Balanced Clock Skew Compensation for Immersive Networked Interactions Based on Inter Media Synchronization Levels

Tae-Young Lee
Center for Human-centered Interaction for Coexistence
Seoul, South Korea 02792
Email: tylee@chic.re.kr

Eunmi Lee
Center for Human-centered Interaction for Coexistence
Seoul, South Korea 02792
Email: lee.eunmi@chic.re.kr

Jaeheung Surh
Center for Human-centered Interaction for Coexistence
Seoul, South Korea 02792
Email: jhsurh@chic.re.kr

Joong-Jae Lee
Center for Human-centered Interaction for Coexistence
Seoul, South Korea 02792
Email: arbitlee@chic.re.kr

Bum-Jae You
Center for Human-centered Interaction for Coexistence and Korea Institute of Science and Technology
Seoul, South Korea 02792
Email: ybj@chic.re.kr

***Abstract*—Increasing user presence in networked social applications is an important issue for discussion to increase immersiveness. The feeling of user presence in these applications is determined not only by the types of media it supports – like video, audio, haptics, and motion capture – but also the quality of experience (QoE) in interactivity it provides. When trying to provide high QoE with such heterogeneous media streams, media synchronization is crucial. However, synchronization in itself presents a trade-off between low latency interactivity and media render quality, which are greatly impacted by playout delay and the clock skew compensation frequency, respectively. To balance this trade-off, we propose the Balanced Clock Skew Compensation, a synchronization method that exploits the boundaries for human perception of synchrony, which we define as Inter Media Synchronization Levels (IMSL). The result is a clock skew compensation approach that minimizes forward and backward jumps in media playout, which are detrimental to user perceived QoE, while maintaining low playout delay. User tests with 18 participants show that balanced clock skew compensation is more preferable than fixed frequency clock skew compensation by a large margin for interactive applications.**

***Index Terms*—Immersive Media Experiences, Synchronization, Clock Skew Compensation Frequency (CSCF), Inter Media Synchronization Level (IMSL), Quality of Experience (QoE), Human Perception**

I. Introduction

With the advent of new interactive devices and the development of network technologies, various augmented reality (AR) and virtual reality (VR) systems and applications have been available to the masses, such as the HTC Vive [1], Oculus Rift [2], Windows Mixed Reality headsets [3], and Microsoft Hololens [4]. The consumption pattern of such content differ from that of conventional media in that consumers interact through the system to share various experiences with remote users, rather than passive media consumption. An example for the application of such systems could be a Tele-immersive environment [5] where two or more people interact with each other while also being able to manipulate objects in a virtual space that they share or joint musical performances [6]. However, in a quality of experience (QoE) perspective, synchronization presents a trade-off between low latency interactions and seamless media rendering. For the system to sustain low latency interactions, the playout delay must be minimized, which requires an accurately synchronized clock to measure. In general, the simplest way to reduce clock skew is to increase the frequency of time correction with respect to a reference clock. However, this could cause frequent forward or backward time jumps, leading to media stuttering or skipping during render.

This work was supported by the Global Frontier Program on Human-centered Interaction for Coexistence (CHIC) through the National Research Foundation of Korea (NRF) funded by the Ministry of Science and ICT (NRF-2010-0029759).

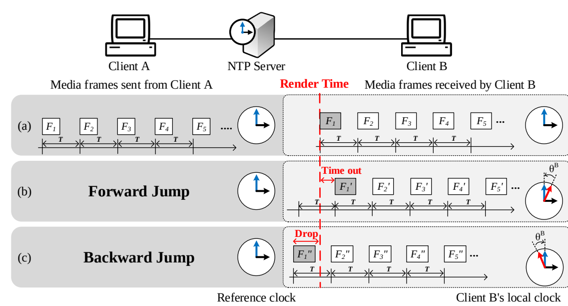

Fig. 1. No, forward, and backward time jumps due to clock skew compensation for Client B's clocks with (a) no offset, (b) positive offset, and (c) negative offset compared to the NTP server's reference clock, respectively.

