi-sensory Synchronization
2009- Stereo/Omni Media Synchronization

Fig. 2.2 Advancement timeline of multimedia and synchronization technologies

2.3.1.1 Historical Background

Back to the years of 1860s and 1870s, the telephone device was invented to allow the analog speech transmissions over wired circuits [11], thus starting a new era of multimedia innovations. In the late 1920s, the idea of television set was proven practical, and the broadcast analog TV service rapidly developed ever since. Later in 1960s, AT&T Bell Labs demonstrated its own analog picturephone which supported a video frame rate of up to 30 fps [1]. In 1974, the microphone array (or microphone antenna) technique was invented by Billingsley [56].

Analog multimedia synchronization between audio and video had been an issue, but it was solved early on by approaches such as taking analog audio and video signals, multiplexing them and transmitting them over a controlled communication channel [1]. In addition, the quality of analog audio and video signals is not reliable; hence, it became the priority problem to solve. The concept of analog multimedia applications (radio and TV) was being accepted by people, who demonstrated more of a curiosity than an everyday demand.

2.3.1.2 Start of Synchronization Perception Studies

It was not until the 1970s and 1980s that the digital multimedia synchronization was realized as a problem. The invention of the computing machines fostered the development of digital media, while the introduction of the best-effort Internet protocols brought people's attention to the concept of delay variations (i.e., *jitter*). People became interested in how the Internet jitter affected digital media fidelity and human perception, and multiple preliminary studies were conducted to discuss the impact of jitter on intra-media synchronization of digital audios. For example, Blesser [14] offers several experimental results demonstrating that the maximum tolerable jitter for 16-bit high-quality audio is 200 ns in one sampling period. Similar work was also done in [79], which recommends a maximum allowable jitter of no more than 10 ns.

2.3.1.3 NTP: A Clock Synchronization Protocol

In 1985, David Mills proposed the first version of Network Time Protocol (NTP) in RFC 958 [2], a protocol designed for synchronizing the clocks of distributed computers connected by the Internet. NTP has gone through four iterations so far, and the latest version is published in RFC 5905 [7].

To synchronize one computing machine (called *slave*) against other (called *master*), the NTP slave computes the round-trip delays by sending a set of User Datagram Protocol (UDP) packets to the remote master. We assume that a packet leaves the slave at t_1 and arrives at the remote master at t_2 (Fig. 2.3a). We also denote that the packet leaves the master at t_3 and returns to the slave at t_4. All times are measured based on the local clocks. Hence, the clock offset between machines is defined by Eq. 2.7 as

$$\delta = \frac{(t_2 - t_1) + (t_3 - t_4)}{2} \tag{2.7}$$

Equation 2.7 implies that the synchronization approach assumes symmetrical round-trip delay. But in reality, the unequal bidirectional latency and jitter can degrade the clock synchronization accuracy. In addition, time measurement is at

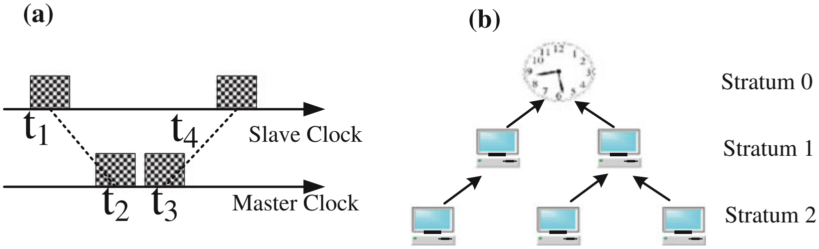


Fig. 2.3 **a** NTP clock offset computation, **b** NTP multi-stratum hierarchy

the application layer, whose accuracy depends on the underlying operating system. In general, NTP can only lead to a synchronization accuracy up to the range of 10 ms [36]. To minimize the impact of jitter and address the issue of computing machine scalability, NTP adopts a multi-stratum hierarchy (Fig. 2.3b), where machines in a *stratum layer* l are synchronized to the corresponding masters in the higher stratum layer $l - 1$ based on Eq. 2.7. A stratum layer in NTP represents the synchronization distance from the reference clock source.

NTP is important in multimedia applications, because it provides a solution to have a coherent notion of time (by accessing the same or a related global clock) across distributed machines. It allows us to identify the temporal correlation between two media objects, which are produced or are operating at different physical systems. We will show that existing studies heavily rely on this global timing state in order to achieve multimedia synchronization throughout the chapter.

2.3.2 Years of Understanding: Early 1990s

Owing to the technological advances of the Internet protocols, many Internet-based digital multimedia systems emerged and were commercialized in late 1980s and early 1990s. Multimedia synchronization became a known and important topic to the research community, and extensive amounts of research were done to understand the synchronization problem. These studies covered a broad synchronization area, including classification and specification modeling, subjective perception evaluation, and synchronization control algorithms.

2.3.2.1 Historical Background

In 1991, IBM and PictureTel introduced the first PC-based black-and-white video conferencing system [63]. In 1992, the teleorchestration service was invented as a stream-oriented interface for continuous media presentation across multiple distributed systems [18], while a real-time virtual multichannel acoustic environment was invented by Gardner based on microphone arrays [27]. The first commercial television program, ABC World News, was broadcasted over the Internet in the same year [3]. The Video On Demand (VOD) service was also started under the Cambridge project, offering video streaming at a bandwidth up to 25 Mbps [4].

The proliferation of new Internet-based multimedia systems and the improvement of digital audio-visual fidelity promoted researchers to address the synchronization problems. The Internet delay variations between the (single) audio and (single) video streams in both live and on-demand video systems exhibited a need for intra-bundle synchronization. The delay variations between multiple audio streams in the microphone array setup showed that intra-media synchronization was also important. The development of the teleorchestration service brought people's attention to inter-receiver/group synchronization. Multimedia synchronization studies thus became a hot topic during this period.

2.3.2.2 Synchronization Classification

To understand the heterogeneous demands of multimedia synchronization, a classification model was needed to identify the structure of synchronization mechanisms. Many models were proposed, with views from different aspects of the synchronization problem [61, 78].

Little et al. Model [51]. Proposed by Little, Thomas, and Ghafoor in 1991, the classification model spans over both spatial and temporal synchronization. The temporal synchronization includes intra-stream synchronization and inter-stream synchronization (i.e., stream synchronization regardless of media modalities), in spite of random network delays. Discrete timed media objects, like still images and texts, are also included in this category. However, neither spatial synchronization nor discrete media is within the scope of this paper.

Steinmetz et al. Model [13, 55]. Meyer, Effelsberg, and Steinmetz presented a more sophisticated synchronization model in 1993, based on the type of synchronization demands. The model is divided into four synchronization layers: (1) *media layer*, i.e., intra-stream synchronization; (2) *stream layer*, including inter-stream synchronization and inter-receiver/group synchronization; (3) *object layer*, describing synchronization of both continuous and discrete media objects; and (4) *specification layer*, prescribing applications and tools for synchronization specification.

Ehley et al. Model [25]. Proposed in 1994, Ehley, Furth, and Ilyas classified the synchronization technologies based upon synchronization locations, i.e., places where the synchronization control schemes are performed. However, only inter-stream synchronization was investigated in each location.

As one can see, the above three synchronization classification models are, in nature, either aligned with each other or mutually orthogonal. There is no single model which is able to cover all orthogonal dimensions.

2.3.2.3 Synchronization Specification

A further understanding of the multimedia synchronization topics requires more systematic specification methods to describe the synchronization problems. This promotes a number of specification models which can generally be grouped into four categories (Fig. 2.4), according to [13]. Here, we focus on their roles in real continuous multimedia implementations and provide a comparison in Table 2.2 in Appendix II.

Axis-based specification. First proposed by Hodges et al. in 1989 [31] and later used by [44, 59], the axis-based specification method aligns media objects in either a real or virtual global timeline axis, based on the start and finish time of each object. The accessibility of this global timeline axis is owed largely by the wide deploy-

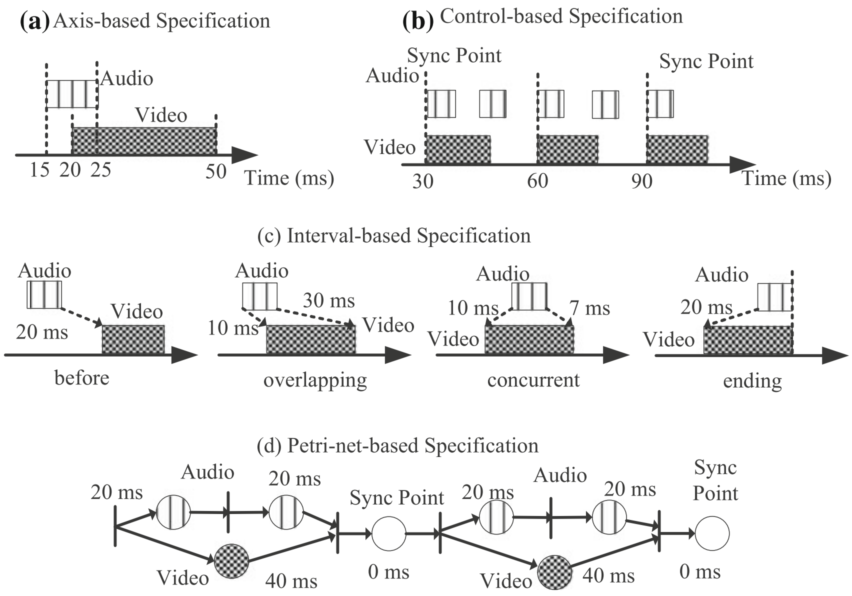


Fig. 2.4 Four synchronization specification models. Each box or circle represents a media frame

ment of NTP, and a virtual axis can be obtained by referencing the clock skews across distributed machines. The duration of each media object must be described in the specification. For example in Fig. 2.4a, we can specify that a video frame is presented between the 20th and the 50th ms, while another audio frame is played between the 15th and the 25th ms. The axis-based specification offers a direct view of temporal relations and synchronization skews of media objects in a global setting, thus facilitating its implementation in real multimedia systems. Media objects in the specification can be added and removed easily due to their mutual independence. However, media data with unknown start and finish time cannot be integrated into the axis-based method, and cannot take advantage of the benefits that this specification method provides.

