

Title	インターネットチャットシステムにおけるタイミング情報の共有に関する研究
Author(s)	益田, 武士
Citation	
Issue Date	2003-03
Type	Thesis or Dissertation
Text version	author
URL	<a href="http://hdl.handle.net/10119/459">http://hdl.handle.net/10119/459</a>
Rights	
Description	Supervisor:石崎 雅人, 知識科学研究科, 修士

修 士 論 文

インターネットチャットシステムにおける  
タイミング情報の共有に関する研究

指導教官 石崎 雅人 助教授

北陸先端科学技術大学院大学  
知識科学研究科知識システム基礎学専攻

150070 益田 武士

審査委員： 石崎 雅人 助教授（主査）

**Ho To Bao** 教授

下嶋 篤 助教授

林 幸雄 助教授

2003 年 2 月

# 目次

1	はじめに	1
1.1	研究の背景	1
1.1.1	チャットとは.	1
1.1.2	チャットの特徴と問題点	2
1.2	研究の目的.	6
1.3	本論文の構成	6
2	関連研究	7
2.1	<b>Alternative Interfaces for Chat</b>	7
2.1.1	<b>Status Client.</b>	7
2.1.2	<b>Flow Client</b>	8
2.2	<b>Threaded Chats.</b>	11
2.3	<b>Tangible Chat</b>	13
2.4	<b>Chat Circles.</b>	15
2.5	本研究との関連	18
3	チャットシステムの構築	20
3.1	入力状態を波形で表示するチャットシステム.	20
3.1.1	波形表示による打鍵情報の可視化.	20
3.1.2	システムの構築.	21
3.1.3	考察	22
3.2	入力情報, タイミング情報を共有できるチャットシステム	24
3.2.1	概要	24
3.2.2	特徴	26
4	評価実験	29
4.1	概要.	29

4. 2	定量的データの分析	3 1
4. 3	定性的データの分析	3 6
4. 4	発言履歴から見るシステムの影響	4 3
4.4.1	発言の擬似的重複	4 3
4.4.2	履歴上での発言順序の入れ替わり	4 4
	まとめと今後の課題	<b>4 6</b>
5. 1	まとめ	4 6
5. 2	今後の課題	4 6
	謝辞	4 7
	参考文献	
	発表論文	
	付録 (最終版のシステム)	
1.	プログラム概要	
2.	アルゴリズム	
3.	プログラムリスト	
4.	システムの利用説明	

# 目 次

1. 1	擬似的な重複の例	3
1. 2	ニックネームと発言の組	5
2. 1	<b>Status Client</b>	8
2. 2	<b>Flow Client</b>	10
2. 3	<b>Treaded Chats</b> の画面	11
2. 4	<b>Chat Circles</b> : メイン画面	16
2. 5	<b>Chat Circles</b> : 発言履歴画面	17
3. 1	スタートウィンドウ	23
3. 2	メインウィンドウと波形ウィンドウ	23
3. 3	状態遷移図	25
3. 4	メインウィンドウ	28
3. 5	サブウィンドウ	28
4. 1	相関図	32

# 目 次

1. 1	特定の人に対しての指し示す例	4
1. 2	特定の内容に対しての指し示し例	4
1. 3	関連が不明の例	4
2. 1	感情の伝達について	15
2. 2	総合アンケート	15
4. 1	評価システム	29
4. 2	要因と水準	30
4. 3	発言数と発言の長さの相関係数	32
4. 4	発言数の平均値と標準偏差値	33
4. 5	発言の長さの平均値と標準偏差値	33
4. 6	発言数における分散分析結果	34
4. 7	発言の長さにおける平均の分散分析結果	34
4. 8	発言数における水準ごとの分散分析結果	35
4. 9	アンケート項目	36
4. 10	項目間の相関係数	37
4. 11	アンケート結果	38
4. 12	質問項目1の結果	39
4. 13	項目1の分散分析結果	39
4. 14	質問項目2の結果	39
4. 15	項目2の分散分析結果	39
4. 16	質問項目5の結果	40
4. 17	項目5の分散分析結果	40
4. 18	質問項目6の結果	40
4. 19	項目6の分散分析結果	40

4. 2 0	質問項目 3 の検定結果.	. . . . .	4 1
4. 2 1	質問項目 4 の検定結果.	. . . . .	4 2
4. 2 2	擬似的重複の割合.	. . . . .	4 3
4. 2 3	表示順序の入れ替わり例	. . . . .	4 4
4. 2 4	発言開始と発言完了の入れ替わりの割合.	. . . . .	4 4

# 第 1 章

## はじめに

### 1.1 研究の背景

#### 1.1.1 チャットとは

近年のインターネットの普及に伴い、インターネットを利用したコミュニケーションツールが頻繁に利用されるようになってきている。こういったコミュニケーションツールには電子メール、電子掲示板、チャットなどがある。電子メールや電子掲示板が非同期のコミュニケーションを行うツールであるのに対して、チャットは、多人数でリアルタイムに会話可能なツールとして利用が盛んである。チャットは利用するシステムの体系から大きく 2 つに分類される。

- ・ **Web** チャットシステム：CGI や JAVA などを用いて **web** ブラウザ上で会話を行うシステム
- ・ メッセンジャー：専用プログラムを用いて、接続されている者同士で会話を行うシステム

**Web** チャットは基本的に面識のない者同士でコミュニケーションを行うものであるのに対し、メッセンジャーツールは知人・友人など、ある特定の人とのコミュニケーションを行う場合に利用されることが多い。最近では代表的なメッセンジャーツールとして、**MSN** メッセンジャー[3]や **i** チャットなどといったものが広く利用されている。**Web** チャット、メッセンジャー共に、コンピューターを利用してリアルタイム