Figure 1 shows how clock skew compensation can cause backward or forward jumps. In the illustrated scenario, media frames (F_1 to F_5) sent from Client A arrive at Client B intact and are rendered, while the reference clock is provided by a remote NTP Server to both clients. Frame F_1 is to be rendered at the render time calculated by Client B. Each of the 3 cases shown in Fig. 1 depicts a situation where clock skew compensation happens after the render time has been calculated and put in place. Ideally, Client B's local clock has no offset with respect to the NTP Server, as in Fig. 1(a), and therefore frame F_1 is correctly rendered at render time. However, in the case of Fig. 1(b), Client B's clock has a positive offset and causes the media rendering to freeze, or timeout, after compensating the clock backwards to the reference time. In the opposite case (Fig. 1(c)), the media rendering skips, or drops, since time is compensated forwards. As shown, time accuracy between remote users does not always guarantee good media playout quality However, the human perception of asynchrony between media is not a perfect reflection of actual synchrony. In others words, although media render may not be synchronized, if the offset is within a certain range, the average human would not be able to perceive this break in synchrony.

In this paper, we propose a clock skew compensation method that balances the accuracy of synchronization and the quality of user experience. Our method controls the balance in two aspects: 1) synchronization accuracy and user-perceived media playback quality, 2) clock accuracy and clock skew compensation overhead. We define the tolerable boundaries, or the maximum offsets, for inter media synchronization between different sets of media and refer to this as the Inter Media Synchronization Level (IMSL). An IMSL encapsulates the user-study results for human perception of asynchrony done by Juan et al. [7].

The rest of the paper is organized as follows: Section II gives an overview of the works related to clock synchronization, and important terminologies are introduced in Section III. In Section VI, we describe our proposed method in detail. Section VII shows the preliminary results of our balanced clock skew compensation, and is finally concluded in Section VIII.

II. Related Work

Many works in the literature dealing with media synchronization concentrate on time accuracy, playout delay, and methodology. The survey paper [8] extensively discusses intra media synchronization, inter media synchronization, and inter destination media synchronization. Use cases and terminologies related to synchronization are explained well in [9]. Playout scheduling schemes for intra stream synchronization are surveyed in [10]. In more recent works, solutions for synchronizing the clocks of multiple nodes to keep clock consistency [11].

Clock skew compensation methods can generally be categorized as skew-based or skewless. Clock skew-based methods are frequently found in wired distributed system environments, and works by estimating clock skew in relation to the reference time using NTP [12] or PTP [13] and compensating for it. Kalman filtering was applied to track and estimate clock offset and skew on low-cost oscillator with time-varying drifts [14]. This method handles corrupted data and does not require any time message exchange during tracking.

To enhance the quality of experience (QoE) of applications with media synchronization, Montagud *et al.* proposed the use of cubic adaptive media playout (AMP) [15] to smoothly manage playout rate, citing that small variations in media playout is subjectively less annoying compared to media jumps. Another work [16] proposes exploiting humans' imperfect perception of synchronization to adaptively output media. However, these works do not consider adaptively changing the rate of clock skew compensation, which could reduce the number of times output control is required. By minimizing the cause for the requirement of such remedies, balanced clock skew compensation could further enhance QoE.

III. Terminology

The basic terms and notations used in this paper dealing with clock skew and media synchronization are defined in this section. Much of these definitions have been adapted from the works of [17]–[21].

- **Time** is denoted in either absolute time, t, or timestamps given by a user's clock when a media frame is captured or received, $t^A_{c,M}$ or $t^A_{r,M}$, where M denotes the media type, the superscript A denotes that the time is given by user A's clock, and the subscript c and r denote capture and receive time, respectively.
- **Clock** is the entity within the system's operating system (OS) that calculates and stores the local time. In this paper, we denote the clock of client A as $C_A(t)$, that of B as $C_B(t)$, and that of the reference clock as $C_R(t)$.
- **Clock Offset** θ is the difference between a client's clock and the reference clock.
- **Clock Skew** is the differential coefficient of the clock offset.
- **Playout Delay** is the time elapsed between a given media frame is captured at the sender side and right before it is ready to be rendered at the receiver side. Playout delay is defined as $PD_M(i)$.
- **Synchronization Skew** is the time difference between the scheduled and/or actual render times of two or more time-correlated objects.