Control-based specification. Developed by Steinmetz in 1990 [76] and later used by [36, 59], multimedia data are synchronized over a set of synchronization points, based on which multimedia systems can detect synchronization skews and realign the presentation of multimedia data. Oftentimes, these time points are placed periodically in order to allow consistent media re-synchronization. Figure 2.4b shows a sequence of synchronization points every 30 ms. The major advantage of this method is that it can explicitly inform users when synchronization should be performed. It also allows the integration of new media objects without major efforts. Its drawbacks are that additional mechanisms are required to specify the synchronization skews and that a timer is required to realize the periodic synchronization points.

Interval-based specification. Proposed by Wahl and Rothernel in 1994 [82] and later used by [20, 47], this specification method presents the logical temporal relations between media objects (e.g., a media object is before, after, or overlapping with another object). The exact start and finish time of each media object are unspecified. Figure 2.4c shows an example of four relations ("before", "overlapping", "concurrent", and "ending") with different delay parameter inputs. Similar to the axis-based approach, the interval-based specification is easy to understand, and adding/removing media objects is relatively simple. However, because it does not demand a knowledge of the duration of each media object and cannot describe synchronization skews, the real specification implementation can be difficult.

Petri-net-based specification. Developed by Little and Ghafoor in 1991 [51] and later used by [19, 33], this type of specification is based on Petri networks. For continuous multimedia, the specification can be described by points and arrows, where a point represents a media frame and an arrow indicates a transition state from one media frame to the other. Synchronization is achieved at each intersection of arrows. For example in Fig. 2.4d, two audio frames must be synchronized against one video frame with 0 ms synchronization skew. The Petri-net-based specification requires complex procedures to build the whole network topology. Adding and removing new media objects may also restructure the existing topology.

2.3.2.4 Synchronization Perception

As users noticed more and more audiovisual synchronization skews (i.e., perceivable lip mismatch between voices and videos) in VOD and conferencing systems over the Internet, researchers became interested in understanding the magnitude of the audiovisual skews that could be noticed by humans. A subjective study conducted by Steinmetz [77] recommends an in-sync region of a maximum 80-ms lip-sync skew when people will not perceive a lip mismatch, and an out-of-sync region of more than 160 ms when human perception can be significantly degraded. In addition, it also concludes that people are less tolerable to a skew when the video signal is behind the audio, than a skew when the audio is behind. The findings can be explained by the fact that the speed of light is much faster than the speed of sound, so people are getting accustomed to late audio signals.

In the same year, the skews between multiple acoustic streams within a microphone array were also studied in [23]. Based on subjective evaluations, it concludes that a skew of 17 ms between the stereo audio signals can be perceivable, and that a maximum skew of 11 ms is preferable.

2.3.2.5 Intra-stream and Inter-stream Synchronization Controls

To preserve temporal correlations at media presentation, a **synchronization control** scheme is often used in a multimedia system. It statically or dynamically adapts one or multiple system components, in order to mitigate synchronization skews during computation and/or distribution. Researchers began to investigate the synchronization control schemes in early 1990s, exclusively focusing on intra-stream and inter-stream synchronization for video conferencing or on-demand systems, owing to the rapid commercialization of these Internet-based applications. Most studies in those early years focused on synchronization of a single audio and a single video stream, where the audio stream was always selected as the reference stream in the master (audio)–slave (video) synchronization prototype, mainly due to the fact that the human perception is more sensitive to the degradation of audio signals. A global time was also assumed to be available between the two signals.

In this chapter, we group different studies based on both location and functionality of the synchronization control mechanisms. For synchronization location, we investigate control algorithms located and executed at both sender and receiver sides. In terms of functionality, we classify synchronization approaches that can either be shared generically by any media modality or be applied only to specific modalities.

Receiver-based Synchronization

The buffering compensation has always been the most common approach to accommodate the jitter and to minimize the inter-stream skew. To facilitate our description, we prescribe that the sender site is n^x, and the receiver site is n^y. The network delay of a media frame $f^x_{i,j}(k)$ (within the sensory stream $s^x_{i,j}$) is $D_{\mathrm{net}}(f^x_{i,j}(k), n^y)$, the buffering delay $D_{\mathrm{buf}}(f^x_{i,j}(k), n^y)$, and the resulting end-to-end latency is approximately $D_e(f^x_{i,j}(k), n^y) = D_{\mathrm{net}}(f^x_{i,j}(k), n^y) + D_{\mathrm{buf}}(f^x_{i,j}(k), n^y)$. Hence, between two buffer status updates, the following two requirements must be satisfied:

1. Intra-stream synchronization: $\forall\ k,\quad D_e(f^x_{i,j}(k), n^y)$ must remain equal, that is, $D_e(f^x_{i,j}(k), n^y) = D_e(f^x_{i,j}(*), n^y)$.
2. Inter-stream synchronization: $\forall\ i,\ j,\ \ |D_e(s^x_{i,j}, n^y) - D_e(s^x_*, n^y)| < \delta_s$ must be satisfied, where s^x_* is the inter-stream reference, and δ_s is the **synchronization threshold** of the inter-stream skew.

A synchronization threshold is defined as the maximum allowable value of the synchronization skew. It is determined by specific synchronization demands of multimedia applications.

When D_e is decided, the buffering delay of each media frame D_{buf} can be computed by subtracting the network latency D_{net} from D_e. Please note that the computation heterogeneity was usually not considered in those early years.

The abrupt change of the buffering delay D_{buf} before and right after a synchronization control update can introduce discontinuities in the media presentation processes. Most studies address this issue and mitigate the degradations of intra-stream synchronization quality based on the following methods (e.g., [9, 49, 68, 73, 83, 84]).

- **Increasing buffering latency**. There have been multiple generic approaches that can be shared among all media modalities. For example, a cost-effective way is to replicate past media frames. A more expensive approach is to interpolate media information by data prediction based on neighboring or past media frames. There are also a number of approaches that can be applied to specific media modalities. For video buffer, we can increase the video inter-frame period. For audio buffer, we may perform timescale modification without pith change (i.e., expanding the playout duration of each audio sample). We may also insert silence packets during silence periods between two utterances.

- **Decreasing buffering latency**. There have also been multiple generic or media-specific approaches. The most common generic approach is to simply skip the presentation of several media frames. For video buffer, we can decrease the video inter-frame period. With respect to audio buffer, we may also perform timescale modification without pitch change (i.e., reducing the playout duration of each audio sample) or remove silence packets during silence periods between two utterances.

Sender-based Synchronization

There are two key components in sender-based synchronization: the network bandwidth estimation and the resulting control scheme. An insufficient bandwidth can exert Internet congestion jitter and losses which can affect both intra-stream and inter-stream synchronization. Bandwidth estimation can be achieved either by packet pair probing [34] or by monitoring the receiver jitter and loss statistics via feedback control loop [67]. Based on the estimated bandwidth, the sender can perform one or multiple options of the synchronization control schemes [66, 68, 73, 83] which include (1) reducing the media sampling rate (e.g., changing audio sampling frequency from 16000 to 8000 Hz or video frame rate from 20 to 10 fps), (2) downgrading the media encoding quality (e.g., reducing video/audio encoded data rate), (3) skipping media data of low priority (e.g., only sending the I-frames during video streaming), and (4) discarding media frames that cannot meet the receiver presentation deadline (based on feedback messages from the receiver that indicate the current playout buffer status).

The sender and receiver synchronization control schemes can be employed jointly. Each scheme can be performed either reactively in response to Internet quality changes or preventively so as to reduce the chance of possible Internet quality degradations [16, 42].

2.3.3 Years of Blossoms: Late 1990s

Multimedia synchronization continued to be a hot topic due to the revolutionary change of the Internet quality and the development of sophisticated multimedia technologies. The study of inter-receiver/group synchronization with the design of media multicast overlay was the main topic in late 1990s.

2.3.3.1 Historical Background

The accessibility of broadband Internet became popular in late 1990s. This fostered the blossoms of multiple real-time applications, including the world's first commercial Voice over Internet Protocol (VoIP) service by VocalTec in 1995 [80], the first 3D massive multiplayer online game (MMOG) by 3DO Company in 1995 [22], and the Caltech-CERN project in 1997 which built a virtual room videoconferencing system that was able to connect the research centers over the world [5].

In parallel with the development of new multimedia applications, the year of 1996 gave birth to many well-known Internet Telecommunication Union (ITU) standards on multimedia codec specifications and streaming protocols, including ITU-T H.263 [45] for low-bandwidth video codec and ITU-T H.323 [46] on packet-based multimedia communication systems. The Internet Engineering Task Force (IETF), on the other hand, proposed RFC 1889 [69], the Real-time Transport Protocol (RTP), and the RTP Control Protocol (RTCP). RTP specifies a standardized packet format for delivering streaming media over the Internet, while RTCP defines the control information for RTP data. Both ITU and IETF standards have experienced several revisions since then, and RFC 1889 has been deprecated and replaced by RFC 3550 [70].

The studies of intra-stream and inter-stream synchronization continued to prevail, due to more sophisticated multimedia applications and new multimedia standards. The evolution of multi-party conferencing systems and MMOG applications, owing to tremendously enhanced Internet bandwidth availability, had sparked massive interests in providing inter-receiver/group synchronization, for the purpose of preserving the fairness and the temporal relations among the participants.

2.3.3.2 Inter-receiver/Group Synchronization Control

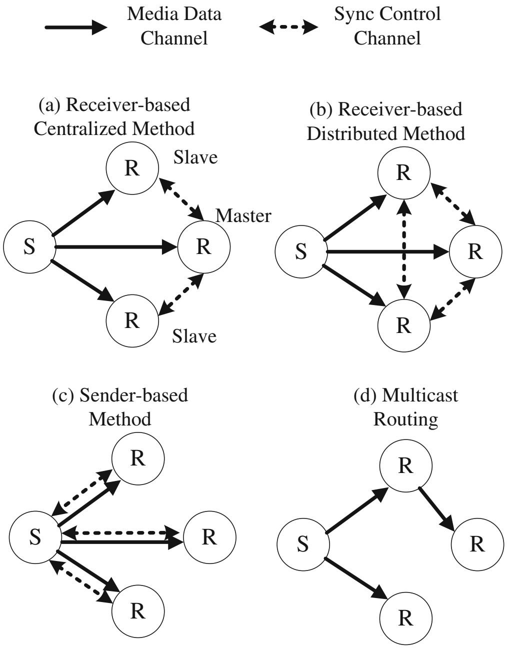
Similar to intra-stream and inter-stream synchronization, inter-receiver synchronization control schemes [16, 42, 58, 59] can also be classified based on synchronization locations and synchronization control functionalities. To facilitate the description, we describe that the sender site is n^{x_0}, and the list of receiver sites is $\{n^1, n^2, \dots\}$. We also denote that the network delay between the sender n^{x_0} and any receiver n^y is $D_{\text{net}}(u^{x_0}, n^y)$ (where u^{x_0} is the media bundle sourced at n^{x_0}), the buffering delay $D_{\text{buf}}(u^{x_0}, n^y)$, and the resulting end-to-end latency is approximately $D_e(u^{x_0}, n^y) = D_{\text{net}}(u^{x_0}, n^y) + D_{\text{buf}}(u^{x_0}, n^y)$. Here, we simplify the problem and assume negligible computation overhead at sender sites.

