Exploration of Possibility of Multithreaded Conversations Using a Voice Communication System

Kanayo Ogura[1], Kazushi Nishimoto[2], and Kozo Sugiyama[1]

[1] Japan Advanced Institute of Science and Technology, School of Knowledge Science, Asahidai 1-1, Nomi, Ishikawa, 923-1292, Japan
[2] Japan Advanced Institute of Science and Technology, Center for Knowledge Science, Asahidai 1-1, Nomi, Ishikawa, 923-1292, Japan
{k-ogura,knishi,kozo}@jaist.ac.jp

Abstract. Everyday voice conversations require people to obey the turn-taking rule and to keep to a single topic thread; therefore, it is not always an effective way to communicate. Hence, we propose "ChaTEL," a voice communication system for facilitating real-time multithreaded voice communications. ChaTEL has two functions to support multithreaded communications: a function to indicate to whom the user talks and a function to indicate which utterance the user responds to. Comparing ChaTEL with a baseline system that does not have these functions, we show that multithreaded conversations occur more frequently with ChaTEL. Moreover, we discuss why ChaTEL can facilitate multi-threaded conversations based on analyses of users' speaking and listening behaviors.

Keywords: CMC (Computer-Mediated Communication), Multithreaded Conversations.

1 Introduction

In everyday conversations, we usually have to share a single topic thread for a while and talk about something related to the current thread. Moreover, we have to preserve the turn-taking rule: only one person speaks at a time and the others are listeners. Therefore, it is impossible for people who do not have a “turn” to talk about what they want, even if they have an interesting topic or a good idea. As a result, regrettably, they often forget the topic or the idea. Such daily conversation is inefficient.

Since text-based chat systems overcome these restrictions, we can converse in multiple topic threads. Schegloff [1] explained the situation of "schism" related to multiple topic threads in face-to-face conversations. Schism is a situation of divisions of the conversational floor. For example, when four people take part in a conversation, one pair of people converse and the other pair of people converse in the same conversational floor. On the other hand, in text chat conversations, a person who takes part in a conversation can converse in several topic threads at the same time. We define multiple topic threads not as dividing people into several separated groups but as concurrently involving a person in several different topic threads; furthermore, we

J. Jacko (Ed.): Human-Computer Interaction, Part III, HCII 2007, LNCS 4552, pp. 186–195, 2007.

do not think a multiple topic thread is equal to schism in our study. Figure 1 shows examples of a situation of schism and multiple topic threads that we defined in our study.

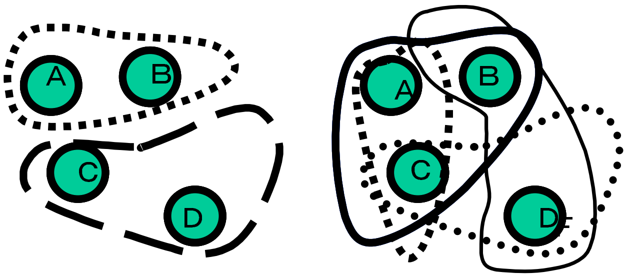

Fig. 1. Examples of schism (left) and multiple topic threads (right)

Based on analyses of text-based chat conversations, we propose a novel voice communication system named "ChaTEL" that allows multithreaded voice communication. The rest of this paper proceeds as follows. Section 2 considers the requirements for enabling us to converse in multiple topic threads with voice based on analysis of a situation of multiple topic threads in text chat conversations. Section 3 explains related works for multithreaded communication systems. Section 4 explains ChaTEL's system set-up. Section 5 describes our experiments to confirm its effects. Section 6 describes our experimental results. Section 7 concludes the paper.

2 Maintaining Multithreaded Conversations in Text-Based Chats

It is very difficult to distinguish simultaneous voice utterances, to memorize them, and to accurately respond to them. Due to such difficulties, we usually have to share a single topic thread. In text-based chats, however, multithreaded conversations are often maintained for a long time. This principally results from the "history of utterances" with which text chat systems are equipped. The history records users' names, messages, and the times messages were sent. They are listed on client application windows. Therefore, users can readily read any utterances and respond to them anytime [2].

In addition, skilled chat users use several special representations to specify receivers and related topics/messages to manage complicated multiple topic threads in text-based chats [3]. Based on analyses of Japanese chat conversations that contain 870 utterances, we found the following three representations for managing multithreads [4]:

1. ">name": specification of receiver(s)
ex.) Are you ready? > Mr. A
2. ">noun": specification of related topic(s)
ex.) I like it very much > chocolate
3. "copy": specification of related phrases in a previous utterance
ex.) I like it very much > chocolate > What a coincidence that you like it so much!