IV. Baseline Clock Skew Compensation-Based Synchronization

In this section, an overview of the baseline clock skew compensation-based system's structure and operation is described. The baseline method employs a fixed frequency clock skew compensation policy. It will be important to understand the baseline system as it will be built upon in the proposed balanced clock skew compensation.

The overall structure of the system for performing media synchronization is shown in Fig. 2. The system is divided into five modules: the data acquisition module, codec module, network module, synchronization module, and rendering module to perform synchronized data transmission. Sensor data and timestamps paired with their frame numbers are captured from each sensor's acquisition cycle. Large captured media frames, like for video, can be compressed to reduce network transmission time. The received media frame is decoded if it was compressed. After decoding, the decoded data is passed to the synchronization module to be scheduled for playback.

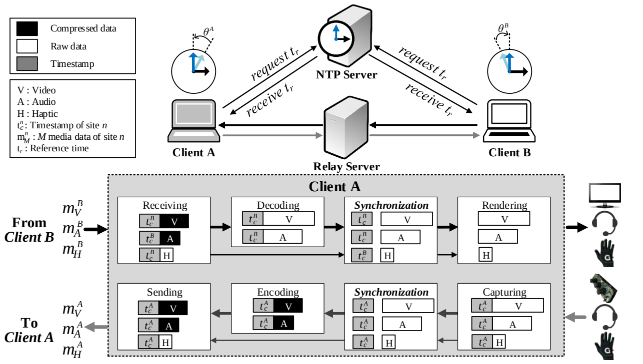

Fig. 2. Overall system architecture for media synchronization. Each client has a network, codec, synchronization, and rendering modules to perform media synchronization.

TABLE I
INTER MEDIA SYNCHRONIZATION LEVEL BETWEEN DIFFERENT MEDIA CALCULATED BASED ON [7]

Media Relation	IMSL
Video-Audio	-41ms ∼ 45ms
Haptic-Video	-45ms ∼ 45ms
Haptic-Audio	-25ms ∼ 42ms
Video-Audio-Haptic	-25ms ∼ 42ms

Playout scheduling is one of the most critical processes in synchronization in which media data are temporally rearranged to keep the temporal consistency within and between each media stream.

Using a fixed frequency clock skew compensation policy, like the baseline method, does not provide a seamless interactive media consumption experience as it introduces needless media jumps to the playback. Each compensation operation introduces this media playback jump, therefore, an adaptive compensation method would reduce such artifacts. To do this for our purposes, the boundaries for the human perception of synchrony must be quantified and a pipeline that leverages these human perceptual limitations must be implemented. Each of these two factors will be addressed in the following sections.

V. Inter Media Synchronization Level

The Inter Media Synchronization Levels (*IMSL*) are the threshold values of cognitive asynchrony between different types of media that should be rendered at the same time, as defined in Section III. The actual values we have obtained are shown the Table I and are obtained through user studies of psychophysics conducted by artificially postponing the rendering time of one media after the of rendering of a reference media [7]. Note that the asynchrony threshold value depends on the type of shared media. These values tell us that synchronization does not have to be absolute for people consuming media, but instead can be asynchronous without being perceptible as long as they are asynchronous within these boundaries. The result of user study considered video to audio, haptic to video, and haptic to audio. If relative media is rendered after the reference media, the case is defined as "$Positive(+)$," and vice versa in the "$Negative(-)$" case. To define our set of IMSL, we take the boundary values obtained in [7] and their intersections for combinations greater than two.