By denoting the synchronization reference site as n^*, the synchronization goal can be formulated by Eq. 2.8 as

$$\forall y : \quad |D_e(u^{x_0}, n^y) - D_e(u^{x_0}, n^*)| < \delta_{\text{rcv}}, \tag{2.8}$$

where δ_{rcv} is the synchronization threshold of the inter-receiver synchronization skew. To further simplify the problem, we assume zero Internet jitter in our discussion.

Fig. 2.5 Group synchronization control algorithms. S: sender site, R: receiver site



Receiver-based Synchronization

One or multiple receivers need to calculate the buffering delay D_{buf} without information from the sender site. Based on the synchronization functionalities, the receiver-based approaches can be further divided into two categories:

Centralized (master–slave) method (Fig. 2.5a). In this method, one master receiver is selected as the synchronization reference site n^*, and all other receiver sites are the slaves [8, 43]. Usually, n^* is chosen as the receiver with the longest D_{net} from the sender, i.e., $n^* = \arg\max_y D_{\text{net}}(u^{x_0}, n^y)$. The detailed procedure can be divided into the following four steps.

1. n^* first decides the one-way latency $D_e(u^{x_0}, n^*) = D_{\text{net}}(u^{x_0}, n^*)$, assuming that $D_{\text{buf}}(u^{x_0}, n^*) = 0$.
2. n^* multicasts $D_e(u^{x_0}, n^*)$ value to all other slave receivers.
3. Each slave receiver n^y measures individual $D_{\text{net}}(u^{x_0}, n^y)$ and decides its own target latency $D_e(u^{x_0}, n^y) = \max\left\{D_{\text{net}}(u^{x_0}, n^y),\ D_e(u^{x_0}, n^*) - \delta_{\text{rcv}}\right\}$.
4. n^y updates its buffering delay D_{buf}, which is approximately $D_{\text{buf}}(u^{x_0}, n^y) = D_e(u^{x_0}, n^y) - D_{\text{net}}(u^{x_0}, n^y)$.

While it is simple to implement the centralized method in real multimedia systems, there are multiple serious drawbacks that may hinder its efficient operation. First, the connectivity between the master and slave receivers cannot be guaranteed due to potential poor Internet conditions and firewall blocking issues. Second, a timely synchronization adaptation in response to sudden Internet changes is not possible. Third, scalability is a common problem in the centralized method, where the computation and network resources may be a bottleneck at the master receiver. Fourth, receiver sites can easily join and leave the session in multimedia applications like MMOG. When the master site suddenly leaves without announcement, the group synchronization will fail immediately.

Distributed method (Fig. 2.5b). In this method, each receiver site decides its own buffering delay D_{buf} in a distributed fashion [41], by periodically multicasting its D_e value to each other. The overall procedure can also be divided into four steps.

1. Each receiver site n^y multicasts its current $D_e(u^{x_0}, n^y)$ value to all other receivers.
2. A specific receiver site (denoted as n^{y_1}) waits until it receives messages from all other sites. It picks the site (denoted as n^*), which usually has the largest D_e value, i.e., $n^* = \arg\max_y D_e(u^{x_0}, n^y)$.
3. n^{y_1} measures its $D_{\mathrm{net}}(u^{x_0}, n^{y_1})$ and decides its own target latency $D_e(u^{x_0}, n^{y_1}) = \max\left\{D_{\mathrm{net}}(u^{x_0}, n^{y_1}),\ D_e(u^{x_0}, n^*) - \delta_{\mathrm{rcv}}\right\}$.
4. n^{y_1} updates its buffering delay D_{buf}, which is approximately $D_{\mathrm{buf}}(u^{x_0}, n^{y_1}) = D_e(u^{x_0}, n^{y_1}) - D_{\mathrm{net}}(u^{x_0}, n^{y_1})$.

Compared to the centralized method, frequent message exchanges among the receivers due to full-mesh communication can bring about tremendous communication overhead. In addition, because each site performs synchronization adaptations without a collaboration, the state of playout buffers of all receiver sites in the overall session may never converge under Internet dynamics (e.g., changing latency). These drawbacks prevent the adoption of the distributed method in real systems.

Sender-based (Maestro) Synchronization

Sender-based (maestro) synchronization is demonstrated in Fig. 2.5c. The receiver sites unicast individual D_{net} information to the sender site (denoted as $n^{x_0} = n^*$, which is then responsible for deciding the receiver buffering delay D_{buf} and the target end-to-end latency D_e of each receiver. The detailed procedure can be listed in five steps.

1. Each receiver site n^y measures its own latency $D_{\mathrm{net}}(u^{x_0}, n^y)$.
2. All receiver sites unicast individual $D_{\mathrm{net}}(u^{x_0}, n^y)$ value to the sender site n^{x_0}.
3. The sender site n^{x_0} selects the largest D_e latency among all receiver sites, i.e., $D_e^{\max}(u^{x_0}) = \max\left\{D_{\mathrm{net}}(u^{x_0}, n^y)\right\}$. For each receiver site n^y, n^{x_0} decides its target latency $D_e(u^{x_0}, n^y) = \max\left\{D_{\mathrm{net}}(u^{x_0}, n^y),\ D_e^{\max}(u^{x_0}) - \delta_{\mathrm{rcv}}\right\}$.
4. n^{x_0} sends $D_e(u^{x_0}, n^y)$ value to the receiver site n^y either by unicast or multicast.
5. n^y updates its buffering delay D_{buf}, which is approximately $D_{\mathrm{buf}}(u^{x_0}, n^y) = D_e(u^{x_0}, n^y) - D_{\mathrm{net}}(u^{x_0}, n^y)$.

The values of D_{net}, D_{buf}, and D_e can be piggybacked in the media packet header during bidirectional media data transmission between the sender and receivers. The resulting message exchanges can be effectively minimized. In addition, the reliability is no longer a problem when receiver sites are joining and leaving a session, as long as the sender is consistently sending media data to the receivers. The sender-based synchronization is, by far, the best method to realize the inter-receiver/group synchronization in the real systems, due to its flexibility, reliability, and the implementation easiness. However, timely synchronization adaptation is still not possible due to the round-trip latency incurred during the synchronization information exchanges.

Multicast Routing with Bounded Delay and Delay Variation

Multicast routing with bounded delay and delay variation is shown in Fig. 2.5d. It is used to control D_{net} for bounding the inter-receiver synchronization skews incurred over the Internet, rather than introducing additional buffering latencies to compensate for the skews. In multisite applications, the distribution of multimedia data from the sender to each receiver may be routed through some intermediate sites. We call it a *multicast overlay*. In designing such a topology, there can be multiple path options from the same sender to the same receiver, but via different intermediate sites. Multiple path options may feature unequal network latencies that will lead to heterogeneous inter-receiver synchronization skews. Multiple synchronization control schemes (e.g., [74, 75, 86]) have been developed to decide a multicast overlay topology with bounded inter-receiver synchronization skews. In general, the overlay design can be formulated as an optimization problem in the following form:

- **Goal**: minimizing the average D_{net} for all sender–receiver pairs.
- **Synchronization constraint** (optional): bounding the resulting delay (i.e., D_{net}) and/or delay variation (i.e., inter-receiver synchronization skew).
- **Bandwidth constraint** (optional): the inbound/outbound bandwidth utilization of each site is also a constraint.

The above problem has been proven NP-hard [86]. The optimization goal can be achieved by combining the shortest bounded path options based on the Dijkstra's algorithm as discussed in [86]. Synchronization and bandwidth constraints are realized by iterating over k-shortest path options between sender and receiver sites in order to find the one which can bound synchronization skews and/or bandwidth utilization [74, 75].

Note that if these multicast studies are employed, one must assume that multimodal multi-stream data from the same sender site follow the same distribution path to the same receiver.

Table 2.3 in Appendix II summarizes the differences of existing group control methods.

2.3.4 Years of Leaps: 2000 to Date

Modern multimedia systems are becoming more powerful in terms of accessibility of computation and network resources, more complex in terms of both hardware and software configurations, and more versatile in terms of application functionalities that can be performed. The leap of modern multimedia and networking technologies and their integration into a single platform has led to many open synchronization problems that await researchers to investigate.

2.3.4.1 Historical Background

Due to the rapid development of wireline and wireless technologies with much better network quality, modern multimedia technological advances are mainly defined by three characteristics:

Scalability. In traditional server–client or multicast systems, bandwidth becomes a bottleneck at media servers when there is a growing number of users requesting media contents. Peer-to-peer technologies are designed to address this scalability issue, so increasing number of end clients can share the bandwidth resources by allowing end clients to request media data directly from other end clients. Applications include the prototypes of MMOG [40] and peer-to-peer TV systems [85]. The cloud infrastructure, on the other hand, offers scalable Central Processing Unit (CPU) resources by outsourcing computation tasks to remote server farms.

Mobility. Mobile devices, including cell phones and tablets, became vital parts of people's everyday lives. Multimedia data are consumed on a variety of mobile terminals with different display sizes [24].

Diversity. The wide acceptance of haptics, accelerometers, body sensors, and many other sensory media (called **Mulsemedia**) in a variety of multimedia applications offers users a completely new experience. New applications can be seen in haptic desktop [53], interactive haptic painting [12], wireless body sensory network for health monitoring [52], accelerometer-based motion analysis systems [54], and many more.