に会話するコミュニケーションツールであるとして、本研究では両者ともチャットシステムと呼ぶ。

チャットシステムは広く利用される一方、タイミング情報の欠乏などの、チャット特有の問題により、会話が円滑に進行しないことが指摘されている[1][2]。

## 1.1.2 チャットの特徴と問題点

対面対話では、同じ空間、同じ時間を共有しており、言語情報と共に、身振り手振りや顔の表情などといった様々な非言語情報を共有し合うことにより、上手くコミュニケーションを行っている。チャットは、身振り手振りや顔の表情といった非言語情報は共有されないが、感情を表現する顔文字や、顔の向きを示す記号により部分的であるが、発言の雰囲気や伝えている。しかしそういった部分的なものでは、対面対話で伝えられる様々な非言語情報を十分には伝えられない。そのため、チャットでは以下のような問題が生じる可能性がある。本節ではチャットシステムの問題点を挙げる。

### (1) 発言過程の情報の欠乏による問題

一般的なチャットシステムでは何か発言したい場合、発言を入力し、最後にリターンキーを押して発言を送信する。発言は送信されて始めて各参加者の画面に表示されるため、発言を送信するまで、各参加者には発言を入力している状態が伝わらない。つまり、これは発言は全て完了形で表現され、発言過程の文字情報とタイミング情報が欠乏していることを意味する。

このような情報の欠乏により、擬似的な重複（同時発話）や割り込みが起こり会話が円滑に進まないことが指摘されている[1][2]。擬似的な重複の例を図 1. 1 に示す。A の「明日は企業の見学会へ行ってきました。」に対して、次に B が応答をしている。しかし A は B が打ち込んでいることがわからないため、A は次の発言を入力し始め、B の応答と A の発言が同時に送信されている。その結果、隣接ターン間で意味的な対応がとれなくなっている。

この問題に対応しているシステムには以下のようなものがある。MSN メッセージャー[3]では参加している者の誰かが発言を入力し始めると各参加者のウィンドウ下部のステータスバーに「○○○が入力しています」と表示される。Tangible Chat[4]ではキーを打つ度に振動が相手に伝わり発言入力中であることを示す。また UNIX の

**talk** コマンドでは相手が発言を入力する度に、画面にその時の相手の入力した文字が表示される。

1: A > 明日は企業の見学会へ行ってきました。(114)  
2: B > おおっ, 大阪ですか(30)  
3: A > そうそう, T社の選考会は怎么样了?(0)  
4: A > うん. 大阪です.(14)  
5: B > なんか資料を提出しなきゃいけないらしくて(59)

( )内は前の発言から要した時間(秒)

図 1. 1 擬似的な重複の例

## (2) 関連情報の欠乏

チャットでは、特定の参加者に対して応答する場合、発言の最後に">○○さん"と入力し、相手を指し示す場合がある（表 1. 1）。この">"は対面対話での発言者が話したい人へ顔を向けて話すことを図示している。また直接特定の人を指し示さず、話題の関するキーワードを同様に発言の最後に">○○○"と入力し、どの内容に対しての応答なのかを指し示す（表 1. 2）。もし、こういった指し示しがない場合、誰のどの発言に対しての応答なのか曖昧性が残り、判断不可能な場合がある（表 1. 3）。例えば表 1. 3 では C の「嫌ですね」が「PC につないでたのが凹んだ事柄」に対しての返答なのか、「雪が積もる事柄」に対しての返答なのか不明である。

表 1. 1 特定の人に対しての指し示し例

発言者	完了時間	発言内容
B	13:13:00	そういや、最近片町にも行ってないなあ
A	13:13:16	C 言語の C は A 言語 B 言語
C	13:13:39	視力が弱まるから、よくいろんなものを見てこなきゃ!!> B
B	13:13:51	Computer の C じゃないんだね > A
C	13:14:01	初めて聞いた!> A



表 1. 2 特定の内容に対しての指し示し例

発言者	完了時間	発言内容
A	19:42:09	最近は近いのでバザールにってるかな
C	19:42:32	バザールも地味だけどけっこう品揃えいいですね~。
C	19:43:02	プラントは安いけど・・・高いものは高いからむかついていってません(笑)
A	19:43:49	まあでも安いものは圧倒的に安いよね > プラント

表 1. 3 関連が不明の例



発言者	完了時間	発言内容
B	14:45:19	メイン PC につないでいたので凹みました。
B	14:45:39	しかし明日は積もるんですかね。
C	14:45:47	嫌ですね。



### (3) 相手の存在情報の欠乏による問題

チャット対話ではコンピュータを介して対話を行うため、相手の状態を示す情報が欠乏している。例えばある参加者の発言がしばらく無い場合に、

- ・ 画面に表示されている発言を読んでいる
- ・ 発言を打ち込んでいる
- ・ 席をはずして、コンピュータの前に存在しない

の複数の原因が考えられる。

### (4) 履歴の不完全性による問題

チャットはコンピュータを利用したコミュニケーションツールであるので、会話履歴を電子データとして保存可能である。しかし、データとして保存されるのは会話内容のテキストデータであることがほとんどである。そのため、履歴からは発言時間を含むタイミング情報は失われ、上述のように重複や割り込みが生じたデータは、どれが同時発言なのか判断不可能となり、理解するのが難しいものとなる。

### (5) 参加者と発言との対応情報の欠乏

チャット対話は全てテキストで行われるので、誰がどの発言を話しているのか分かりづらい。例えば対面対話では、顔の向きや声のイントネーションによって、誰の発言であるかは容易に判断可能である。一般的なチャットでは、ニックネーム（参加者の仮の名前）を発言の前に置き、誰が発言しているかを示す（図 1. 2）。更に参加者ごとに色分けすることにより、この分かりづらさを解消する機能を実装しているシステムもある[3]。

```
take>こんにちは.  
miho>こんにちは take さん
```

図 1. 2 ニックネームと発言の組

## 1.2 研究の目的

本研究の最終目標は対話を円滑に進行させることが可能なシステムの構築である。特に上述の問題の1つである、チャットにおけるタイミング情報の欠乏による対話がうまく進行しない問題を解決するチャットシステムを構築することである。