The IMSL value can be scaled by a coefficient σ to balance accuracy and efficiency:

$$IMSL'_{M,m} = \sigma \times IMSL_{M,m} \ , \ (0 \leq \sigma \leq 1). \qquad (1)$$

If the σ value is close to zero, it enhances clock accuracy. Conversely, if the σ value converges to 1, it focuses on efficiency for the system. The $IMSL'_{M,m}$ value then becomes the input value for the clock skew compensation frequency (CSCF) decision maker.

ISML plays an important role in our proposed clock skew compensation-based synchronization as it allows for a more lenient and less aggressive compensation policy, which in turn lowers the amount of media jumps it presents throughout playback. Compared to the baseline fixed frequency clock skew compensation policy, our system is able to adaptively change the compensation frequency lower, to minimize unneeded compensations that cause media jumps, or higher, to prevent larger jumps that may occur in the future due to skew accumulation. In short, by employing ISML as the standard for synchronization, less compensation has to be made, which increases the QoE.

VI. Balanced Clock Skew Compensation

In the proposed balanced clock skew compensation method, we add a skew balancer (Fig. 3(e)) to the baseline compensation method. The skew balancer is comprised of two parts: the playout controller (Section VI-A) that manages the playout of media and the components that manage the clock skew compensation frequency (Section VI-B). Inside the skew balancer shown in Fig. 3, the playout controller is within the area shaded in gray and the components related to the management of clock skew compensation frequency are within the white area.

A. Playout Controller

The role of the playout controller (Fig. 3(f)) is to monitor the playout delay of each media stream and determine its synchronized playout delay. The playout controller regularly monitors and calculates the exponentially weighted moving average (EWMA) [22] of the playout delay $PD_V^*(i)$, $PD_A^*(j)$, $PD_H^*(k)$ and their variations $\Delta PD_V^*(i)$, $\Delta PD_A^*(j)$, $\Delta PD_H^*(k)$ at a constant frequency for each media stream to reduce the influences of rapid network condition changes. The synchronized playout delay PD is set to that of the maximum value among media streams, making the lagging stream the reference for synchronization. The playout delays $PD_m(i)$ are sent to the inter media synchronization skew monitor within the clock skew compensation frequency management system.

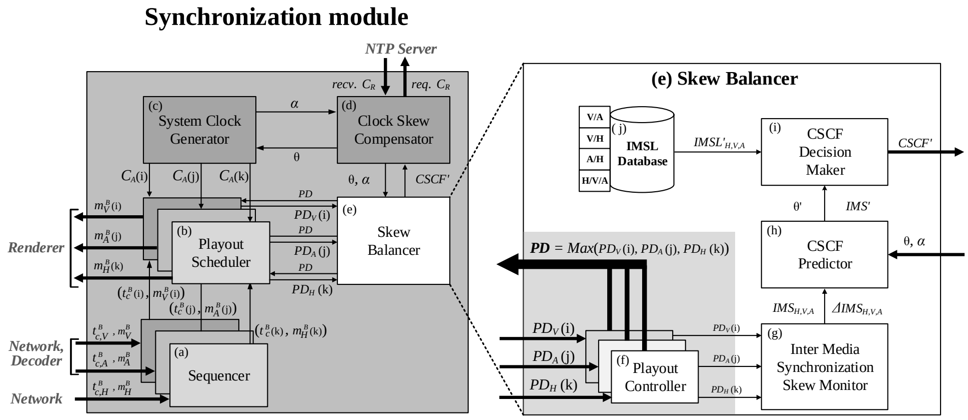

Fig. 3. Balanced Clock Skew Compensation based on IMSL in Inter Media Synchronization through the addition of the proposed skew balancer. The skew balancer includes the playout controller and the clock skew compensation frequency (CSCF) manager. Notations are marked for the scenario of Client A receiving and rendering media data from Client B.