The scalable multimedia applications are composed of new system components that have new demands for multimedia synchronization [81]. The mobile backhaul has very tight temporal and frequency synchronization constraints that NTP is unable to resolve. The diversity of low-cost sensory devices also requires data synchronization and affects human perception in new use applications. Multimedia synchronization becomes an even more challenging problem because of technological developments.

2.3.4.2 Precision Time Protocol (PTP)

The wireless industry demands a more precise clock source and temporal synchronization in the range of microseconds, which NTP cannot achieve. Hence, the Institute of Electrical and Electronics Engineers (IEEE) presents the new IEEE-1588 standard, named the Precision Time Protocol (PTP) [6]. Similar to NTP, PTP also uses a *master* (i.e., the device that is synchronized against) and *slave* (i.e., the device that needs synchronization) architecture to distribute synchronization packets. But PTP is able to achieve a clock synchronization accuracy up to the range of sub-microseconds in a *PTP-compliant network* (i.e., all networking devices between a

PTP master and a PTP slave need to support PTP). Compared to NTP, PTP provides the following improvements.

First, PTP packet timestamping is at the dedicated PTP chip close to the physical transmission medium, providing much better precision than NTP's application-layer measurement. The accuracy and reliability of the hardware timestamp depend on the quality of the crystal oscillator in the *PTP-compliant device* (i.e., the device that has PTP chip support). The crystal oscillator generates high-frequency pulse signals, which serve as a frequency reference to the PTP chip for high-precision timestamping. Oscillators used in PTP-compliant devices usually include Rubidium oscillator, oven-controlled crystal oscillator (OCXO), and temperature compensated crystal oscillator (TCXO). When selecting an oscillator, three factors need to be considered: (1) the time accuracy when the PTP device is freely running without a time source, (2) the short-term clock stability, and (3) the temperature-dependent clock drift. In general, rubidium provides the best quality but is also the most expensive, while TCXO is an affordable solution but with the worst stability among the three.

Second, the asymmetrical bidirectional latency introduced between an NTP master and an NTP slave is mainly caused by the delays incurred at the intermediate network devices (e.g., router or switch). To compensate for this asymmetry and provide a better clock accuracy, PTP has the notion of *transparent clock*. If a device is a transparent clock, the time that a PTP packet enters and leaves the device is recorded, and the *residence time* incurred at this device is added to the *correction field* of the PTP packet. When a PTP client decides its time drift from the PTP master, the value inside the correction field will be used, in order to compensate for the bidirectional asymmetry.

Third, for scalability, a PTP-compliant device can also be a *boundary clock*. A boundary clock can have multiple networking interfaces, where one or multiple interfaces behave as the PTP slaves of other master clocks, and the rest behave as the PTP master clocks for other slaves. When a boundary clock sees multiple master clocks from different interfaces, it uses the best master clock algorithm to select the best synchronization master, where multiple candidate master clocks are prioritized by user predefined configurations as well as clock traceability, accuracy, variance, and unique identifier.

2.3.4.3 RTP/RTCP-Based Synchronization Control Implementation

It is not until 2000s and beyond that RTP and RTCP become extensively used for real-time multimedia streaming and synchronization [15, 48, 57]. RTP defines the distribution format of media data. Three main fields are included in the RTP header that are directly related to synchronization: (1) payload type, indicating the media modality of the payload; (2) sequence number, representing the index of the RTP packet in each sensory stream for the intra-stream synchronization; and (3) timestamp, describing the local (relative) timestamp of media data units within each sensory stream, a must field for satisfying various synchronization demands. Note that RTP itself does not specify a global time status. In other words, we are unable to identify the temporal correlation across different sensory streams, without the help of other clock synchronization algorithms or protocols.

On the other hand, RTCP provides a communication channel for synchronization control support between the streams and sites. There are mainly three types of packets supported in RTCP:

1. Receiver report (RR). The receivers send RR messages to the senders specifying the packet loss rate and the jitter statistics. The RR packets may be further extended to specify the receiver buffer status [71]. This allows the sender to dynamically perform various synchronization control adaptations based on real-time streaming quality feedback, including the bandwidth allocation, and sending media data that only meet the receiver buffer deadline (as discussed in Sect. 2.3.2).
2. Sender report (SR). The senders send SR messages periodically to the receivers. An NTP/PTP (global) timestamp field is included in the SR message in order to compute the one-way latency between each sender and receiver, and to meet various synchronization demands.
3. Source description (SDES) RTCP packet. The canonical (CNAME) identifier in SDES packet is used to associate multiple media streams from a participant in multiple correlated sessions [70]. This will be useful for inter-stream synchronization.

RFC 7272 [72] investigates the use of RTCP to achieve inter-receiver synchronization. Note that both RTP and RTCP do not natively provide support for the specification of synchronization references. Hence, the reference information must be tackled in the application implementation itself.

2.3.4.4 Synchronization Perception of New Media

There are also a number of subjective studies that have investigated the impact on the human perception of synchronization skews.

Curcio and Lundan [21] evaluated the synchronization in mobile terminals with a maximum image size of 176 × 144 pixels. They show that in the mobile setting with a video frame rate below 15 fps, people are more tolerant to a synchronization error when the video spatial resolution is reduced. They also conclude that the annoying threshold of lip synchronization skew can be as large as 200–300 ms due to a degraded motion smoothness.

Ghinea and Ademoye [29] conducted perceptual measurements on the impact of synchronization skews between smell sensory data (i.e., olfaction) and audiovisual content, assuming the audiovisual lip synchronization skew is zero. Their results show a synchronization threshold of -30 s when olfaction is ahead of audiovisual data, and of $+20$ s when olfaction is behind. A skew within the synchronization threshold will not be perceived by humans. The paper also evaluates the impact of synchronization skew on the acceptability of the olfactory media. Participants are asked if "the olfactory smell was distracting" or "annoying" when the synchronization skews between olfactory and audiovisual data are introduced for different video clips. The results demonstrate that a mis-synchronization has minimal impact on the olfactory perception. Similar works have also been done in [62] which shows that people enjoy a synchronization skew of less than 5 s between olfaction and video data.

Hoshino et al. [32] measured the quality of olfactory–haptic synchronization skew. The authors conclude that the threshold of annoyance is in the range of 1–3 s.

Fujimoto et al. [26] subjectively evaluated the synchronization skew between haptics and video data. They show that a skew below 40–80 ms is hardly perceptible, and that skews greater than 300 ms are annoying.

For the (intra-media) synchronization quality of the 3D stereoscopic videos, Goldmann et al. [30] argue that a synchronization skew of 120 ms between the left and right views is satisfactory, and a skew of 280 ms can lead to poor synchronization perception. The authors also show that the human perception impact of the same synchronization skew can vary depending on heterogeneous activities.

Multiple studies have evaluated human perceptual quality of inter-receiver (or inter-destination) synchronization [28, 58, 65]. In general, people will not feel annoyance at an inter-receiver synchronization skew of less than 2 s in a social TV scenario. The number drops to 400 ms in an interactive competition, when the fairness of the game becomes a primary consideration.

There are also a number of perception-driven *adaptive media playout* (AMP) schemes for synchronization control, based on extensive subjective evaluations and feedbacks. The goal is to adapt media buffer in a way that allows people to perceive minimal noticeable differences during media presentation. These AMP schemes have been deployed in multiple applications, including video conferencing, 3D tele-immersion, and live TV multicast [35, 60, 64].

For further details on synchronization perception, readers can refer to Part 3 of the book (Chap. 10–14).

2.3.5 Remarks

Several remarks can be made from the above discussions. First, there is no classification model that can capture both multi-demand and multilocation synchronization requirements. Second, the synchronization reference is usually chosen statically (e.g., the audio for inter-stream synchronization). However, new multimedia systems are not limited to traditional conferencing and on-demand applications, and the audio information may not be the most important media data. Third, most of existing studies focus on the skews incurred over the Internet. None of them manages to investigate the heterogeneity of the computation demands and to integrate the multilocation synchronization controls systematically and consistently in a single multimedia application. In the next section, we will show that the synchronization-related issues mentioned above have become a challenge in NG-MS. We will present solutions in addressing these issues.

2.4 Synchronization in Next-Generation Multimedia Systems

NG-MS, like 3D tele-immersion (TI), Omnidirectional video, Virtual Reality (VR), and Internet of Things (IoT) applications, relies on rich multimodal multichannel media contents to provide geographically distributed users with a joint and realistic experience. Existing synchronization models and synchronization control schemes (as discussed in Sect. 2.3) show lots of limitations because synchronization in NG-MS is characterized by the following three attributes:

1. **Demands of scale and device heterogeneity**. Multiple sensory devices with heterogeneous media modalities can be configured in an NG-MS (e.g., multi-view videos, spatial audios, etc.). This requires both intra-media and intra-bundle synchronization. The immersive environment adds the demand for inter-sender synchronization, in addition to inter-receiver synchronization, to preserve the seamless interaction among both the sender and receiver sites.
2. **Multilocation synchronization controls**. An NG-MS can generally be divided into multiple locations, each of which can affect the synchronization skews. As an example, let us consider the TI system shown in Fig. 2.6. At the *capturing tier*, the sender site captures time-dependent multimodal media frames and encodes them in real time. The computation heterogeneity can contribute to the skews in all synchronization layers, as defined in Sect. 2.2.2. At the *distribution tier*, multimodal multi-stream data are sent from each sender gateway to multiple receivers. Synchronization skews are mainly caused by the Internet jitter and the use of an overlay network to distribute media contents. At the *presentation tier*, the multimedia streams are decoded and played at the corresponding output devices. The buffering latency is often introduced to compensate for the synchronization skew that has accumulated so far.
3. **Diverse applications on a single multimedia platform**. A variety of applications can be served on a single TI platform, including media consulting, remote education, and collaborative gaming. Different media modalities and sensory streams can have varying contributions to the functionality of each application, so they will have a different impact on the human perception [36]. Because the synchronization references usually represent the most important media information against which to synchronize, they must be selected depending on the user activities and their specific underlying application functionalities.

New synchronization attributes, arising from next-generation advanced multimedia and networking technologies, are not fully addressed in existing practices and standards. Hence, we will present next a multidimensional synchronization model that aims to work in the setting of NG-MS.