Table 1. Frequencies of usage of each representation

	> name	> noun	copy
Adjacent	**34**	**22**	**4**
Not adjacent	**80**	**58**	**16**

Table 1 shows the frequencies of usage of each representation. “Adjacent” means that the representation relates to the immediately previous utterance, and “Not adjacent” means that the representation relates to an utterance more than two utterances before. These three representations are used more in the “Not adjacent” cases than in the “Adjacent” ones. “Not adjacent” cases appear in multithread situations. These results show that text chat users manage complicated multiple topic threads with these representations to simplify following each thread.

3 Related Works

Few studies have aimed to facilitate multithreaded voice conversations. Berc et al. [5] developed “Pssst,” which provides whispering functions for some members in a teleconference using video links. In Pssst, only users of the whispering channel can join both the plenary meeting and the whispering channels. This is an incomplete multithread situation. The users who join only the plenary meeting channel are in the usual single-thread situation. Even those who join the whispering channel can join only two threads. Nishimoto et al. developed “VoiceCafe” [6], an asynchronous teleconference system using voice. Each voice utterance is automatically transcribed by a voice recognition system that analyses the question and answer relations of the utterances. Based on the relations, a tree structure of the relations of the utterances, such as an interface of a BBS system, is automatically constructed. Though this system enables users to converse in multiple topic situations, that is not its principal purpose; the researchers aimed to construct a proper tree structure. Aoki et al. [7] developed an audio space system that allows multiple simultaneous conversations in a conversation space. This system identifies conversation groups based on the turn-taking timing of utterances. Therefore, people in a group can talk in a usual single-thread manner; this system allows multiple single-threaded conversations to exist in a communication space.

4 Our Proposed New Voice Communication System: ChaTEL

We propose a communication system called “ChaTEL” that facilitates multithreaded voice conversations based on findings from text chat analysis. This system is equipped with a “history of conversation” as well as functions that specify receivers of messages and related messages. It is based on server/client architecture identical to typical text chat systems. Using a client, users record voice messages and upload them to the server. A user listens to a message by downloading it from the server.

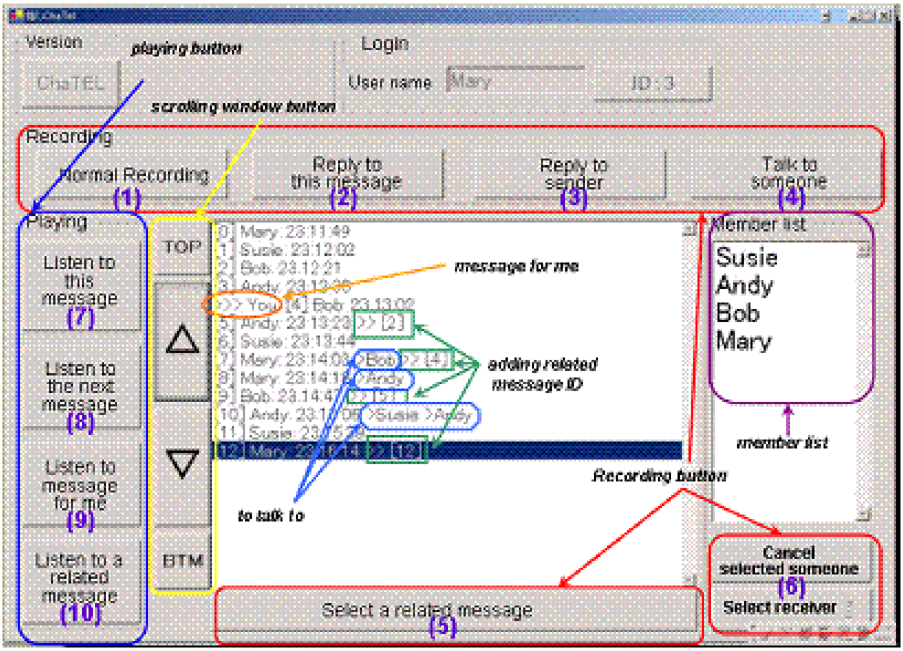

Fig. 2. ChaTEL user interface

Figure 2 shows ChaTEL's user interface. The user wears a headset to talk and to listen to messages. A keyboard is only necessary for inputting a user name for logging in to the system. After logging in, the user can operate all functions by using an LCD display with a touch panel. We prepared the following three ways to record a message:

1. Normal recording

The user can simply record a voice message by pressing the "normal recording" button (Figure 2-(1)). Then the "stop recording" button appears over the entire user interface. After the recording of the message is finished, the user pushes the stop recording button. The recorded voice message is immediately uploaded to the server, and the metadata of the message (ID number, speaker's name, and uploaded time) are dispatched to all clients. Metadata are added at the bottom of the history.