B. Clock Skew Compensation Frequency Management

The goal of clock skew compensation frequency management is to modulate the clock skew compensation frequency (CSCF) of the clock skew compensator by monitoring the inter media skew (IMS) between media streams. The inter media synchronization skew monitor (Fig. 3(g)) first calculates the inter media synchronization skew using the playout delay calculated by the playout controller. The inter media synchronization skew information, the current clock offset, and the current clock skew are then used by the clock skew compensation frequency (CSCF) predictor (Fig. 3(h)) to predict the offset and inter media synchronization skew. These predictions and the inter media synchronization levels (ISML) definitions (Fig. 3(j)) are used by the CSCF decision maker (Fig. 3(i)) to calculate the new and more appropriate CSCF.

1) Inter Media Synchronization Skew Monitor: The inter media synchronization skew monitor's role is to observe the maximum absolute inter media synchronization skew $\varepsilon_{H,V,A}(t)$ and the change in that value $\Delta\varepsilon_{H,V,A}(t)$ between all media streams and the reference media stream. For the reference media, we used haptics, as it had the highest framerate between all three streams. The calculations are formulated as follows:

$$\begin{aligned} \varepsilon_{H,V}(t) &= PD_H(t) - PD_V(t), \\ \varepsilon_{H,A}(t) &= PD_H(t) - PD_A(t), \\ \varepsilon_{H,V,A}(t) &= \max\{|\varepsilon_{H,V}(t)|,\ |\varepsilon_{H,A}(t)|\}, \\ \Delta\varepsilon_{H,V,A}(t) &= \varepsilon_{H,V,A}(t) - \varepsilon_{H,V,A}(t-1). \end{aligned} \tag{2}$$

The absolute value of ε is used since skew in either direction are equally important. These values are then passed on to the CSCF predictor.

2) CSCF Predictor: The role of the CSCF predictor is to predict the offset and inter media synchronization skew using the inter media synchronization skew information, the current clock offset, and the current clock skew. To do this, we employ an interacting multi-model (IMM) Kalman filter [18]. The CSCF predictor takes in the clock offset θ, clock skew α, and inter media synchronization skew ε as inputs to the standard IMM Kalman filter.

TABLE II
CLOCK SKEW WITH RESPECT TO DIFFERENT CLOCK SKEW COMPENSATION FREQUENCIES (CSCF)

CSCF (Hz) Period (min)	0.016 (1)	0.033 (2)	0.050 (3)	0.066 (4)	0.083 (5)
Average Skew (ms)	-0.33	-1.19	-3.27	-4.59	-5.60
Standard Deviation (ms)	+0.92	+7.0	+8.31	+5.84	+68.90

3) CSCF Decision Maker: The CSCF decision maker decides the CSCF using the Gaussian function. The CSCF Decision Maker compares the predicted offset and inter media synchronization skew from the CSCF predictor with the values from Table II and the IMSL database, respectively. The skew statistics in Table II were obtained by monitoring the clock skew between four PCs for up to 72 hours with the given CSCF. When the inter media synchronization skew is in the range of the IMSL bounds, the CSCF decision maker does not update the CSCF, even if the predicted clock skew value is relatively high.

VII. PERFORMANCE EVALUATION

We evaluate the performance of the proposed algorithm both qualitatively, through a user study, and quantitatively. The experiments are conducted with existing methods that are used to improve media playout quality and our proposed method that uses IMSL. The first is a fixed frequency compensation policy, where the media is synchronized to the clock and the clock is compensated at a fixed interval[1]. The second method is adaptive media playout (AMP), where media playout speed is modulated to hide jumps, in conjunction with fixed frequency compensation. We show that the proposed method shows fewer jumps in media playback and allows for a more interactive and immersive experience.

[1]All fixed frequency compensation policies use a compensation frequency of 1 min.