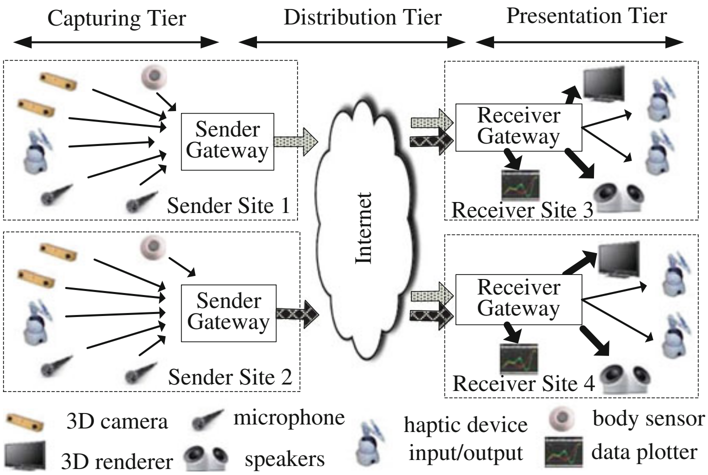


Fig. 2.6 The general architecture of TI systems

2.4.1 *New Multidimensional Synchronization Classification Model*

The scale and device heterogeneity, the multilocation synchronization control, and the application diversity of NG-MS, like the TI system, require considerations of three orthogonal dimensions in a synchronization model when performing temporal multimedia synchronization evaluations. The past models only captured one of the dimensions (e.g., Ehley's model considered only the location, while Steinmetz's model considered only the device heterogeneity). Here, we present a possible next-generation multidimensional synchronization model (Fig. 2.7), which includes the following:

1. **Dimension of scale and device heterogeneity**. This dimension is based on Steinmetz's model [13, 55]. It includes five layers. Four layers are used to meet the synchronization demands that we have discussed: intra-stream, intra-media, intra-bundle, and intra-session layers. The object layer in Steinmetz's model is removed because we only focus on continuous multimedia streams. The fifth layer, the specification layer, is used to specify the synchronization requirements of a multimedia application. Depending on the availability of sensory devices and participant sites, a multimedia application may only need a subset of the four synchronization demands. The specification layer also specifies the synchronization references and defines synchronization skews. Because the synchronization references may be updated online throughout a multimedia session, the specification layer should recompute synchronization skews based on the new references accordingly.

2. **Dimension of multilocation synchronization controls**. The orthogonal location-based dimension is directly extended from Ehley's model [25]. The location can either be a subcomponent of the media processing pipeline or an aggregation point during media distribution. We believe that there is a need to achieve multimedia synchronization in all locations, so that synchronization skews in one location will not be propagated to future locations. Hence, the multidimensional synchronization model adds the synchronization control at each location together with temporal support for large scale of heterogeneous devices.
3. **Dimension of application-dependent synchronization**. We argue that there is a strong demand to add this additional dimension, because NG-MS has a wider use space that can have numerous applications in different contexts. The dimension is used to describe the impact of the application heterogeneity on human perception of multimedia synchronization. It is not possible to use uniform synchronization references in a multimedia system that can have multiple applications. Each application achieved by a multimedia system must identify its own references based upon the functionality of performed activities and end user interests. Please note that this dimension must determine the synchronization references and work jointly with the specification layer in the dimension of scale and device heterogeneity, so that synchronization skews can be formulated based on specific applications. Appendix III presents an example of synchronization reference selection policy used in our current TI implementation.

2.4.2 Multilocation Collaborative Synchronization Controls

To demonstrate the usage of the multidimensional synchronization model in Fig. 2.7, we present the multilayer temporal synchronization control scheme at multiple locations (tiers) of the TI system, shown in Fig. 2.6. We rely on the RTP/RTCP protocol stack to achieve TI synchronization implementation.

2.4.2.1 Capturing Tier Control

The purpose of the capturing tier control is to constrain the synchronization skews arising from the computation heterogeneity of multimodal media data sourced at the same sender site. The heterogeneity is due to the fact that multiple time-correlated media frames can carry different amounts of media information, which require unequal CPU resources. The resulting variations of the computation overhead within and across the sensory streams cause intra-stream, intra-media, and intra-bundle skews.

To achieve bounded skews at the capturing tier, we utilize CloudStream [37], a cloud-based media encoding parallelization and scheduling scheme for data-intensive media, like 3D multi-view videos. The computation tasks are outsourced to cloud server farms to generate multiple resolutions of video streams to support a growing

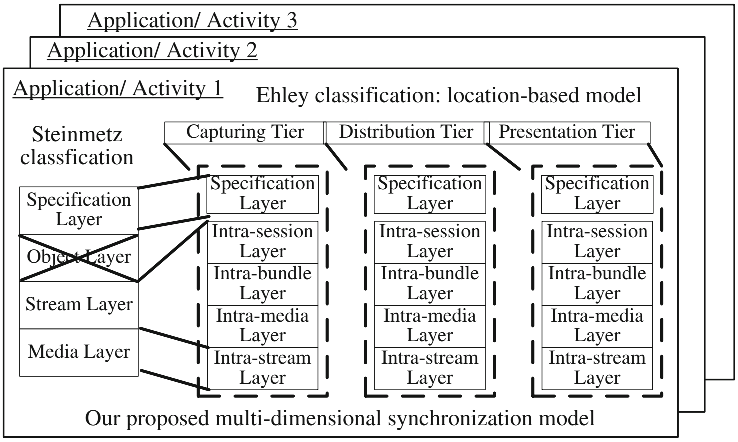
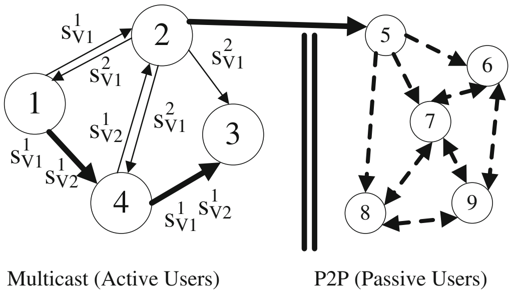


Fig. 2.7 The multidimensional synchronization model

number of end devices that demand different video qualities. In CloudStream, the media parallelization pipeline speeds up the computation process of data-intensive media modalities, by dividing each media frame (e.g., multi-view image) into multiple nonoverlapping data partitions, and encoding these partitions in parallel on multiple cloud nodes. This can effectively reduce the intra-bundle skews when comparing to media modalities with negligible overhead. The system scheduling component decides the correct amounts of computation resources (e.g., number of requested cloud nodes) for parallel encoding of each media frame. Different media frames of a same sensory stream may use different computation resources. This allows much reduced computation jitter (i.e., difference in encoding time) among media frames of a same sensory stream and across multiple streams. The computation jitter is closely related to the intra-media and intra-stream synchronization.

Fig. 2.8 A hybrid (multicast+P2P) synchronized distribution topology



2.4.2.2 Distribution Tier Control

The goal is to design an overlay topology with bounded synchronization skews during the media distribution over the Internet. A TI system is a combination of interactive and on-demand applications, because some *active receivers* are participating in or will join the shared activity, while other *passive receivers* are simply watching the active users conducting activities. The active receivers produce and send media packets, so they demand a much better interactive quality (lower latency) than the passive sites who only receive the media streams. Hence, we present a **hybrid** approach [10, 38], by performing media multicast among the active sites while relying on a peer-to-peer overlay for the rest of passive sites (Fig. 2.8). The hybrid approach only focuses on the multi-view video distribution and the resulting intra-session and intra-media (video) synchronization. Audio, haptic, and other media modalities are assumed to add negligible bandwidth overhead, so their packets can be multiplexed and follow the same distribution path as the video reference stream in the same media bundle. In other words, the intra-bundle and intra-media (audio, haptic, etc.) skews have already been minimized during the media distribution. In addition, the Internet jitter and the resulting intra-stream synchronization are not studied in this hybrid approach, and we assume they will be addressed in the presentation tier.

- **Multicast overlay**. The multicast overlay, proposed by Huang et al. [38], is based on solutions in [74, 75] (Sect. 2.3.3.2), which iterate over *k*-shortest path options for each sender–receiver pair, in order to find the paths which can achieve both synchronization and bandwidth constraints. Huang et al. [38] make three major extensions to [74, 75]. First, multiple video streams within the same media bundle are allowed to follow different paths from the same sender to the same receiver. For example, Fig. 2.8 shows that site 1 decides to multicast two video streams using different overlays: $s^1_{\mathrm{V},1}$ to both sites 2 and 4, and $s^1_{\mathrm{V},2}$ only to site 4. Hence, site 2 has to receive $s^1_{\mathrm{V},2}$ via the intermediate site 4. Second, previous studies only address the inter-receiver/group synchronization problem, while the overlay by Huang et al. [38] adds the constraints of both intra-media (video) and inter-sender synchronization to the problem formulation. For intra-media synchronization, all video streams captured by multiple cameras at the same site need to be synchronized, so that there is no inconsistency when changing views of that site. For inter-sender synchronization, multiple media bundles captured by different sender sites also need to be synchronized, so that the receiver sites will not watch these media bundles with temporal inconsistency. Third, the video reference streams now have priority in allocating bandwidth resources to preserve the most important synchronization information.
- **Peer-to-peer overlay**. Arefin et al. [10] follow existing studies in [85] to build a peer-to-peer distribution overlay, based on peer availability and bandwidth utilization fairness. Each video stream has its own individual distribution overlay, so multiple video streams from the same sender site can also follow different peer-to-peer paths to the same receiver site. Each peer-to-peer distribution overlay has a limited number of intermediate peer sites on the distribution path. This bounds the

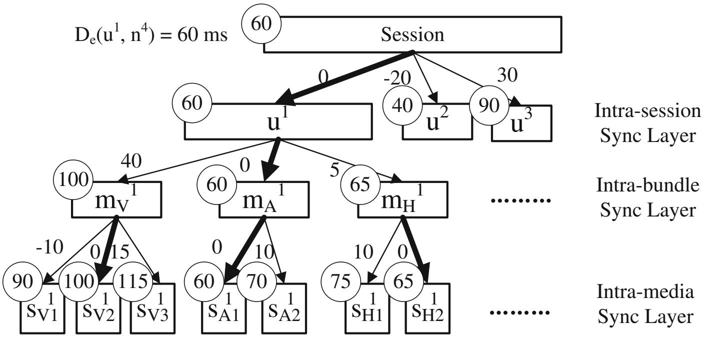


Fig. 2.9 Synchronization tree specification model

one-way video distribution latency and also constrains both intra-media (video) and intra-bundle (inter-sender and inter-receiver) skews.