具体的には、参加者の打鍵情報がリアルタイムに共有される機能、発言表示順序を発言開始時間に基づいた表示順序とする機能をもったシステムを構築した。

## 1.3 本論文の構成

第2章では本研究と関連する研究をまとめ、本研究の位置づけを明確にする。

第3章では、構築したシステムの概要、構築方法、特徴、操作方法について述べる。

第4章では、システムの評価実験について、その方法述べ、結果について分析・考察する。

第5章では、本研究のまとめと今後の課題について述べる。

# 第 2 章

## 関連研究

本章では 1 章で挙げたチャットにおける問題点を解決することを目的とした、チャットシステムを紹介し、最後に本システムとの関連を述べる。

### 2.1 Alternative Interfaces for Chat

**Alternative Interfaces for Chat** の目的は、チャットにおけるタイミング情報の欠乏問題を解決するシステムの構築である [4]。

手法としては、発言過程の発言内容とその間の時間情報を共有する手法をとっている。2 種類のプロトタイプシステム **Status Client**, **Flow Client** を構築しており、前者は参加者の状態情報を共有する機能を、後者は発言の開始時間から完了時間までの時間情報を共有する機能を持つ。

#### 2.1.1 Status Client

**Status Client** の画面を図 2. 1 に示す。 **Status Client** では 3 つのフォームがあり、上から発言履歴の表示欄、参加者の状態情報を表示するユーザーリスト欄、最後の 1 行フォームが発言記入欄である。

**Status Client** では参加者の状態情報を表示する以下の機能を実装した。

- ・ 発言中は、発言履歴表示欄、ユーザーリスト欄共に、ユーザー名が強調表示され、5 秒後に元に戻る
- ・ 各ユーザーの最後の発言はユーザーリストの各ユーザー名の横に表示される
- ・ 発言中の発言内容は、発言者がキーを打つ度にユーザーリストの発言者の

横に随時表示され、この発言内容は灰色で表示される（完了した発言は黒）

- ユーザーリスト欄の発言は **10** 秒かけて徐々に消えていく

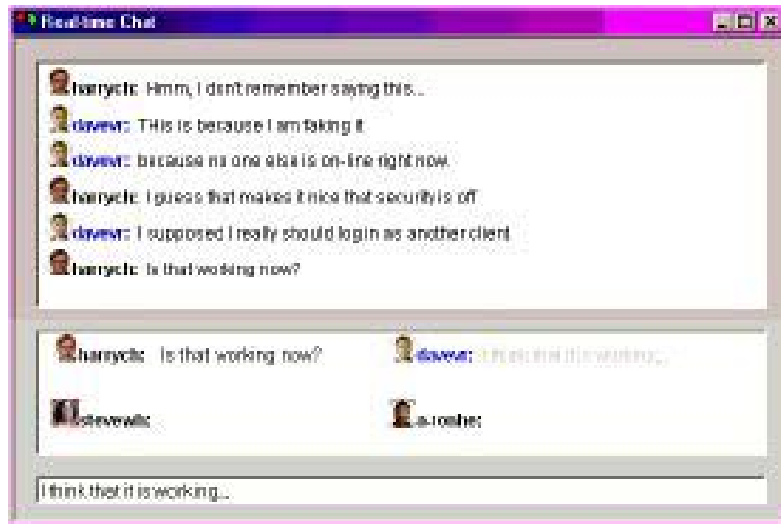


図 2. 1 Status Client[4]

評価実験では、5 人による対話を 2 対話利用している。評価対象は一般的なテキストベースのチャットと **Status Client** である。

分析には、アンケート結果と、発言履歴から全発言数、修正された発言数を用いた。アンケート調査から、**Status Client** の方が好まれたと示している。しかし、発言履歴からは、大きな違いは得られなかったとしている。

## 2.1.2 Flow Client

図 2. 2 に **Flow Client** の画面を示す。**Flow Client** は **Status Client** での評価を踏まえ、改良されたシステムである。特に発言の順番の保護や、タイピング速度の遅い人でも不自由なく会話の流れについていけるシステムの構築を目標とした。システムは予備実験の結果を踏まえて、最終的なシステムが完成するまで 2 回のユーザインタフェースの改良が行われた。システムの特徴は以下の 4 点である。

- (1) ユーザー毎の発言欄の確保
- (2) 発言過程の可視化

(3) 履歴の移動

(4) **snap-back-scrolling** 機能

### (1) ユーザー毎の発言欄の確保

**Flow Client** では、1 ユーザーに 1 つの発言表示列を与えられ、発言はその列内に表示されていく。図 2. 2 では、参加者は 2 名いるので、2 列表示されている。これにより同時に複数のユーザーが発言を開始しても、発言は重なることはなく、各ユーザーは常に発言可能な状態にある。また、ユーザーの発言表示列の右側には、現在の入力状態が表示される欄がある。

### (2) 発言過程の可視化

**Flow Client** では発言履歴は全て、発言表示列内のテキストボックス内に表示され、テキストボックスの色は発言者の状態を示している。発言入力があると、発言者の発言表示列に黄色の四角形のテキストボックスが現れる。発言内容はこの時はまだテキストボックス内には表示されず、ウィンドウ右端の空白欄にキー入力がある度に反映される。キー入力を完了しないまま途中で入力をやめると、テキストボックスは一時的に青色へ変化し、テキストボックス内に現在の入力文字が表示される。発言を完了すると、テキストボックスは赤色で表示され、発言内容がテキストボックス内に反映される。

### (3) 履歴の移動

**Flow Client** では、履歴は常に動いている。横軸に時間軸を取り、右ほど新しい発言となる。発言は時間と共に左へと移動し、発言表示列の左端まで移動すると、画面からは隠れてしまう。画面に表示されていない過去の発言を見る場合は、スクロールバーを利用することにより閲覧可能である。

発言表示列の下側には、現在を起点とした時間の目盛りが付与してある。図 2. 2 では、時間は発言表示列の一番右側を現在時間とし、そこから左へ「-5secs」、 「-10secs」と 5 秒きざみで表示される。