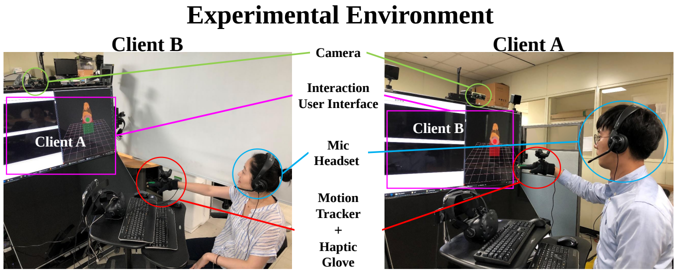

Fig. 4. Experimental environment. Each site captures and sends video, audio, and haptic data to the other site.

TABLE III
MEAN AND STANDARD DEVIATION FOR USER STUDY SURVEY SCORES PER SYNCHRONIZATION METHOD
(BEST MARKED IN RED)

	Fixed Sync (with AMP)		Fixed Sync (without AMP)		Balanced Sync	
	Average	Std Dev	Average	Std Dev	Average	Std Dev
Q1	2.11	0.66	1.61	0.95	3.89	0.66
Q2	2.11	0.57	1.5	0.9	3.44	0.83
Q3	2.28	0.8	1.44	0.96	3.89	0.74
Q4	4.22	0.85	1.5	1.01	4.78	0.42
Q5	3.27	0.93	1.56	1.26	4.78	0.53

	Questions
Q1	How well were the media streams synchronized? ($1 = bad$ to $5 = excellent$)
Q2	How desynchronized was the worst media stream? ($1 = bad$ to $5 = excellent$)
Q3	How interactive did the system feel? ($1 = bad$ to $5 = excellent$)
Q4	How often did the media stop or skip? ($1 = very\ frequently$ to $5 = none$)
Q5	How bad was/were the worst quality media streams? ($1 = bad$ to $5 = excellent$)

A. Experimental setup and scenario

The experimental environment is shown in Fig. 4 for communicating and interacting with remote users in a shared virtual space. Each site displays the virtual environment and the video feed from each site on a 55-inch full-HD 3D display. The PC used at each site is comprised of an Intel Core i7 CPU, 32GB of memory, and an NVIDIA GTX1080 running on Windows 10. The relay server contains an Intel E5 CPU, 32GB of memory, and an NVIDIA 300 series graphics card on Windows Server 2008 R2. In the interaction with two users, media frame from each client is transmitted via relay server. An NTP Server (Meinberg LANTIME M1000) provides the reference time for clock synchronization. We use several sensors: a full-HD camera (developed ourselves) for video, an HTC Vive tracker for motion, a haptic glove (Neuro Digital Technologies) for haptics, and a microphone/headphone combo for audio. The captured data from each sensor is transmitted to the other client.

The virtual environment contains a virtual hand from each client site. The virtual hand moves in sync with the motion data from the Vive tracker attached to the user's hand. Each site also displays the video stream captured from each site's webcam alongside the virtual environment on the screen. This way, the physical hand movements in the video stream and the tracked virtual hand in the virtual world can be seen simultaneously, allowing for a more complete evaluation of synchronization between media types.

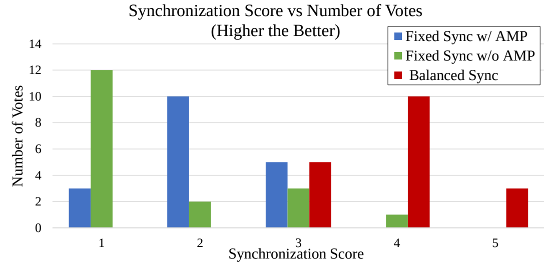

Fig. 5. User test synchronization scores for 3 different synchronization: AMP, fixed frequency-based, and balanced (proposed).

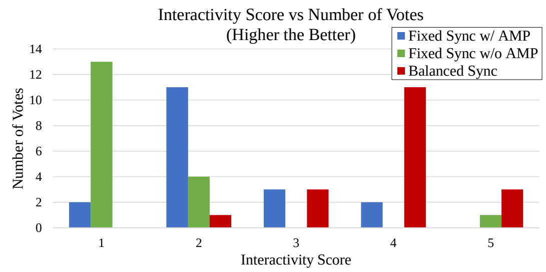

Fig. 6. User test interactivity scores for 3 different synchronization: AMP, fixed frequency-based, and balanced (proposed).