2.4.2.3 Presentation Tier Control

The goal of the presentation tier control is to add buffer compensation and to bound the multilayer synchronization skews that are propagated from the capturing and distribution tiers. Existing buffer control algorithms and protocols in Sects. 2.3.2 and 2.3.3 must be extended to integrate the hierarchical synchronization references (one reference for each synchronization layer). To systematically model and visualize the interaction of synchronization layers during the buffer adaptations under scalable system configurations, we present a novel *synchronization tree* specification model.

Figure 2.9 shows an example of the specification tree formed by the receiver site n^4, where the inter-sender synchronization is demanded in the intra-session layer. Each vertex in the tree indicates a media object (i.e., session, media bundle, media modality, and media stream). The bolded edges denote the synchronization reference in each synchronization layer (see Appendix III for TI applications). The edge cost represents the synchronization skews, relative to the synchronization reference, of the media object during its presentation. In other words, the edge cost is ΔD_e, where D_e is the end-to-end latency of each media object after the buffer compensation, as discussed in Sect. 2.2.3. The vertex value (i.e., the circled number) specifies the D_e value of the corresponding media object, which can be computed by summing the edge costs on the path from the tree root to the current vertex, plus the root value. For example, we assume in Fig. 2.9 that the root value $D_e(u^1, n^4)$ is 60 ms. Hence, $\Delta D_e(u^2, n^4) = 60 - 20 = 40$ ms, $\Delta D_e(m^1_{\mathrm{H}}, n^4) = 5$ ms, and $D_e(m^1_{\mathrm{H}}, n^4) = 60 + 0 + 5 = 65$ ms. Note that the intra-stream skews should be zero due to the buffer compensation, so they are omitted in Fig. 2.9.

Based on the tree model, Huang et al. [39] propose a new buffer control algorithm. It uses an iterative approach to decide the minimal D_e of all media objects that satisfy the synchronization constraints (i.e., with bounded synchronization skews in all synchronization layers).

2.5 Conclusion

We have seen multiple generations of multimedia systems, and particularly in the past two decades, owing to the rapid development and availability of broadband Internet technologies, computation powers, and high-quality media sensors. We have shown that multimedia synchronization has always been a challenge, and lots of synchronization research works have been done in the area of models, protocols, control algorithms, distribution network, subjective perception, etc. We have defined multi-demand synchronization requirements in multiple layers and formulate synchronization skews. We have grouped achievements of synchronization research into four primary generations, based on human understanding and technological development of multimedia and synchronization systems.

In the years of birth (in and before 1980s) when digital media technologies were not mature and the Internet was still new to most people, researchers mostly focused on understanding synchronization of analog media, and NTP was proposed to achieve Internet clock synchronization at a coarse granularity. In the years of understanding (early 1990s) when the Internet became gradually adopted and multiple digital video and audio applications were invented, researchers proposed classification and specification models to understand and describe the synchronization problems. Subjective evaluations and control algorithms were mostly done for stereo audio and audiovisual synchronization, which were mostly needed during these years. In the years of blossoms (late 1990s) when broadband Internet became more available and there were growing number of multi-party multimedia applications, lots of research works were done for inter-receiver or group synchronization, in the area of video multicast, multi-party conferencing and MMOG. In the years of leaps (2000 to date) when the Internet has been part of daily life, there have been a growing number of users demanding heterogeneous of media contents via both wireline and wireless networks. Synchronization has become a larger challenge because of system scalability, demand for high-precision clock distribution over wireless medium, and human subjective perception on heterogeneous media data presented on multiple forms of end devices (TVs, PCs, and mobiles).

In the future, we foresee a revolution of distributed multimedia systems with a wider variety of multimodal sensory devices, diversity of applications and activities, and complexity of spatial and other contextual information. Due to major advances in multimodal devices, IoT, distributed and mobile computing and network technologies, and due to the drop of their integration cost, these systems are already deployed and will be deployed at much faster pace and in a much broader applications and user environments such as VR and TI spaces, smart homes and smart cities, telehealth, and other applications. Although we only use TI system as an example, we believe our generic multidimensional multi-contextual synchronization model can be extended to other applications. Multimedia data will be generated everywhere making use of a large variety of devices. Multimedia data will also be aggregated at the edges of a network as well as in the network. The evolution of multimedia systems is consistently posing new synchronization challenges. These challenges require to revisit past and current synchronization practices and standards, and demand development of new contextual-dependent approaches and principles as new multimedia environments arise.

Appendix

Appendix I: Mathematical Symbols and Denotations

Table 2.1 summarizes the mathematical symbols and denotations in this chapter.

Table 2.1 Mathematical symbols and denotations

Symbols	Denotations
t	Time
δ	Clock offset between two computing machines
x	Site index
y	Site index
i	Media modality index. $i = 1$ or "V": videos,
	$i = 2$ or "A": audios, $i = 3$ or "H": haptics
j	Sensory stream index
k	Media frame index
$*$	Synchronization reference index
n^x	Site x
n^*	Intra-session synchronization reference site
u^x	Media bundle outputted by n^x
u^*	Synchronization reference media bundle outputted by n^*
m_i^x	i-th media modality outputted by n^x
m_*^x	Intra-bundle synchronization reference modality outputted by n^x
$s_{i,j}^x$	j-th sensory stream of m_i^x outputted by n^x
$s_{i,*}^x$	Intra-media synchronization reference stream of m_i^x outputted by n^x
s_*^x	Inter-stream synchronization reference stream of u^x outputted by n^x
$f_{i,j}^x(k)$	k-th media frame of $s_{i,j}^x$ of m_i^x outputted by n^x
$f_{i,j}^x(*)$	Intra-stream synchronization reference frame of $f_{i,j}^x(k)$ outputted by n^x
D	Experienced latency of a media object
D_{net}	Latency incurred over the network
D_{buf}	Latency incurred during buffer control

(continued)

Table 2.1 (continued)

Symbols	Denotations
D_e	End-of-end latency
$D(u^x, n^y)$	Latency of u^x from its captured time, when it is being delivered to n^y
$D(m_i^x, n^y)$	Latency of m_i^x from its captured time, when it is being delivered to n^y
$D(s_{i,j}^x, n^y)$	Latency of $s_{i,j}^x$ from its captured time, when it is being delivered to n^y
$D(f_{i,j}^x(k), n^y)$	Latency of $f_{i,j}^x(k)$ from its captured time, when it is being delivered to n^y
$\Delta D(u^x, n^{y_0})$	Intra-session (inter-sender) synchronization skew of u^x against u^*,
	when it is being delivered to receiver site n^{y_0}
$\Delta D(u^{x_0}, n^y)$	Intra-session (inter-receiver) synchronization skew of u^{x_0} against n^*,
	when it is being delivered to receiver site n^y
$\Delta D(m_i^x, n^y)$	Intra-bundle synchronization skew of m_i^x against m_*^x,
	when it is being delivered to receiver site n^y
$\Delta D(s_{i,j}^x, n^y)$	Either intra-media synchronization skew of $s_{i,j}^x$ against $s_{i,*}^x$,
	or inter-stream synchronization skew of $s_{i,j}^x$ against s_*^x,
	when it is being delivered to receiver site n^y
$\Delta D(f_{i,j}^x(k), n^y)$	Intra-stream synchronization skew of $f_{i,j}^x(k)$ against $f_{i,j}^x(*)$,
	when it is being delivered to receiver site n^y
$\mathbf{O}(s_{\mathrm{V},i}^x))$	Camera orientation of $s_{\mathrm{V},i}^x$
$\mathbf{O}^{x,y}$	Desired view orientation of n^x's videos for receiver site n^y
$\mathrm{CF}(s_{\mathrm{V},i}^x, n^y)$	Contribution factor of $s_{\mathrm{V},i}^x$ to the receiver site n^y

Appendix II: Comparison Summary of Synchronization Studies

We summarize two comparison tables for the synchronization studies we have discussed in Sect. 2.3. Table 2.2 is for discussing the synchronization specification models in Sect. 2.3.2.3. Compared to interval-based and Petri-net-based specification models, Table 2.2 shows that both axis-based and control-based specification models are easy to implement and add/remove media objects, but still they require additional information and mechanisms during synchronization specifications.

Table 2.2 Comparisons of four specification models discussed in Sect. 2.3.2

Specification models	Axis	Control	Interval	Petri-net
Implementation	Easy	Easy	Complex	Complex
Media objects	Independent	Independent	Dependent	Dependent
Adding/Removing media objects	Easy	Easy	Complex	Complex
Media object duration	Required	Not required	Not required	Required
Synchronization skew	Supported	Need additional mechanism	Supported	Supported

Table 2.3 Comparisons of inter-receiver/group synchronization control algorithms

Control algorithms	Receiver-based (Master–slave)	Receiver-based (Distributed)	Sender-based (Maestro)	Multicast routing
Centralized/distributed	Centralized	Distributed	Centralized	Centralized
Adding/removing receivers	Complex if master is changed	Easy	Easy	Complex
Communication overhead	Medium	Large	Small	Large
Adaptation responsiveness	Round-trip delay	Slow	Round-trip delay	N/A

Table 2.3 is for evaluating the inter-receiver/group synchronization control algorithms in Sect. 2.3.3.2. In general, centralized approaches have lower communication overhead, and adaptive responsiveness is much faster when compared to distributed approaches.

Appendix III: Synchronization Reference Selection in Tele-immersive (TI) System

In this section, we present an example of synchronization reference selection methodology in our current TI implementation. Note that the selection rule is policy-based, meaning that it can vary depending on specific end user interests in different multimedia applications.

Intra-stream Synchronization

The reference frame or the intra-stream synchronization reference is usually selected as the first media frame within a sensory stream at each system control update. Hence, other media frames behind it can be played at the output devices by consulting their original captured inter-frame periods at the media sensor.

Intra-media Synchronization

The intra-media synchronization reference is selected as the reference stream which has the largest contribution to end user interests within a media modality. The media contribution can vary depending on the characteristics of each modality. Here, we discuss four commonly deployed media modalities which we have used.