#### (4) snap-back-scrolling 機能

画面に表示されていない過去の発言を閲覧するには、スクロールバーをマウスでドラッグして利用する。その時に、マウスのボタンを離すと、瞬時に現在の時間軸の場所まで画面が戻る。



図 2. 2 Flow Client[4]

#### 評価実験

評価実験は前回の実験と同様、5人による対話を利用している。比較対象は一般的なテキストベースのチャットと **Flow Client** である。

タイピングのスピードや、精度に関して、**Flow Client** は優位であると期待したが、全発言数や各ユーザー事の発言数、発言の長さの平均は両システム共に類似した結果となり、差は得られなかった。

また履歴閲覧時に瞬時に戻る機能である "**snap-back-scrolling**" は、履歴活用の利便性を高めるため、ユーザーが素早く有用な履歴を見つけられる機能として著者は期待した。本機能を使用することにより履歴活用が有効に行われたかを、繰り返し発言数で比較した。両システムを比較すると、繰り返し発言の数は全発言数に対して、一般的なチャットでは 5% であるのに対して、**Flow Client** では 1% 以下までに下がっていた。しかしアンケート調査では **Flow Client** のような履歴表示方法は好まれなく、修正発言数の減少は、履歴の利用機会が増えたのではない事がこのことよりわかった。被験者は **Flow Client** のユーザーインターフェースはとても慣れにくいと評価し、垂直のスクロール形式を好んでいた。また、時間と共に自動的に表示が移動する事に関しても、長くなった発言に正確に対応できない場合があると述べている。

## 2.2 Threaded Chats

Threaded Chats の目的は、チャット対話での非言語情報の欠乏による問題を解決するシステムの構築である[5].

手法としては図 2. 3 のように、会話構造を木構造で表現し、ある発言に対しての応答は木の枝のように表現される。(windowsOS のファイルエクスプローラーや、電子掲示板に類似している.)

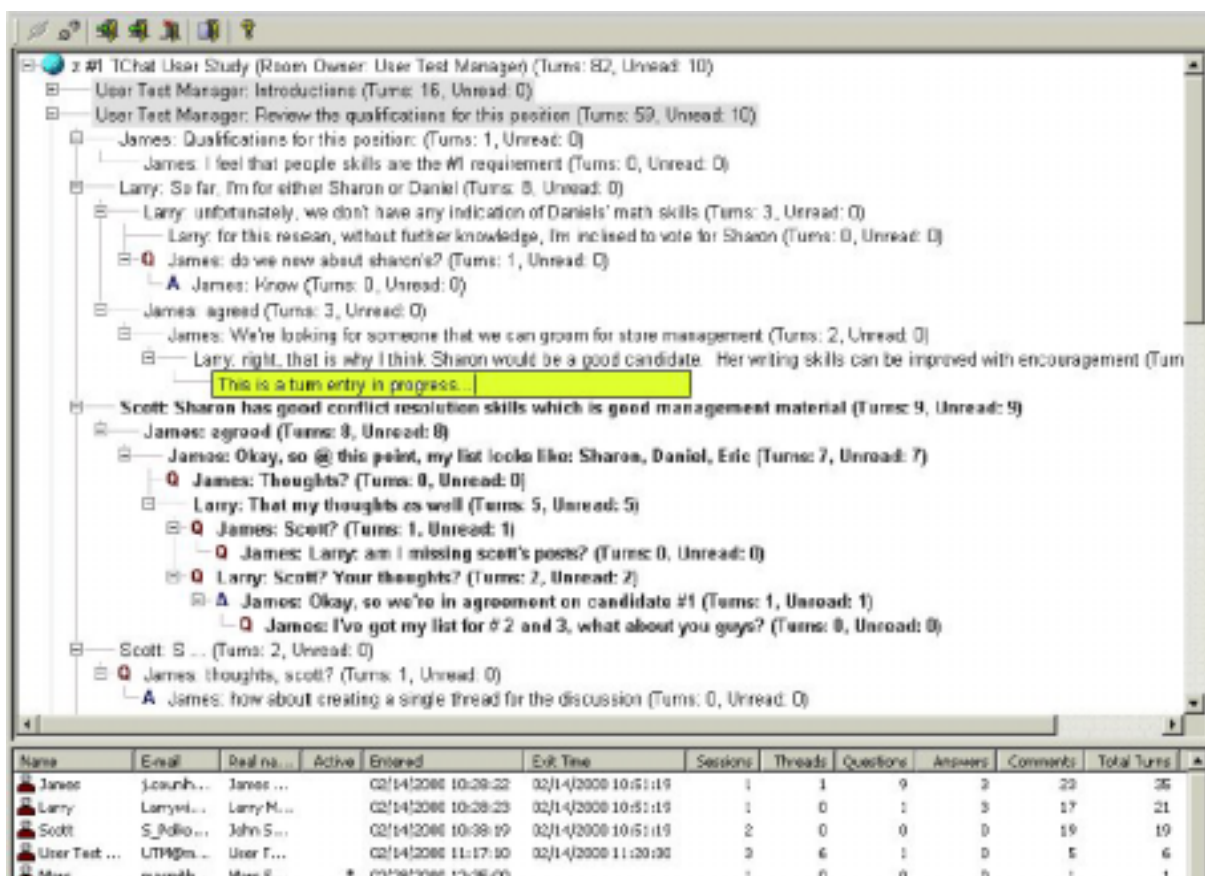


図 2. 3 Threaded Chats の画面[5]

システムの主な特徴は以下の 3 点である。

- (1) 発言間の関連を容易に把握可能
- (2) 発言を送信後でも修正・編集・移動可能
- (3) 発言の編集権限の管理

### (1) 発言間の関連を容易に把握可能

会話内容を木構造にすることにより、参加者はどの発言とどの発言とが関連しているかを視覚的に容易に把握できる。発言を置く場所を発言者が任意に決定可能なので、参加者の意図どおりの構造をした履歴表示となる。また表示順序が従来のチャットシステムとは異なり時間順に並んでいないので、最新の発言は太字で表示され、新規の発言であることを強調できる。更に補助機能として、発言内容に質問記号"?"が有る場合は、自動的に質問とみなし、発言タイトルの左側に"Q"と付けられ、その質問に対する返答には"A"と付けられる。