B. User Study Results

For the user study, 14 men and 4 women within the age range of 20 to 40 participated using the experimental setup with different synchronization methods. Each participant was familiarized with the system and the virtual environment (*e.g.*, the haptic interaction). Participants used the system in pairs and were given 5 minutes on each synchronization method – fixed frequency clock skew compensation with AMP, that without AMP, and the proposed balanced clock skew compensation. Each participant was instructed to answer a set of survey questions regarding the system in between switching synchronization methods. These questions are shown in the lower half of Table III. Each question is a multiple choice question using a 5 point rating scale.

The results, organized by average scores in Table III, show that balanced clock skew compensation-based synchronization consistently outperforms other tested methods. Looking at the participants' scores given for synchronization (Fig. 5) and interactivity (Fig. 6) show that participants perceive the balanced method as being more in sync and more interactive.

One comment regarding the tested systems that came up multiple times is that the audio was hard to listen to using fixed frequency synchronization. The audio would skip at random points when data was dropped and it would generate a high-pitched noise when the data was timed out. From this, we concluded that intra media synchronization is key for audio data. Another comment that repeatedly presented itself was that the participants immediately noticed the remote user's motions on video were not synchronizing with those in the

TABLE IV
NUMBER OF TIMED OUT DATA (FORWARD JUMP) AND DROPPED DATA (BACKWARD JUMP)

	Video (30 fps)		Audio (43 fps)		Haptics (60 fps)	
# of Frames	162000		232200		324000	
Jump Type	Timeout	Drop	Timeout	Drop	Timeout	Drop
Fixed Freq.	31.96%	2.90%	19.49%	0.18%	37.95%	2.48%
Ours	9.18%	2.47%	9.29%	1.72%	9.66%	1.78%

virtual environment when AMP was present, and that the video skipped and paused at times when AMP was not used. They voiced their frustration on the lack of synchronization and absence of smooth video playback, citing them as the reasons for their low scores for interactivity. Balanced synchronization, meanwhile, showed little to no desynchronization, leading to the participants preferring the proposed method over others.

C. Quantitative Analysis

A quantitative analysis on the number of drops and timeouts for each media stream for fixed frequency synchronization and balanced synchronization was done to further investigate the results. The results from around 20 minutes of system use is shown in Table IV. As can be seen from this table, each synchronization method encounters significantly more forward jumps (timeouts) than backward jumps (drops), which is understandable since increase in playout delay is much more common than otherwise. However, fixed frequency synchronization accumulates media jumps much more frequently and consistently than balanced synchronization.

It is interesting to note that jumps accumulate most frequently on haptic data. This is perhaps due to the high framerate of haptic data. However, audio data encounters less jumps compared to video even at a higher framerate, which we attribute to the relative size of each data (*i.e.*, video data is larger and more prone to timeouts). This can be seen by looking at the percentage of jumps given in Table IV. Percentage-wise, haptics and video suffer similar amounts jumps, showing that there is a trade off between framerate and frame size.

VIII. CONCLUSION

In this paper, we propose an IMSL-based clock skew compensation method to reduce the number of backward and forward jumps that occur when performing clock skew compensation, while maintaining low latency. The proposed Balanced Clock Skew Compensation balances two aspects of synchronization for networked immersive social applications. First, synchronization with IMSL reduces the number of backwards and forward jumps, achieving a balance between synchronization accuracy and quality of experience. Second, we predict the clock skew and inter media synchronization skew using a Kalman filter and adjust the CSCF accordingly, in order to balance between clock accuracy and clock skew compensation overhead. We show that the proposed method achieves better results than the fixed frequency based clock skew compensation method with or without AMP in terms of user experience and immersion. Further research may be done on ways to help this method cope with sudden changes in playout delay, perhaps with a better prediction model for delays, or ways to extend it for inter destination synchronization for support more than three users.

