

M.A.S.: A Protocol for a Musical Session in a Sound Field where Synchronization between Musical Notes is not guaranteed

Yuka Obu*, Tomoyuki Kato**, Tatsuhiko Yonekura*

*Graduate School of Science and Engineering, Ibaraki University, 4-12-1 Nakanarusawa-cho, Hitachi-shi, Ibaraki, 316-8511, Japan
e-mail: {nm02002r, yonekura}@cis.ibaraki.ac.jp

**KORG INC. 1-15-12 Simotakaido, Suginami-ku, Tokyo, 168-0073, Japan

Abstract

When a musical session is performed via the network, it is necessary to interact in real time; however, there is the problem of delay, and the time lag between the musical notes may become an impediment. For this problem, we propose a new protocol for a musical session called Mutual Anticipated Session (M.A.S.), which is a type of ensemble that controls the timing of the sounds and composes music like in a canon-like style. In the M.A.S., one player's performance precedes the other players', so we call this performance "precedent musical performance", and within the precedent musical performance, we call the time lapse between the players' performance "precedent time." The remote ensemble system is constructed by using M.A.S., and we investigate the usability of the M.A.S. system and the suitability of the precedent time.

1 Introduction

Recently, two-way application systems on the Internet, such as chat, on-line games, and real-time music software [1], have become increasingly popular. There is, however, the problem of delay on the Internet. When we perform a musical session via the network, we need to interact in real time, while the lag between the sounds, which is the lag between the periods when musical note sounds, may become an impediment. When lag is generated by the network delay, rhythms between the players might be perturbed and it becomes difficult to continue the session.

To solve this problem, "Open Remote GIG [2]" and "GDSM (Global Delayed Session Music) [3]" have been proposed. Open Remote GIG uses a protocol called RMCP [4][5][6]. In Open Remote GIG, distributed players can interact over the Internet under the assumption that the tempo is constant and the chord progression is repetitive, like that of 12-bar blues. Each player can improvise while listening to the others' performances that are intentionally delayed for

multiples of the constant period of the repetitive chord progression. In this method, it takes double length of the period for the player to recognize the feedback of the performance. For example, if the period of the chord progression is 12-bar blues, the feedback returns 24 measures later. Therefore, quality of the musical interaction obviously decreases. In GDSM, basic accompaniment is played repetitively, and player improvises while listening to the accompaniment and the others' performances. The performances of the other players start at the beginning of the repetition period. In this method, each player listens to the different music which can't be expected at each site. Accordingly, a player can't influence over the other players as he intended. This causes the decline of the quality of the interaction.

On the other hand, a few reports have been made about synchronization protocol among several sound generation nodes on the Ethernet [7][8][9][10][11].

Another ways to overcome the problem of delay, modeling of distribution of the network transaction, acceleration of the Internet and compression method of transmission data etc. have been studied [12]. These aspects, however, may not be the essential solution for reduction of the lag between the sounds.

In this paper, therefore, we propose a new protocol for musical sessions, namely, Mutual Anticipated Session (M.A.S.), which is a type of ensemble that controls the timing of the sounds and composes music in a canon-like style.

2 Asynchronous Sound Field

2.1 The problem of lag between sounds

Human auditory temporal resolution is high, as it can sense a few milliseconds difference in the timing of a sound. A human can detect the difference in impression when the gap of the timing of the sounds, which are played as a chord, is about 5 or 10 ms [13]. In a musical ensemble, if

the sounds produced by each player are sensed to have different timing, it may become difficult to perform.

When a musical ensemble is performed between two places that are geographically distant from each other, the produced sounds are received with delay, which is the case with an ensemble transmitted over the Internet, where the data transmission is influenced by lag. If the lag between the sounds is less than 50 ms, the players may be able to accommodate themselves to the delay. However if the lag is as great as 300 ms, which is almost equivalent to an eighth note in temporal length in a musical score whose tempo is 120 bpm, rhythms between the players might be perturbed. Moreover, the more closely the delayed sounds are followed by the players, the more the delay is amplified in turn. Consequently, it becomes almost impossible to continue the performance.

2.2 Asynchronous sound field

In this paper, the sound field described above is referred to as the asynchronous field, that is, the sound field where synchronization between the musical notes is not guaranteed.