### (2) 発言を送信後でも修正・編集・移動可能

従来のチャットシステムでは発言を入力し、送信後は編集不可能である。しかし、**Threaded Chats** では 1 度発言を送信した後も自由に修正・編集可能である。また発言の配置場所の再移動も可能である。

### (3) 発言の編集権限の管理

発言の編集機能は履歴の話題の一貫性を高めるのに効果的だが、誰もが自由に編集可能であれば不正利用の恐れがある。**Threaded Chats** では UNIX のパーミッション属性のような権限の仕組みを各発言に対して付加する機能を実装している。

## 評価実験

評価実験は、11 グループは 3 人対話で収録し、後の 7 グループは 4 人対話を収録した。比較対象は本システムと一般的なチャットの 2 つである。対話内容は雇用者の決定に関する話題であり、課題遂行型の対話であった。以下の 5 つの観点を踏まえて分析を行っている。

- **Treaded Chats** は、従来のチャットよりも話題の一貫性があるか
- **Treaded Chats** の履歴は、従来のチャットよりも読みやすいか
- **Treaded Chats** は、従来のチャットよりも発言は長い（急ぐ必要はないため）
- **Treaded Chats** では、過去の発言を修正するための発言は減少したか（完了した発言でも修正・編集可能なため）
- **Treaded Chats** では、発言を途中で中止するような場合が減少したか

分析には「楽しめたか」、「混乱はなかったか」、「満足いく意思決定はできたか」などのアンケート調査による調査結果と会話履歴を用いた。

アンケートによる評価では **Treaded Chats** は一般的なチャットよりも極めて低い評価となった。理由としては、今回の被験者達はチャットをインフォーマルなコミュニケーションをするためのツールとして利用しており、今回のような課題遂行型の対話には慣れていないからだと考察している。

**Threaded Chats** では発言は長くなり、必然的に発言数は減少することを予想していた。発言数は **Threaded Chats** の方が一般的なチャットよりも少ない結果となったが、発言の長さの平均は両システムともほとんど違いはなかった。これにより **Threaded Chats** は、従来のチャットよりも少数の発言で課題を遂行可能であると主張している。

また参加者のバランス（各参加者が隔たり無く会話に参加できたか）を評価するため、各参加者の発言数の標準偏差を利用した。**Treaded Chats** は一般的なチャットよりも発言数の標準偏差は有意に低い結果となった。これにより **Treaded Chats** ではより参加者間のバランスが取れていたとしている。

## 2.3 Tangible Chat

**Tangible Chat** は、チャットで必ず利用し、かつ最も自然な行為である打鍵に着目し、打鍵行為によって生じる物理的作用である振動を相手に伝達するシステムである [6].

**Tangible Chat** の主な機能は以下の 2 点である。

1. 対話状況を非意図的に入力，伝達する機能
2. 非言語情報をそのまま提示する機能

打鍵により発生するキーボードの振動をセンサによって自動的に抽出・伝達する。これにより，利用者は特に意識することなく振動が伝達されるので，より自然な対話を行うことが可能であると述べている。

キーボードの振動は相手の椅子に敷いてあるクッションにそのまま振動として伝わる。これにより，対話状況認識の余分な認知負荷を必要としないと述べている。

## 評価実験

**Tangible Chat** では **2** つの実験を行っている。実験 **1** では振動有り・無しと課題内容での違いを検証するための実験である。実験 **2** では，比較対象として **MSN** メッセンジャーを用いた。これは打鍵情報の可視化手法を振動による提示と文字による提示との差を検証するためである。

### ・ 実験 1

評価実験では **2** 人対話 **14** 対話のデータを利用した。比較対象は一般的なテキストベースのチャットと，**Tangible Chat** の **2** つのシステムである。チャット内容は意思決定課題と対立課題の **2** 通りで，それぞれ振動有り・無しで行った。（各被験者は計 **4** 回のチャットを行う）

分析データには，各課題に対して協調できたか・合意できたかなどを調査するアンケートと，振動の有り・無しでの精神的負担・発言のしやすさ・発言タイミングのとりやすさ等についての総合アンケートを用いた。

結果の一部を表 **2. 1**，**2. 2** に示す。意思決定課題においては「あなたは感情をどのくらい出していたか」という質問に対し，振動有りの場合に被験者自身が楽しさを感じていることが確認できた。対立課題においては十分な有意差は得られないながらも，振動無しの方がやや高い値となった。また振動有りの場合の対立課題は，決裂しやすい傾向が見られた。これを裏付けるものとして，議論中に相手は譲歩しなかったと感じたというアンケート結果が出ている。また発言のしやすさ・発言タイミングの取りやすさについて，振動有りの方が有効であることが検証された。

表 2. 1 感情の伝達について[6]

	質問項目	振動あり	振動なし	t 値
		平均	平均	
意思決定課題	あなたは感情をどのくらい出していたか (楽しい)	4.3	4.1	2.00**
対立課題	相手の感情をどのくらい感じたか (楽しい)	3.6	3.9	1.55*
	議論の中で、相手は譲歩したと思うか	2.8	3.3	2.10**
	議論は合意に達したか、それとも決裂か	2.5	3.1	1.89*

\*は 10%で、\*\*は 5%の有意水準片側 t 検定で有意

表 2. 2 総合アンケート[6]

質問項目	振動あり		振動なし		t 値
	平均	分散	平均	分散	
発言のしやすさについて	3.5	0.6	2.8	1	2.63**
発言タイミングのとりかたについて	3.6	1.5	2.6	0.8	2.61**

\*\*は 10%の有意水準片側 t 検定で有意

## ・ 実験 2

実験 2 では、2 人対話 4 対話のデータを利用している。比較対象は入力状況を振動で伝達する本システムと、視覚的に明示する MSN メッセンジャーである。チャット内容は対立課題のみで、各被験者は計 2 回のチャットを行う。

分析データには実験 1 で使用した総合アンケートと同様である。

全ての質問項目に対して、有意な差は得られなかったと報告している。

## 2.4 Chat Circles

Chat Circles の目的は、従来のチャットよりも表現豊かな意思伝達が可能なシステムの構築である[7].