REFERENCES

[1] HTC, "https://www.vive.com," visited in January 2018.
[2] Oculus, "https://www.oculus.com/," visited in January 2018.
[3] Microsoft, "https://www.microsoft.com/en-us/windows/windows-mixed-reality," visited in January 2018.
[4] ——, "https://www.microsoft.com/en-us/hololens," visited in January 2018.
[5] R. Mekuria, A. Frisiello, and M. P. P. Cesar, "Network support for social 3-d immersive tele-presence with highly realistic natural and synthetic avatar users," in *7th ACM International Workshop on Massively Multiuser Virtual Environments (MMVE'15 ACM)*, 2015.
[6] M. Sithu, Y. Ishibashi, and N. Fukushima, "Effects of dynamic local lag control on sound synchronization and interactivity in joint musical performance," *ITE Transactions on Media Technology and Applications*, 2014.
[7] J. M. Silva, M. Orozco, J. Cha, A. E. Saddik, and E. M. Petriu, "Human perception of haptic-to-video and haptic-to-audio skew in multimedia applications," *ACM Transactions on Multimedia Computing, Communications and Applications*, 2013.
[8] FernandoBoronat, J. Lloret, and M. García, "Multimedia group and inter-stream synchronization techniques: A comparative study," *ACM Transactions on Multimedia Computing, Communications and Applications*, 2009.
[9] M. Montagud, F. Boronat, H. Stokking, and R. van Brandenburg, "Inter-destination multimedia synchronization: schemes, use cases and standardization," *Multimedia Systems*, 2012.
[10] N. Laoutaris and I. Stavrakakis, "Intrastream synchronization for continuous media streams: A survey of playout schedulers," *IEEE Network*, 2002.
[11] R. Ben-El-Kezadri, G. Pau, and T. Claveirole, "Turbosync: Clock synchronization for shared media networks via principal component analysis with missing data," in *IEEE INFOCOM*, 2011.
[12] D. L. Mills, "Network time protocol version 4 reference and implementation guide," 2006.
[13] "Standard for a precision clock synchronization protocol for netwroked measurement and control systems," *IEEE*, 2008.
[14] H. Kim, X. Ma, and B. R. Hamilton, "Tracking low-precision clocks with time-varying drifts using kalman filterings," in *IEEE/ACM Transactions on Networking*, 2012.
[15] M. Montagud, F. Boronat, B. Roig, and A. Sapena, "How to perform amp? cubic adjustments for improving the qoe," *ELSEVIER Computer Communication*, 2017.
[16] P. Huang, Y. Ishibashi, and M. Sithu, "Enhancement of simultaneous output-timing control with human perception of synchronization errors among multiple destinations," in *Computer and Communications (ICCC), 2016 2nd IEEE International Conference on*, 2016.
[17] Z. Yang, J. Pan, and L. Cai, "Adaptive clock skew estimation with interactive multi model kalman filters for sensor networks," in *IEEE International Conference on Communications*, 2010.
[18] Z. Yang, L. Cai, Y. Liu, and J. Pan, "Environment-aware clock skew estimation and synchronization for wireless sensor networks," in *IEEE INFOCOM*, 2012.
[19] Z. Huang, K. Nahrstedt, and R. Steinmetz, "Evolution of temporal multimedia synchronization principles: A historical viewpoint," *ACM Transactions on Multimedia Computing Communcations and Applications*, 2013.
[20] S. U. Din and D. Bulterman, "Synchronization techniques in distributed multimedia presentation," in *4th International Conferences on Advances in Multimedia(IARIA MMEDIA 2012)*, 2012.
[21] K. O. Saputra, W.-C. Teng, and T.-H. Chen, "Hough transform-based clock skew measurement over network," in *IEEE Transactions on Instrumentation and Measurement*, 2015.
[22] P. Cisar, S. Bošnjak, and S. M. Cisar, "Ewma algorithm in network practice," *International Journal of Computers Communications & Control*, 2010.