2. Recording by specifying a related message

While selecting a message from the history, the user can record a new message that relates to the selected message by pushing the "Reply to this message" button (Figure 2-(2)). The selected message's ID is added to its metadata recorded by this method, such as ">>[2]," which shows related message IDs.

3. Recording by specifying receiver(s)

After normal recording, the user can specify receivers by selecting a receiver from the member list and pushing the "Select receiver" button (Figure 2-(6)). In addition, after selecting a receiver from the member list, the user can push the "Talk to someone" button (Figure 2-(4)) and record a new message. Otherwise, while selecting a message from the history, the user can record a new message that specifies a receiver by pushing the “Reply to sender” button (Figure 2-(3)). The specified receiver names are added to the message’s metadata recorded by such methods as “>Susie>Andy.” If the user him/herself is specified as the receiver of a message, the description “>>>You” is added to the top of the message’s metadata.

We prepared the following four ways to listen to messages:

1. Listen to this message (Figure 2-(7))
   While selecting a message from the history, the user can listen to it by pushing this button.
2. Listen to the next message (Figure 2-(8))
   While selecting a message from the history, the user can listen to the next message (the message just after the selected message) by pushing this button.
3. Listen to a message for me (Figure 2-(9))
   If there are messages with the description “>>>You,” the user can listen only to them by pushing this button.
4. Listen to a related message (Figure 2-(10))

If the message is related to a previous message (description “>>[message ID]” is added at the end of the message’s metadata in the history), the user can listen to the related message by pushing this button.

5 User Studies

The aim of the experiments included: 1) to investigate whether multithreaded conversation can be done by voice; 2) to evaluate the usefulness and the effectiveness of the newly added ChaTEL recording/listening functions for facilitating multithreaded conversations; 3) to observe which newly added ChaTEL recording functions are more effective when multithreaded conversations are facilitated; 4) to understand how to record and listen to each message in multithreaded conversations using the newly added ChaTEL recording/listening functions.

We conducted experiments using ChaTEL in four groups that each consisted of four subjects. All subjects were accustomed to text chat, but they had never experienced a PC voice communication system. As a control system, we prepared a "baseline" system equipped only with the following recording/listening functions: "Normal recording," "Listen to this message," and "Listen to the next message." The experiments were conducted in non-face-to-face situations using either ChaTEL or the baseline. Two groups used the baseline system first and then ChaTEL, while the other groups used ChaTEL first then the baseline to eliminate the influence of order.

We prepared four initial topics: "Where do you want to travel?"; "Where are you from?"; "What food do you recommend to others?"; and "What do you want to get now?" We apportioned two of the four topics to each subject; the combinations of topics for the subjects differed. We instructed all users to converse for about 20 minutes using either the baseline system or ChaTEL. We asked them to begin talking about the provided topics and to discuss them completely. However, we allowed them to join other topics that were not assigned and to start talking about any new topics. We sampled the conversation histories, member lists, files of each message, and the log data of all users' actions that contained information on the time that each button was pushed. Based on these data, we analyzed each conversation's thread structures and each user's actions of recording and listening to messages.

6 Results

6.1 Possibility of Multithreaded Conversations Using Voice Communication System

We illustrated the structure of topic-thread relations for each experiment's data to compare ChaTEL and the baseline for degree of occurrence and duration of multithread situations. Figure 3 shows sample diagrams of the structures of topic-thread relations. Several dependent trees as well as branches are found in each tree. A tree corresponds to a topic thread. On the other hand, a branch corresponds to a question and answer pair. Based on Figure 3, we counted the average number of concurrent topic threads and branches per second. The results are shown in Table 2. Additionally, we calculated the duration of multithread situations for all concurrent threads. The results are shown in Table 3.

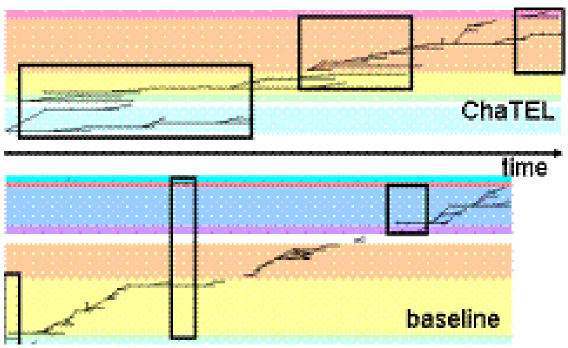

Fig. 3. Sample diagrams of structures of topic thread relations. Rectangles show places where multithreaded conversations occur.