Chat Circles ではメイン画面と発言履歴画面と 2 つのウィンドウから構成されており、メイン画面では、現在の発言、及び参加者の状態を確認でき、発言履歴画面では、過去の発言を閲覧できる。この 2 つのウィンドウはタブで切り替えることができる。

図 2. 4 に Chat Circles のメイン画面を示す。手法としては、図 2. 4 のように、色

や図形を用いて参加者の存在や活動状態をグラフィカルな表現で視覚化を行い，対面対話に近似させることを目標としている．

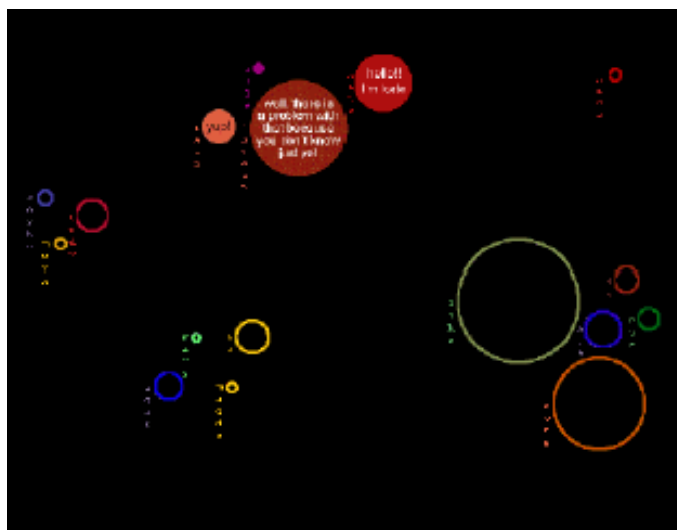


図 2. 4 Chat Circles : メイン画面[7]

システムの特徴は以下である．

- (1) 参加者の存在と活動情報の可視化
- (2) 参加者のフィルタリング機能
- (3) 履歴の可視化

### (1) 参加者の存在と活動情報の可視化

**Chat Circles** ではチャット参加時に色の選択を行い，参加者ごとにアバターとして色分けされた円で表示される．参加者が発言を入力すると，発言者の円が発言の長さに合った大きさまで拡大され，円の中に発言が表示される．ある程度時間が経つと発言はフェードアウトしていき，次第に円は元の大きさまで縮小する．これは対面対話において，発言をすると声音は徐々に消えていく特徴に対応している．

## (2) 参加者のフィルタリング機能

**Chat Circles** では、参加者を興味を持った者同士の小さなグループに細分化するため、近接ベースを基本としたフィルタリングを行う。アバターはウィンドウ内を自由に動くことが可能であり、アバターには視野範囲が設定されている。フィルタリング機能により参加者は、視野範囲内にいる参加者の発言のみ発言内容を見ることが可能である。視野範囲外にいる発言者については、発言者の円が大きく膨らんでいることのみ確認可能であり、円の中の発言内容を見ることは不可能である。これも対面対話において、現実世界において遠くの方で誰かが話している様子が見えるが、発言内容までは聞き取ることができないような場合に対応している。

## (3) 履歴の可視化

図 2. 5 に **Chat Circles** の発言履歴画面を示す。参加者は色分けされた縦に延びる線で表示される。図 2. 5 では、参加者は 3 名いる事がわかる。発言履歴画面では、縦軸に時間、横軸に発言の長さを示し、一番下が現在時間であり、発言は発言の長さに応じて縦線に交わる形で横線で表示される。発言内容を閲覧する場合は、マウスを閲覧したい発言まで移動すると、そこに自動的に発言内容が表示される。

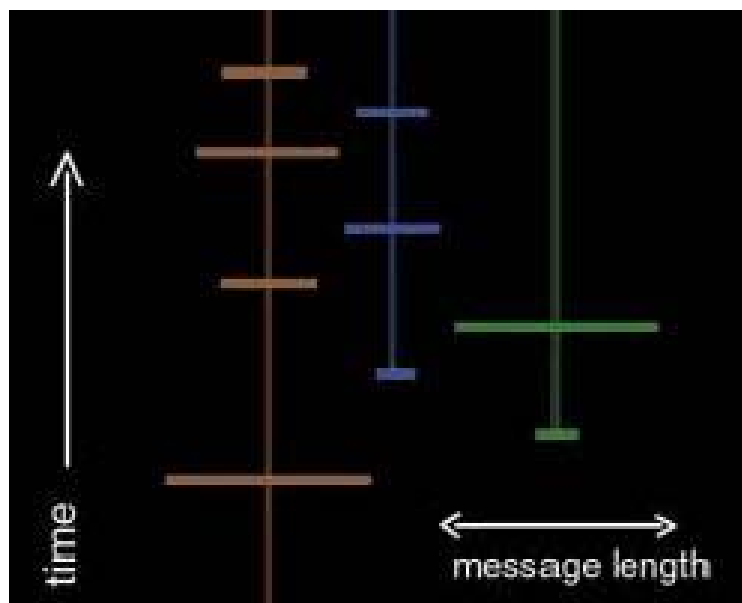


図 2. 5 Chat Circles : 発言履歴画面[7]



## 2.5 本研究との関連

本節では、前節までに説明した関連研究を踏まえて、本システムの特徴、違いを述べる。本研究でのシステムは **Flow Client** システムで扱った問題である、タイミング情報の欠乏問題を扱う。以下の 3 点を設計の指針として、この問題に対処するシステムを構築した。