Multi-view videos. Multi-view video streams capture the same physical object at the same time, but from different viewpoints. The importance of each video stream is decided by their contributions of 3D image pixels to the end user viewpoint [36], which can be computed using the orientation difference between the sender camera and the receiver view. Given the sender n^x's camera orientation of a video stream $s^x_{V,i}$ (denoted as $\mathbf{O}(s^x_{V,i})$), and the desired view orientation of n^x's videos for receiver site n^y (denoted as $\mathbf{O}^{x,y}$), the visual contribution or the *contribution factor* (CF) of $s^x_{V,i}$ to the receiver site n^y is defined by 2.9 as

$$\mathrm{CF}(s^{x}_{\mathrm{V},i}, n^{y}) = \mathbf{O}(s^{x}_{\mathrm{V},i}) \cdot \mathbf{O}^{x,y} \tag{2.9}$$

Hence, the video reference stream is elected as the video stream with the largest CF within the video modality for each receiver.

Spatial audios. Multiple omnidirectional microphones concurrently record the same physical ambient environment. The contribution of each audio stream is decided by its signal-to-noise ratio (SNR), a metric indicating the intelligibility of the speaker's utterances. SNR can be computed online by estimating the noises during silence periods. We prescribe that the audio reference stream is the audio stream with the largest SNR within the audio modality.

Haptics or Body sensory streams. Multiple haptic or body sensory streams may record different parts of a physical object. In the TI systems, we decide the haptic/body reference stream as the one with the largest data rate within the haptic/body sensory modality, because a larger data rate for these sensory streams usually means higher precision information.

Intra-bundle Synchronization

The importance of media modalities can vary at different applications, and the intra-bundle synchronization reference is defined as the most important reference modality. Empirically, for TI systems, we can classify different applications based on real user perceptual feedback. (1) Users attach more importance to the intelligibility of audio signals in a conversation-oriented application (e.g., conferencing or remote education), so the reference modality is the audio. (2) The clarity of video signals is of the greatest significance in a collaborative task with fine motor skills (e.g., rock–paper–scissor gaming or cyber-archeology), so the video is selected as the reference modality. (3) The body sensory streams can have the largest contribution in the telehealth or the remote rehabilitation application, because the doctors need to evaluate the patient's health status by consistent body sensory feedback. Thus, we choose the body sensory modality as the reference.

Intra-session Synchronization

In multisite interactive multimedia systems, the most active site usually demands higher quality streaming bundles in order to guarantee uninterrupted collaborations in a session. The intra-session synchronization reference of inter-sender or inter-receiver synchronization is, thus, selected as the media bundle corresponding to the most active user among all senders or receivers. In the TI systems, for example, this user usually takes the lead in the multimedia applications (e.g., a trainer in the remote education, a director in the conferencing, or a doctor in the telehealth). The selection of the lead person is context-dependent, so it must be specified explicitly by the media applications.

References

1. BELLLABS: The picture of the future. Bell Labs Rec. **47**(5), 134–186 (1969)
2. RFC-958: Network Time Protocol (NTP). http://www.ntp.org/. Accessed 28 Apr 2017
3. Cornell University: The CU-SeeMe Project. http://ftp.icm.edu.pl/packages/cu-seeme/html/Welcome.html. Accessed 28 Apr 2017
4. The Cambridge iTV Trial. http://koo.corpus.cam.ac.uk/projects/itv/. Accessed 28 Apr 2017
5. Caltech/CERN Project. http://pcbunn.cithep.caltech.edu/. Accessed 28 Apr 2017
6. IEEE-1588 standard: Precise time synchronization as the basis for real time applications in automation. https://standards.ieee.org/findstds/standard/1588-2008.html. Accessed 28 Apr 2017
7. RFC-5905: Network Time Protocol version 4: Protocol and algorithms specification. http://www.ntp.org/. Accessed 28 Apr 2017
8. Akyildiz, I.F., Yen, W.: Multimedia group synchronization protocols for integrated services networks. IEEE J. Sel. Areas Commun. **14**(1), 162–173 (1996)
9. Anderson, D.P., Homsy, G.: A continuous media I/O server and its synchronization mechanism. IEEE Comput. **24**(10), 51–57 (1991)
10. Arefin, A., Huang, Z., Nahrstedt, K., Agarwal, P.: 4D Telecast: Towards large scale multi-site and multi-view dissemination of 3DTI contents. In: Proceedings of IEEE 32nd International Conference on Distributed Computer Systems (ICDCS), Macau, China, pp. 82–91 (2012)
11. Basilio, C.: Antonio meucci inventore del telefono. Notiziario Tec. Telecommun. Ital. **12**(1), 114 (2003)
12. Baxter, B., Scheib, V., Lin, M.C., Manocha, D.: DAB: interactive haptic painting with 3D virtual brushes. In: Proceedings of ACM Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH), Los Angeles, CA, pp. 461–468 (2001)
13. Blakowski, G., Steinmetz, R.: A media synchronization survey: reference model, specification, and case studies. IEEE J. Sel. Areas Commun. **1**, 5–35 (1996)
14. Blesser, B.: Digitization of audio: a comprehensive examination of theory, implementation, and current practice. AES J. Audio Eng. Soc. **26**(10), 739–771 (1978)
15. Boronat, F., Cebollada, J.C.G., Mauri, J.L.: An RTP/RTCP based approach for multimedia group and inter-stream synchronization. Springer J. Multimedia Tools Appl. **40**(2), 285–319 (2008)
16. Boronat, F., Lloret, J., Garcia, M.: Multimedia group and inter-stream synchronization techniques: a comparative study. Elsevier Inf. Syst. **34**(1), 108–131 (2009)
17. Bulterman, D.: Specification and support of adaptable networked multimedia. Springer Multimedia Syst. **1**(2), 68–76 (1993)
18. Campbell, A., Coulson, G., Garcła, F., Hutchison, D.: Orchestration services for distributed multimedia synchronisation. In: Proceedings of IFIP International Conference on High Performance Networking (HPN), Liegel, Belgium (1992)
19. Chung, S.M., Pereira, A.L.: Timed petri net representation of SMIL. IEEE Multimedia **12**(1), 64–72 (2005)
20. Courtiat, J., de Oliveira, R.C.: Proving temporal consistency in a new multimedia synchronization model. In: Proceedings of ACM International Conference on Multimedia (MM), Boston, USA, pp. 141–152 (1996)
21. Curcio, I., Lundan, M.: Human perception of lip synchronization in mobile environment. In: Proceedings of IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM), Espoo, Finland, pp. 1–7 (2007)
22. Damer, B.: Avatars: Exploring and Building Virtual Worlds on the Internet, pp. 383–386 . Peachpit Press (1998)
23. Dannenberg, R., Stern, R.: Experiments concerning the allowable skew of two audio channels operating in the stereo mode. Pers. Commun. (1993)
24. Deventer, M., Stokking, H., Hammond, M., Cesar, P.: Standards for multi-stream and multi-device media synchronization. IEEE Commun. Mag. **54**(3), 16–21 (2016)

25. Ehley, L., Furth, B., Ilyas, M.: Evaluation of multimedia synchronization techniques. In: Proceedings of IEEE International Conference on Multimedia Computing and Systems (ICMCS), Boston, USA, pp. 110–119 (1994)
26. Fujimoto, T., Ishibashi, Y., Sugawara, S.: Influences of inter-stream synchronization error on collaborative work in haptic and visual environments. In: Proceedings of IEEE Symposium on Haptic Interfaces for Virtual Environment and Teleoperator System (HAPTICS), Reno, USA, pp. 113–119 (2008)
27. Gardner, B.: A realtime multichannel room simulator. In: Proceedings of 124th Meeting of the Acoustical Society of America, New Orleans, USA (1992)
28. Geerts, D., Vaishnavi, I., Mekuria, R., van Deventer, O., Cesar, P.: (2011) Are we in sync? Synchronization requirements for watching online video together. In: Proceedings of the 29th ACM Conference on Human Factors in Computing Systems (SIGCHI), Vancouver, Canada, pp. 311–314
29. Ghinea, G., Ademoye, O.A.: Perceived synchronization of olfactory multimedia. IEEE Trans. Syst. Man Cybern. **40**(4), 657–663 (2010)
30. Goldmann, L., Lee, J.S., Ebrahimi, T.: Temporal synchronization in stereoscopic video: Influence on quality of experience and automatic asynchrony detection, hong kong, china. In: Proceedings of IEEE International Conference on Image Processing (ICIP), pp. 3241–3244 (2010)
31. Hodges, M., Sasnett, R., Ackerman, M.: Athena Muse: a construction set for multimedia applications. IEEE Softw. **6**(1), 37–43 (1989)
32. Hoshino, S., Ishibashi, Y., Fukushima, N., Sugawara, S.: Qoe assessment in olfactory and haptic media transmission: Influence of inter-stream synchronization error. In: Proceedings of IEEE International Workshop on Communications Quality and Reliability (CQR), Naples, FL, USA, pp. 1–6 (2011)
33. Hsu, P., Chen, Y., Chang, Y.: STRPN: a petri-net approach for modeling spatial-temporal relations between moving multimedia objects. IEEE Trans. Softw. Eng. **29**(1), 63–76 (2003)
34. Hu, N., Steenkiste, P.: Estimating available bandwidth using packet pair probing. Carnegie Mellon University Techical Report, CMU-CS-02-166 (2002)
35. Huang, Z., Nahrstedt, K.: Perception-based media packet scheduling for high-quality tele-immersion. In: Proceedings of IEEE International Conference on Computer Communications (INFOCOM), Orlando, USA, pp. 29–34 (2012)
36. Huang, Z., Wu, W., Nahrstedt, K., Arefin, A., Rivas, R.: TSync: A new synchronization framework for multi-site 3D tele-immersion. In: Proceedings of ACM Workshop on Network and Operating Systems Support for Digital Audio and Video (NOSSDAV), Amsterdam, the Netherlands, pp. 39–44 (2010)
37. Huang, Z., Mei, C., Li, L., Woo, T.: CloudStream: delivering high-quality streaming video through a cloud-based H.264/SVC proxy. In: Proceedsings of IEEE International Conference on Computer Communications (INFOCOM), Shanghai, China, pp. 201–205 (2011)
38. Huang, Z., Wu, W., Nahrstedt, K., Rivas, R., Arefin, A.: Synccast: synchronized dissemination in multi-site interactive 3D tele-immersion. In: Proceedings of ACM Multimedia Systems Conference (MMSYS), San Jose, USA, pp. 69–80 (2011)
39. Huang, Z., Nahrstedt, K., Liang, K.: Human-centric multi-layer synchronization scheme with inter-sender synchronization skew control. In: Proceedings of IEEE International Workshop on Quality of Multimedia Experience (QoMEX), Singapore, pp. 25–30 (2014)
40. Iimura, T.: Zoned federation of game servers: A peer-to-peer approach to scalable multi-player online games. In: Proceedings of ACM Proceedings of 3rd ACM SIGCOMM workshop on Network and system support for games (NetGames), Portland, Oregon, pp. 116–120 (2004)
41. Ishibashi, Y., Tasaka, S.: A distributed control scheme for group synchronization in multicast communications. In: Proceedings of International Symposium Communications (ISCOM), Japan, pp. 317–323 (1999)
42. Ishibashi, Y., Tasaka, S.: A comparative survey of synchronization algorithms for continuous media in network environments. In: Proceedings of IEEE Conference on Local Computer Networks (LCN), Tampa, FL, USA, pp. 337–348 (2000)