Table 2. Average number of concurrent topic threads and branches per second for baseline and ChaTEL (Both results are significantly different.)

average numbers of concurrent topic threads		average numbers of branches	
baseline	ChaTEL	baseline	ChaTEL
1.20	**1.62**	**1.97**	**2.63**

From Table 2, we can see significantly more concurrent topic threads and branches in ChaTEL than in the baseline. In addition, Table 3 shows that multithreaded situations occur more often and are maintained longer with ChaTEL than with the baseline system in all concurrent threads.

Table 3. Duration of multithread situations for each concurrent thread (sec.)

Numbers of concurrent threads	2 threads	3 threads	4 threads
baseline	**965**	**241**	**0**
ChaTEL	**1418**	**880**	**181**

From these results, we conclude that it is basically possible to concurrently converse on multiple topics with voice communication. Furthermore, the fact that multithreaded situations occur more often and are maintained longer with ChaTEL shows that ChaTEL's newly added recording/listening functions contributed to facilitating multithreaded conversations.

In Section 1, we defined a situation of multithreaded conversations not as division of people who are in the same floor of conversation but as a situation having several threads per user. We examined the number of users in each thread based on conversational histories and diagrams of constructions of topic thread relations when multiple topic threads occur. The results are shown in Table 4.

Table 4. Average Number of conversations for each participant when multiple topic threads occur

Number of participants	2 participants	3 participants	4 participants
baseline	**11**	**8**	**0**
ChaTEL	**6**	**8**	**8**

From Table 4, we can see all users take part in eight multiple topic threads using ChaTEL. This result shows that the definition of the situation in multiple topic threads we explain in Section 1 is satisfied. From this result, we can see that situations of multiple topic threads do not occur and continue easily in the baseline and that situations of multiple topic threads occur and continue easily in ChaTEL. In addition, we conclude that all users in ChaTEL continue to converse in multiple topic threads even if they involve complicated constructions of conversations.

6.2 Usages of Listening/Recording Operations

To see complexity of listening/recording operations, we counted all conversational times and utterances in the baseline and ChaTEL. These results are in Table 5. Moreover, we calculated all times of conversations and all utterances in the baseline and ChaTEL to see frequency of each listening/recording operation. These results are in Tables 6 and 7.

Table 5. all conversation times and utterances in baseline and ChaTEL

	baseline	ChaTEL
All times of conversations (sec)	**5528**	**5689**
All utterances	**339**	**337**

Table 6. Frequency and Ratio of recording operation in ChaTEL

	Frequency	Ratio
Normal recording	**47**	**14.0%**
Recording by specifying a related message	**265**	**78.6%**
Recording by specifying receiver(s)	**25**	**7.4%**
Total	337	100%

From Table 5, we can see there is not much difference between the baseline and ChaTEL in the number of utterances. This result shows that the complexity of listening/recording operations in ChaTEL does not have an influence on the number of utterances and usability of the listening/recording operations.

From Table 6, we can see that users in ChaTEL often use the operation of recording by specifying a related message—nearly 80% of the time. On the other hand, we can see that users use the operation of recording by specifying receiver(s) only 7.4% of the time. In Table 1, users more frequently use the operation of specifying receiver(s) than the operation of specifying a related message in text chat conversations. This result in text chat conversations is in contrast to the results in voice communication systems using baseline and ChaTEL.

Table 7. Frequency and Ratio of listening operations in baseline and ChaTEL

	baseline		ChaTEL	
	frequency	ratio	frequency	ratio
Listen to this message	**595**	**42.6%**	**1002**	**57.0%**
Listen to the next message	**801**	**57.4%**	**737**	**42.0%**
Listen to a message for me	**--**	**--**	**1**	**0.1%**
Listen to a related message	**--**	**--**	**16**	**0.9%**
Total	1396	100%	1756	100%

From Table 7, we can see that users in the baseline more frequently use the operation of listening to the next message than the operation of listening to this message. On the other hand, we can see that users in ChaTEL more frequently use the operation of listening to this message than the operation of listening to the next message. In addition, we can see that the operation of listening to a message for me and listening to a related message, which are prepared for ChaTEL, have a combined use ratio of only 1%.