1. 発言タイミングの共有
2. 発言開始時間に基づいた順番取りのサポート
3. 従来のチャット利用者でも違和感なく操作可能なユーザーインターフェース

1 については、発言タイミングを共有する手法として、初めから発言過程の文字情報を全て共有するのではなく、いくつかの手法を考えた。始めにプロトタイプシステムとして、参加者の打鍵が有り無し情報のみ伝えるシステムを実装した。次に発言開始時間のみを伝える機能、文字情報も含めた打鍵情報を伝える機能を実装した。機能の詳細については 3 章で述べる。2 については、**Flow Client** における発言過程の表示方法を参考にして、システム設計を行った。3 については、**Flow Client** での生じた問題であるユーザーインターフェースの問題について、従来のチャット利用者でも違和感なく操作可能なシステム設計を行った。具体的には、上から下へ表示される履歴表示構造を取り入れ、1 つの履歴表示欄に参加者全員の発言が表示されるようにした。システムの詳細は 3 章で述べる。

**Treaded Chats** では従来のチャットとは全く異なったユーザーインターフェースを構築しており、結果として、従来からのチャットユーザーにとっては取っ付きにくく、表示構造が複雑なため、瞬時に会話を理解しなければならないリアルタイムに会話を行うシステムとしては問題点が残る。

**Tangible Chat** では、タイミング情報を共有するため、振動を伝達する手法を取っている。しかし、振動では 2 人対話でのみタイミング情報を共有可能であり、参加者が 3 人以上の場合は振動と参加者との対応を取ることが困難である。本研究では、多人数でもタイミング情報が共有可能な手法を用いた。

**Chat Circles** ではグラフィカルな機能を多様に用いておりテキストベースのチャ

ットとは異なるものと考えられる。また **Chat Circles** では、履歴を保存することは不可能である。一般のチャットシステムの特徴として、履歴を電子データとして保存し、後から閲覧することが可能である。本研究では、テキストだけでなく、参加者名・発言開始時間・発言完了時間といった、発言の様々な属性も同時に保存可能な機能を実装した。履歴を統計処理する場合に便利なように、ファイルは **CSV** 形式のファイルで保存することにした。

最後に **Tangible Chat** を除く他のシステムでは、チャット利用の作業領域がかなり広く、**PC** 上での他の作業との同時進行は困難である。現状のチャットシステムのほとんどがディスプレイの **3** 割程度のサイズがデフォルトのサイズとなっており、これはチャット利用者の多くが、他の作業と並行してチャットを行う事を考慮したと考えられる。本システムでも、他の作業と並列に進行することが可能なようディスプレイの約 **3** 割のサイズとした。

## 第 3 章

# チャットシステムの構築

### 3.1 入力情報を波形で表示するチャットシステム

本節ではプロトタイプ版のシステムである波形チャットについて述べる。始めに手法とその特徴を述べ、システムの概要、特徴、操作方法を述べる。

#### 3.1.1 波形表示による打鍵情報の可視化

発言過程の情報の欠乏による問題を解決する手法として、打鍵情報の可視化を試みた。打鍵はチャットでは発言入力の際に必ず発生し、利用者が意識することなく伝達可能な非言語情報である。本システムでは、打鍵情報を可視化する手法として、打鍵情報を波で表示するシステムを構築した。波形表示での可視化の利点は以下の 2 点である。

- (1) 3 人以上の打鍵情報も確認可能
- (2) 過去の打鍵情報も確認可能

#### (1) 3 人以上の打鍵情報も確認可能

MSN メッセンジャー[3]ではステータスバーに「相手を書いています」と表示されるが、特定の誰が書いているのかは不明であり、3 人以上での対話だと、どちらの打鍵情報が表示されているのかわからない。また **Tangible chat**[5]では 2 人対話のみを想定したシステムであるので、複数の打鍵情報は送受信不可能である。本システムでは、各参加者ごとに波で打鍵情報を表示する。

## (2) 過去の情報も確認可能

打鍵情報の掲示方法として、MSN メッセンジャー[3]の「相手を書いています」表示や Tangible chat[6]による振動伝達では、その瞬間の打鍵情報のみを掲示している。この掲示方法だと、しばらくの間打鍵がなかった場合に、その沈黙の時間が、前の発言から全く打鍵がないのか、打鍵があったがその後送信されずに沈黙のままなのかといった両義性を持つ。よって本システムでは、ある程度過去の打鍵情報も波で表示し続けることにより、この両義性を解消した。

### 3.1.2 システムの構築

本システムの画面を図 3. 1, 図 3. 2 に示す。システムは主に 3 つのウィンドウから構成されており、図 3. 1 のスタートウィンドウで必要項目を記入後、接続が完了し終わると、メインウィンドウが表示される。

#### ・メインウィンドウ

図 3. 2 の左側がメインウィンドウである。メインウィンドウは主に発言の閲覧・記入を行うためのウィンドウである。左上方の複数行のフォームは発言表示欄であり、各利用者が入力した発言がサーバーにデータが送信された順に上から表示されていく。左下方の 1 行フォームは発言記入欄であり、ここに発言を入力し、最後にリターンキーを押すと発言は送信される。入力した発言は接続が未完成のままだと送信不可能であり、左下のステータスバーに「接続かんりょう」と書かれている時に送信可能である。

一方、ウィンドウの右側には現在接続されている利用者のユーザー名のリストが表示され、その下には現在の時刻が表示される。時間は PC 内のシステム時間であり、発言表示欄内の発言時刻はサーバーのコンピューターの時間である。

#### ・波形ウィンドウ

図 3. 2 の右側が波形ウィンドウである。波形ウィンドウでは各利用者の打鍵情報を波で表示する。波形ウィンドウはプログラム起動当初は表示されず、他の利用者がサーバーに接続された時に表示され、接続されるごとにそのユーザーの打鍵情報としてその都度表示されていく。図 3. 2 では、参加者が 3 名であるので、自分以外の 2