To perform the musical ensemble in the asynchronous sound field, we have to control the timing of the sounds so that players are not perturbed.

3 Mutual Anticipated Session

3.1 Precedent musical performance

When the lag between the sounds is sensed, it is difficult to continue the performance. However, because of the network lag, it is impossible to avoid the lag when the remote musical ensemble is performed via the network. Thus we propose a new protocol for musical ensembles in the asynchronous sound field.

The player cannot sound the musical note prior to the reception of the data. However, after the reception, it is possible to sound the notes with a delay of arbitrary length. Accordingly, we dare to place an intentional delay into the timing of the sound, and because of this, the players may be able to perform comfortably and the ensemble between the two places can be realized [Fig. 1]. In this protocol, one player's performance precedes the other players', so we call this protocol "precedent musical performance", and within the precedent musical performance, we call the time lapse between the players' performance "precedent time."

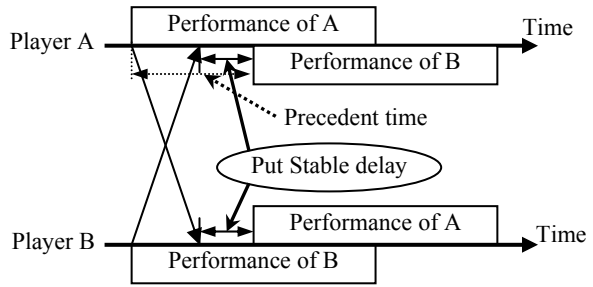


Fig. 1: Precedent Musical Performance

3.2 Mutual Anticipated Session

A Mutual Anticipated Session (M.A.S.) is a type of ensemble that uses the precedent musical performance. In the precedent musical performance, we need to know the appropriate precedent time length, and for this point, canon-like style is a good reference point and motivates us to find the optimum length of the time suitable for the M.A.S.

The interval between beats seems to be important in music, and this interval may be one of the significant human factors that help meter perception. Therefore, we think that it is reasonable to state that when the delayed timing of the notes goes with each beat, it may become easier to perform the musical session.

We distributed a questionnaire survey, and the feedback confirmed that if the music is four-four time, a precedent time of two-beat or four-beat is suitable for the M.A.S.

4 Implementation of M.A.S.

The remote ensemble system has been constructed by using the protocol mentioned above.

4.1 Server and Client

We assume that this system transmits and receives the data via LAN or WAN. We employ a server and a client model for the prototype system [Fig. 2]. With several clients being connected to one server, several players can realize the ensemble. The server controls the starting time of the performance and the timing of each sound. Clients transmit the data via the server.

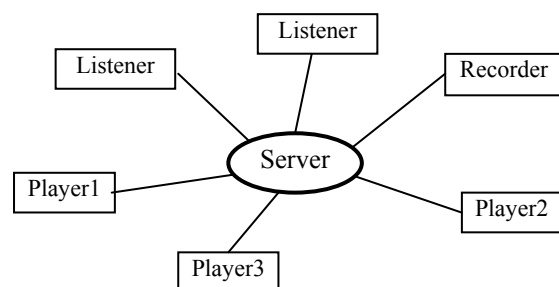


Fig. 2: Server and Client

This form of transmission of data from client-A to client-B takes a longer time than direct transmission of the data; however, the lag due to the Internet may not significantly influence the timing because two-beat (four-beat) is converted into a temporal length of 1000 ms (2000 ms) in the case of the tempo being 120 bpm, while the delay on the Internet is not more than tens of milliseconds.

The players and audience all exist at clients. By being connected to the server as audience clients, audiences can listen to and record the performance in real time without any extra process. For the audience, all players' performances are delayed; therefore, although the player performs a precedent musical performance, the audience may be able to listen to all players in synchrony in real time [Fig. 3].