43. Ishibashi, Y., Tsuji, A., Tasaka, S.: A group synchronization mechanism for stored media in multicast communications. In: Proceedings of Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM), Kobe, Japan, pp. 692–700 (1997)
44. ISO International Standard: Information technology hypermedia/time-based structuring language (HyTime). https://www.iso.org/standard/18834.html (1992). Accessed 28 Apr 2017
45. ITU-H263: Video coding for low bit rate communication. http://www.itu.int/rec/T-REC-H.263/en/ (2005). Accessed 28 Apr 2017
46. ITU-H323: Packet-based multimedia communications systems. http://www.itu.int/rec/T-REC-H.323/en/ (2009). Accessed 28 Apr 2017
47. King, P.: Towards a temporal logic based formalism for expressing temporal constraints in multimedia documents. Technical Report, 942, LRI, Universite de Paris-Sud, Orsay, France (1994)
48. Leroux, P., Verstraete, V., De Turck, F., Demeester, P.: Synchronized interactive services for mobile devices over IPDC/DVB-H and UMTS. In: Proceedings of IEEE/IFIP International Workshop on Broadband Convergence Networks (BCN), Munich, Germany, pp. 1–12 (2007)
49. Little, T.: A framework for synchronous delivery of time-depdent multimedia data. Springer Multimedia Syst. **1**(2), 87–94 (1993)
50. Little, T., Ghafoor, A.: Synchronization and storage models for multimedia objects. IEEE J. Sel. Areas Commun. **8**(3), 413–427 (1990)
51. Little, T., Ghafoor, A.: Spatio-temporal composition of distributed multimedia objects for value-added networks. IEEE Comput. **24**(10), 42–50 (1991)
52. Lo, B., Thiemjarus, S., King, R., Yang, G.: Body sensor network - a wireless sensor platform for pervasive healthcare monitoring. In: Proceedings of IEEE International Conference on Pervasive Computing (PERCOM), pp. 77–80 (2005)
53. Marcheschi, S., Portillo ,O., Raspolli, M., Avizzano, C., Bergamasco, M.: The haptic desktop: a novel 2D multimodal device. In: Proceedings of IEEE International Conference on Robot and Human Interactive Communication (ROMAN), Kurashiki, Okayama, Japan, pp. 521–526 (2004)
54. Mayagoitia, R.E., Nene, A.V., Veltink, P.H.: Accelerometer and rate gyroscope measurement of kinematics: an inexpensive alternative to optical motion analysis systems. Elsevier J. Biomech. **35**(4), 537–542 (2002)
55. Meyer, T., Effelsberg, W., Steinmetz, R.: A taxonomy on multimedia synchronization. In: Proceedings of IEEE Workshop on Future Trends of Distributed Computing Systems, Lisbon, Portugal, pp. 97–103 (1994)
56. Michel, U.: History of acoustic beamforming. In: Proceedings of Berlin Beamforming Conference (BeBeC), Berlin, Germany (2006)
57. Montagud, M., Boronat, F.: On the use of adaptive media playout for inter-destination synchronization. IEEE Commun. Lett. **15**(8), 863–865 (2011)
58. Montagud, M., Boronat, F., Stokking, H., van Brandenburg, R.: Inter-destination multimedia synchronization: schemes, use cases and standardization. Springer Multimedia Syst. **18**(6), 459–482 (2012)
59. Montagud, M., Boronat, F., Stokking, H., César, P.: Design, development and assessment of control schemes for IDMS in a standardized RTCP-based solution. Elsevier Comput. Netw. **70**(1), 240–259 (2014)
60. Montagud, M., Boronat, F., Roig, B., Sapena, A.: How to perform AMP? Cubic adjustments for improving the QoE. Elsevier Comput. Commun. **103**, 61–73 (2017)
61. Montagud Climent, M.A., Jansen, AJ., Cesar Garcia, PS., Boronat, F.: Review of media sync reference models: Advances and open issues. Media Synchronization Workshop (MediaSync), Brussels, Belgium (2015)
62. Murray, N., Lee, B., Qiao, Y., Muntean, G.:The influence of human factors on olfaction based mulsemedia quality of experience. In: Proceedings of IEEE International Conference on Quality of Multimedia Experience (QoMEX), Lisbon, Portugal, pp. 1–6 (2016)
63. PictureTel: Picturetel In Project With I.B.M., New York Times. http://www.nytimes.com/1991/10/22/business/company-news-picturetel-in-project-with-ibm.html (1991). Accessed 28 Apr 2017

64. Rainer, B., Timmerer, C.: A quality of experience model for adaptive media playout. In: Proceedings of IEEE International Workshop on Quality of Multimedia Experience (QoMEX), Singapore, pp. 1–4 (2014)
65. Rainer, B., Petscharnig, S., Timmerer, C.: Is one second enough? Evaluating QoE for inter-destination multimedia schronization using human computation and crowdsourcing. In: Seventh International Workshop on Quality of Multimedia Experience, pp. 1–6 (2015)
66. Ramanathan, S., Rangan, P.: Feedback techniques for intra-media continuity and inter-media synchronization in distributed media systems. Comput. J. Oxford Univ. Press **36**(1), 19–31 (1993)
67. Ramanathan, S., Rangan, P.V.: Continuous media synchronization in distributed multimedia systems. In: Proceedings of ACM International Workshop on Network and Operating System Support for Digital Audio and Video (NOSSDAV), La Jolla, CA, USA, pp. 289–296 (1992)
68. Ravindran, K., Bansal, V.: Delay compensation protocols for synchronization of multimedia data streams. IEEE Trans. Knowl. Data Eng. **4**(5), 574–589 (1993)
69. RFC-1889: Obsolete version—RTP: a transport protocol for real-time applications. http://tools.ietf.org/html/rfc1889/ (1996). Accessed 28 Apr 2017
70. RFC-3550: RTP: a transport protocol for real-time applications. http://tools.ietf.org/html/rfc3550/ (2003). Accessed 28 Apr 2017
71. RFC-3611: RTP control protocol extended reports (RTCP XR). http://tools.ietf.org/html/rfc3611/ (2003). Accessed 28 Apr 2017
72. RFC-7272: Inter-destination media synchronization (IDMS) using the RTP control protocol (RTCP). http://tools.ietf.org/html/rfc7272 (2014). Accessed 28 Apr 2017
73. Rothermel, K., Helbig, T.: An adaptive stream synchronization protocol. In: Proceedings of ACM International Workshop on Network and Operating System Support for Digital Audio and Video (NOSSDAV), Durham, NH, USA, pp. 189–202 (1995)
74. Rouskas, G.N., Baldine, I.: Multicast routing with end-to-end delay and delay variation constraints. IEEE J. Sel. Areas in Commun. **15**(3), 346–356 (1997)
75. Shi, S.Y., Turner, J.S., Waldvogel, M.: Dimensioning server access bandwidth and multicast routing in overlay networks. In: Proceedings of ACM International Workshop on Network and Operating System Support for Digital Audio and Video (NOSSDAV), Danfords on the Sound, NY, USA, pp. 83–92 (2001)
76. Steinmetz, R.: Analyse von synchronisation mechanismen mit anwendung im multimedia-bereich. In: Proceedings of GI ITG Workshop Sprachen und System zur Parallelverarbeitung, Arnoldshain, Germany, pp. 39–47 (1990)
77. Steinmetz, R.: Human perception of jitter and media synchronation. IEEE J. Sel. Areas Commun. **14**(1), 61–72 (1996)
78. Steinmetz, R., Nahrstedt, K.: Multimedia Computing, Communications and Applications. Prentice Hall (2015)
79. Stockham, T.: A/D and D/A converters: their effect on digital audio fidelity. IEEE Digital Signal Process. 55–66 (1972)
80. Tov, S.Y.: Happy 10th birthday, VoIP, The Marker. http://archive.li/TqIrI (2005). Accessed 28 Apr 2017
81. Vaishnavi, I., Cesar, P., Bulterman, D., Friedrich, O., Gunkel, S., Geerts, D.: From IPTV to synchronous shared experiences: challenges in design: distributed media synchronization. Signal Process. Image Commun. **26**, 370–377 (2011)
82. Wahl, T., Rothernel, K.: Representing time in multimedia systems. In: Proceedings of IEEE International Conference on Multimedia Computing and Systems (ICMCS), Boston, USA, pp. 538–543 (1994)
83. Woo, M., Qazi, N., Ghafoor, A.: A synchronization framework for communication of pre-orchestrated multimedia information. IEEE Netw. **1**(8), 52–61 (1994)
84. Yavatkar, R.: MCP: A protocol for coordination and temporal synchronization in collaborative applications. In: Proceedings of the IEEE International Conference Distributed Computing Systems (ICDCS), Yokohama, Japan, pp. 606–613 (1992)

85. Zhang, X., Liu, J., Li, B., shing Peter Yum T.: CoolStreaming/DONet: A data-driven overlay network for peer-to-peer live media streaming. In: Proceedings of IEEE International Conference on Computer Communications (INFOCOM), Miami, USA, pp. 2102–2111 (2005)
86. Zimmermann R, Liang K.: Spatialized audio streaming for networked virtual environments. In: Proceedings of ACM International Conference on Multimedia (MM), Vancouver, Canada, pp. 299–308 (2008)