6.3 Analysis of How Users Listen to Messages in Multithreaded Conversations

In section 6.2, we saw that users in ChaTEL record messages using operations of recording by specifying a message and receiver(s) at a rate of about 90%. Based on this result, we can expect that users in ChaTEL select operations of listening related to these recording operations. On the other hand, we can expect that users in the baseline listen to messages in order of the history of conversations and that they tend to listen to the newest message users record because they can understand the contents of each message after they listen to a message.

Based on data of actions of listening operations, we examined the distance between a newly listened to message and a previously listened to message (a distance of listening operations) and the distance between a newly listened to message and a newly recorded message (a distance of delay). This result is shown in Table 8.

We explain how to analyze the distances of listening operations and of delay. In distance of listening operations, we consider an operation of listening to be the next message users listen to after the previous message listened to as the standard pattern of listening operations, and we add the score of each listening operation. We use absolute values to descript each point of distance. The following are patterns of listening operations and points. In addition, Table 9 shows the frequencies of patterns of listening operations.

1) NEXT: listening to next message: point = 0
2) REPEAT: listening to the same message: point = -1
3) BACK: listening to n (n>0) utterances forward message: point = -n-1
4) SKIP: listening to n (n>1) utterances next message: point = n-1

In a distance of delay, we calculated the difference between a listened to message ID and a newly recorded message ID as the distance of delay.

Table 8. Avarages of distance of listening and delay (*** shows significant 1% standard)

	baseline	ChaTEL	
Distance of listening	0.49	1.63	***
Distance of delay	1.01	2.53	***

From Table 8, in the baseline, we can see that a distance of listening is near point 0. This result shows that users in the baseline tend to listen to messages in order of the history of conversations. On the other hand, in ChaTEL, we can see that the distance of delay is twice that of the baseline. This result shows that users in ChaTEL can listen to messages besides the newly recorded ones.

Table 9. Averages of frequencies of patterns of listening (*** shows significant 1% standard)

	baseline	ChaTEL	
NEXT	**69.1**	**74.1**	
REPEAT	**4.6**	**6.7**	
BACK	**5.3**	**19.6**	***
SKIP	**9.7**	**16.1**	

From Table 9, we can see that users in ChaTEL tend to listen to past messages distant from the newest recorded message. From Table 8 and Table 9, we can conclude that users in the baseline tend to listen to messages in order of the history of conversations and that users in ChaTEL tend to listen to messages distant from the newest message.

7 Conclusion

In this paper, we proposed a novel communication system named "ChaTEL" to achieve multithreaded voice communication. We equipped ChaTEL with a "history of conversation" as well as functions to specify receivers of messages and related messages; it was based on the findings of our study on text chats where users often converse in a multithreaded manner. Based on user studies we confirmed that ChaTEL facilitates multithreaded voice communication. In addition, users in ChaTEL can listen to messages distant from the newest message without being restricted to the history of conversations. In the future, we plan to improve the system to be used in mobile situations using these results in this study.

Acknowledgement. This research is partly supported by the fund from Kinki Mobile Radio Center Inc., Japan.

References

[1] Schegloff, E.A.: Issues of Relevance for Discourse Analysis: Contingency in Action, Interaction and Co-Participant Context. In: Hovy, E.H., Scott, D. (eds.) Computational and Conversational Discourse, pp. 3–38. Springer, Heidelberg (1996)

[2] Rhyll, V.: Applying membership categorization analysis to chat-room talk, How to Analyse Talk in Institutional Settings: A Casebook of Methods. In: McHoul, A., Rapley, M. (eds.) Continuum, London, pp. 86–99 (2001)

[3] Winiecki, D.J.: Instructional discussion in online education, Practical and research-oriented perspectives. In: Moore, M., Anderson, R.M. (eds.) Handbook of Distance Education, Lawrence Erlbaum Assoc., NJ (2003)

[4] Ogura, K., Ishizaki, M.: The Characteristics of Topic Flow in Chat Conversations, Tech. Report SIG-SLUD, A202-3, Japan Society of Artificial Intelligence, pp. 13–19 (2002)

[5] Berc, L., Gajewska, H., Manasse, M.: Pssst: Side Conversations in the Argo Telecollaboration System. In: Proc. the 8th ACM UIST, pp. 155–156 (1995)

[6] Nishimoto, T., Kitawaki, H., Takagi, H.: VoiceCafe: An Asynchronous Voice Meeting System, Information Technology Letters, LK-005, pp. 273–274 (2003)

[7] Aoki, P.M., Romaine, M., Szymanski, M.H., Thornton, J.D., Wilson, D., Woodruff, A.: The Mad Hatter's Cocktail Party: A Social Mobile Audio Space Supporting Multiple Conversations. In: Proc. ACM CHI 2003, pp. 425–432 (2003)