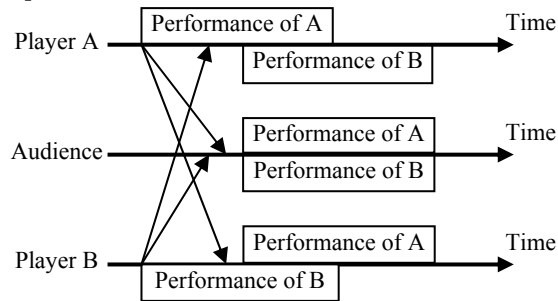


Fig. 3: Player and Audience

We employed TCP and UDP for the transmission. TCP is the connection type and is highly reliable, but because of the transmission process' complexity, it may not be suitable for the data transmission of the performance. On the other hand, UDP is the connectionless type and can get rid of this complexity, although it may cause packet loss. Therefore, in this system, TCP is employed to transmit the important data such as tempo or the signal for the start of the performance, whereas UDP is employed for the data transmission in the performance [Fig. 4].

Although using UDP risks the loss of data of a note due to packet loss, it may not have any negative influence on the performance. This is because the players can continue the performance, even if they make a mistake and a few notes have been lost. This system uses MIDI as the sound data and one note consists of the NOTE ON and NOTE OFF message. For that reason, if the NOTE OFF message has been lost, the sound cannot be stopped. As a result, a certain method is required to stop the sound when the packet loss of the NOTE OFF message has occurred.

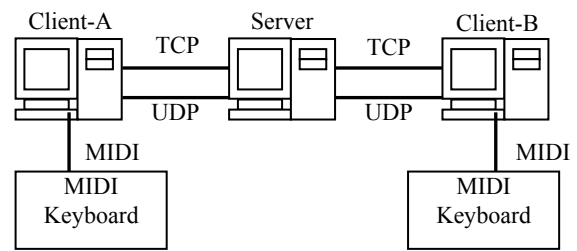


Fig. 4: Construction of the system

4.2 Time control

To realize the precedent musical performance, it is necessary to delay one player's performance from the other players' with exact precedent time. In this system, the client-A sends the data to client-B before the timing of the sound, and client-B waits for the note to be sounded.

To control the timing of the sound, each client has a timer. After receiving the message to start the ensemble, the client resets and starts the timer. The client uses this timer to record the time when the data of the note is inputted and to control the timing of the sound. Because of the network delay, the timer does not start at exactly the same time for each client.

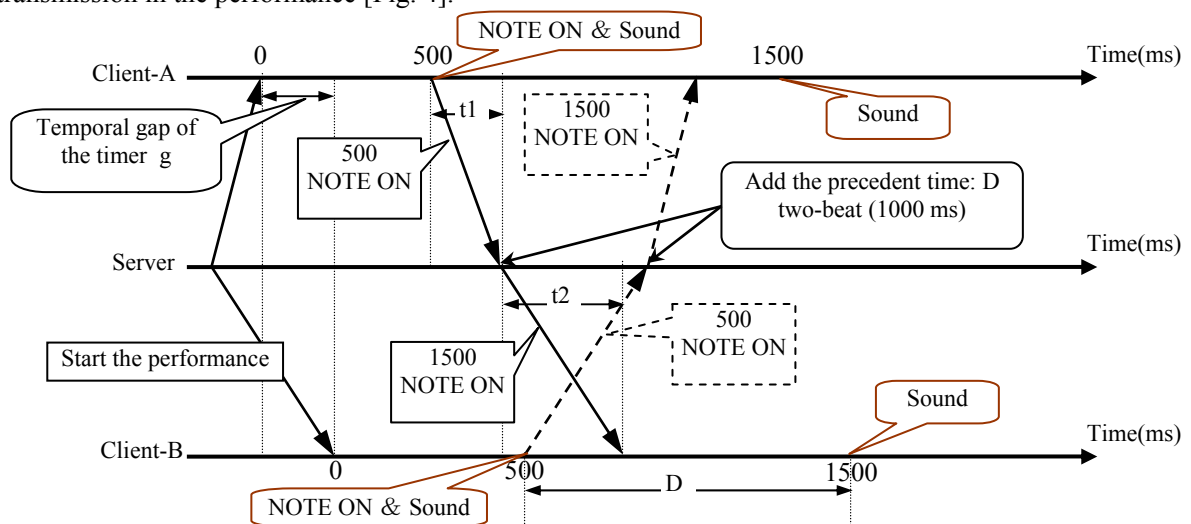


Fig. 5: Flow of the process

However, assuming the precedent time as D , the transmission time from the client-A to the server as t_1 , the transmission time from the server to the client-B as t_2 , and the temporal gap of the start time of the timer as g , the solution to the equation $D > t_1 + t_2 + g$ indicates that there may be no timing problems for the ensemble.

The flow of the process is as follows. First, the server sends a start message to each client, then each client starts the timer, and players begin the performance. The note played by client-A is sounded immediately at client-A, and at the same time, it is sent to the server with the value of client-A's timer (i.e. time-stamp). The server adds the precedent time to the time-stamp of the received data and sends the data to client-B. That is, the server specifies the timing of the sound. After receiving the data from the server, client-B waits for the specified timing of the sound, and then the note is sounded. Figure 5 shows an example of this process. In this example, the NOTE ON is performed (the sound is played) at 500 ms after the start time of the ensemble at client-A, and the precedent time is two-beat (1000 ms). This is the case with Client-B.

As mentioned previously, this system uses MIDI messages for each note, which consists of the NOTE ON and NOTE OFF message, and a message is sent by UDP which may be lost. To solve the problem of message loss, we establish a new message, that is, the EXTEND message. This message is sent to the client for each beat. Meanwhile, the notes keep sounding as the client receives the EXTEND message. If the client receives either the NOTE OFF message or if the client times out to receive the EXTEND message, the note ceases to sound.

5 Performance

We conducted experiments to evaluate the usability of the M.A.S. system and the suitability of precedent time.

5.1 Process and Environment

The environment is shown in Fig. 6. We simulated a remote ensemble that is performed via the simulated Internet, using NIST Net. The NIST Net is a software package produced by the U.S. National Institute of Standard Technology to simulate various network types [14], and we used the NIST Net to create a network that has various delays.

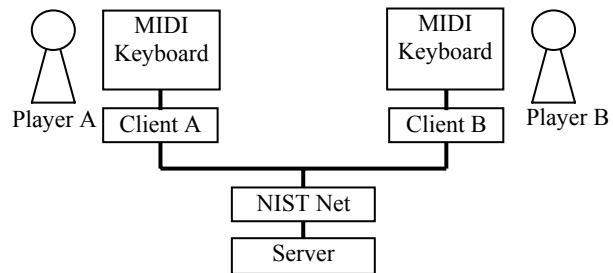


Fig. 6 The environment of the experiment

The duration of the Ping of the LAN inside our laboratory is less than 1 ms and the delay of the sound source is about 10 ms. Because these parameters are negligible for players, we need not consider them. We simulated two types of delay: a shorter delay whose average duration is 57 ms with a deviation of 8.1 ms, and a longer delay whose average length is 383 ms with a deviation of 93.5 ms.

We used 12 music pieces, whose lengths are from 8 to 16 measures. A pair of subjects played one music piece in three types of precedent time: case 1, no precedent time; case 2, 1000 ms (two-beat); case3, 2000 ms (four-beat). The four subjects (all experienced instrument players) were asked for their subjective estimation to evaluate the level of comfort for 10 levels and to freely give their impressions and opinions.

5.2 Discussions

We classified the experimental results into two types of delay.

Table 1 shows the results of the t-test between the precedent times. In Table 1, Y indicates that there is a significant difference and N indicates that there is not a significant difference. Those results show that when there is precedent time, players can perform more comfortably than when there is no precedent time.

Table 2 shows the result of the t-test between the two types of delay. There is a significant difference only in case 1. From this, in the presence of controlled delay, the length of the delay may have less influence on the players' performance.

Table 1: t-test between the precedent times

Delay	case 1 and case 2	case 1 and case 3	case 2 and case 3
Shorter	Y	Y	N
Longer	Y	Y	N

Table 2: t-test between the delays

Precedent time	Shorter delay and Longer delay
case1	Y
case2	N
case3	N

Moreover, we confirmed that the players could perform more comfortably when there are fewer dissonant chords [Fig. 7]. The correlation value between the percentage of dissonant chords and the level of comfort of each subjects are -0.627, -0.750, -0.555, -0.723 and the average of these correlation value is -0.664. This indicates that there is a relatively strong negative correlation between the percentage of dissonant chords and the level of playing comfort. From results gleaned from the questionnaire on subjective estimation, we also confirmed that the controlled delay provided the players with the ability to perform with a high level of comfort. This estimation of the level of comfort depends neither on the precedent time (2-beat or 4-beat) nor the delay length (shorter or longer).

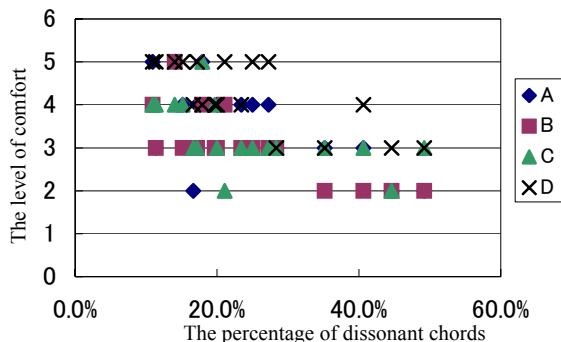


Fig. 7: Dissonant chords and the level of comfort

6 Conclusion

In this paper we proposed a new protocol for a musical session called Mutual Anticipated Session (M.A.S.) and carried out experiments to evaluate the validity of both the M.A.S. and precedent time.

From the results above, we could confirm the validity of the system that uses M.A.S. Using this system, many types of ensembles, e.g. chorus or concerts, may be performed via the Internet.

In the future, we should plan to develop experiments for novice or infant players. We also plan to investigate the suitable precedent time for each music piece. The study using more than three players should also be conducted.

References

- [1] Amar Chaudhary, Adrian Freed, Matthew Wright: An Open Architecture for Real-time Music Software, Proc. of the 2000 ICMC, ICMA (2000)
- [2] Masataka Goto, Ryo Neyama: Open RemoteGIG: An Open-to-the-public Distributed Session System Overcoming Network Latency, IPSJ Trans. Vol.43, No.2, pp.299-309 (2002.2) (in Japanese)
- [3] Yoichi Nagashima: Expanded Models of GDS (Global Delayed Session) Music, Proc. of FIT 2002 (2002) (in Japanese)
- [4] Masataka Goto, Yuji Hashimoto: A Distributed Cooperative System to play MIDI Instruments -Toward a Remote Session, IPSJ 93-MUS-4-1, Vol.93, No.109, pp1-8 (1993) (in Japanese)
- [5] Masataka Goto, Ryo Neyama: RMPC: Remote Media Control Protocol -Time Scheduling Extension and Remote Session with Delay, IPSJ 97-MUS-21-3, Vol.97, No.67, pp.13-20 (1997) (in Japanese)
- [6] Masataka Goto, Ryo Neyama, Yoichi Muraoka: RMCP: Remote Music Control Protocol -Design and Applications-, Proc. of the 1997 ICMC, ICMA, pp446-449 (1997)
- [7] Dominique Fober: Real-time Midi data flow on Ethernet and the software architecture of MidiShare, Proc. of the 1994 ICMC, ICMA, pp447-450 (1994)
- [8] Dominique Fober, Stephane Letz, Yann Orlarey: MudiShare joins the Open Source Softwares, Proc. of the 1999 ICMC, ICMA, pp311-313 (1999)
- [9] Matthew Wright, Adrian Freed: Open Sound Control -A New Protocol for Communicating with Sound Synthesizers-, Proc. of the 1997 ICMC, ICMA (1997)
- [10] Matthew Wright: Implementation and Performance Issues with Open Sound Control, Proc. of the 1998 ICMC, ICMA (1998)
- [11] Dominique Fober, Stephane Letz, Yann Orlarey: Clock Skew Compensation over a High Latency Network, Proc. of the 2002 ICMC, ICMA, pp548-552 (2002)
- [12] Dominique Fober, Yann Orlarey, Stephane Letz: Real Time Musical Events Streaming over Internet, Proc. of the International Conference WEB Delivering of Music IEEE2001, pp147-154, (2001)
- [13] Yoichi Nagashima: Delay measurement of MIDI synthesizers and study of its problem, <http://nagasm.suac.net/ASL/> (in Japanese)
- [14] NIST Net URL: <http://www.antd.nist.gov/tools/nistnet/>