名の波形ウィンドウが表示されている。ウィンドウでは横軸に時間、縦軸に打鍵の有無を表わしている。波は時間とともに右側への流れていき、ウィンドウの右端まで行くと波は消去され、左端からまた始まる。波の色、種類、右端まで進むまでの時間は利用者が設定可能である。波の色は、ウィンドウ左下の 5 色から選択したい色の場所でマウスをクリックすると変化する。波の種類・時間はメニューバーから変更可能であり、種類は三角・四角・点の 3 種類、時間は 6 秒・1 分・10 分の 3 種類から利用者の好みに合わせて選択可能である。

### 3.1.3 考察

9 名の被験者による「発言タイミングの取りやすさ」、「波形での相手の入力状態の見易さ」、「どの種類の波が一番、見やすいか」、「どれぐらいの時間の波が画面上に見えたら良いか」などの簡単なアンケート調査を行った。波形表示が有る場合と無い場合で大きな差は得られなかった。しかし波形による打鍵表示は視覚的に見やすかったとの意見が得られた。また波の種類と時間の調査では、波の種類について特に点での表示は見にくい意見が多くあった。時間については、6 秒では短過ぎであり、10 分では長過ぎであるので、1 分ぐらいがちょうど良いとの意見が多数であった。

以上の結果より、発言タイミングを共有する手法として、単に打鍵が有り無しだけを伝えるだけでは、あまり効果が期待できないと考えられる。そこで、次のステップとして、発言開始時のタイミング情報を含む、入力文字情報を共有可能なシステムを構築した。

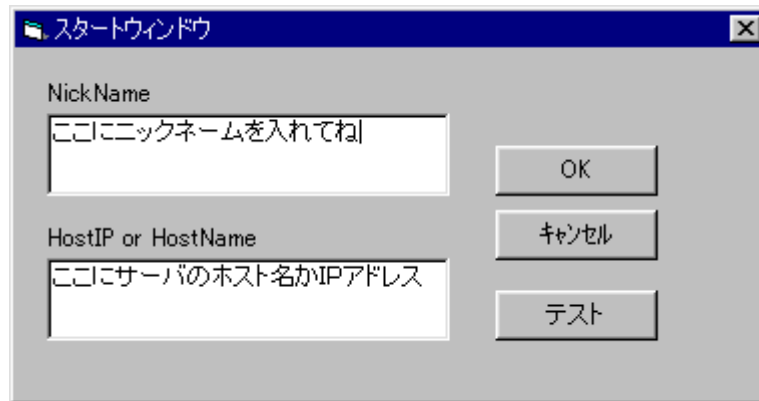


図 3. 1 スタートウィンドウ

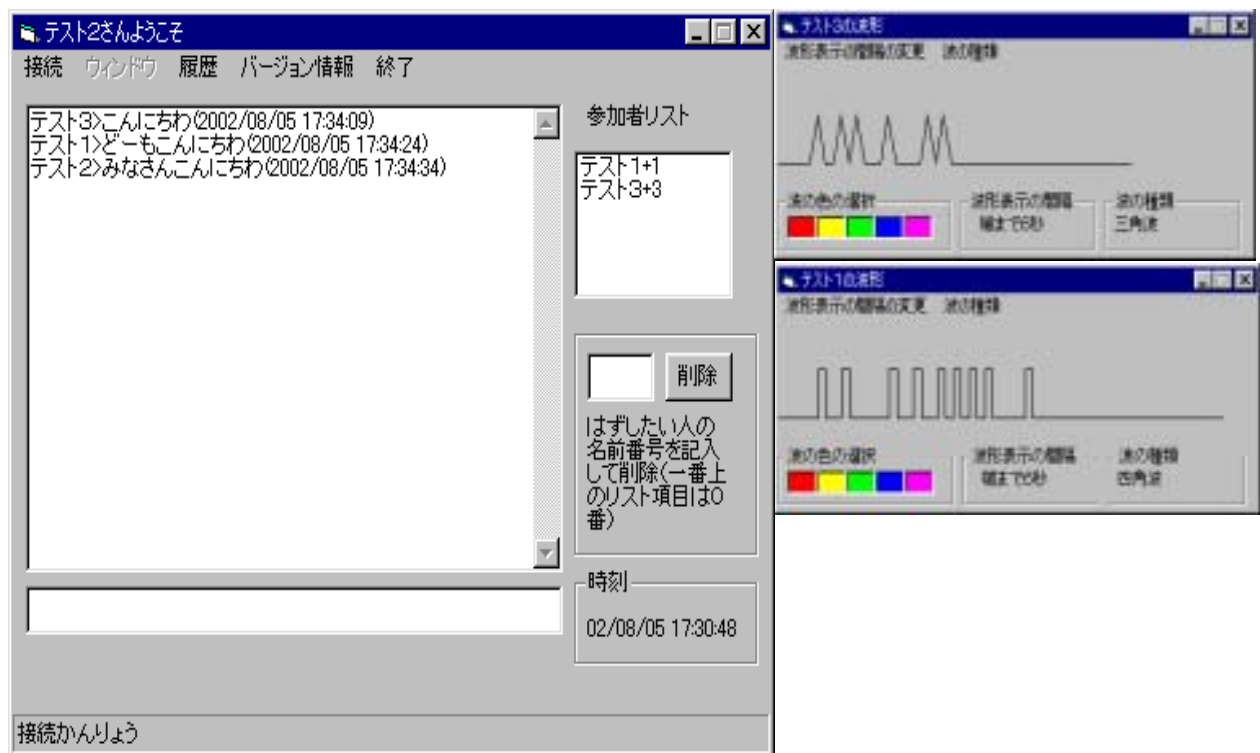


図 3. 2 メインウィンドウと波形ウィンドウ

## 3.2 入力情報，タイミング情報を共有できるチャットシステム

本節ではシステムの設計の指針である **1)発言タイミングの共有**，**2)発言開始時間**に基づいた順番取りのサポートをもとに構築したシステムについて述べる．初めにシステムの概要，特徴，使用方法について述べる．

### 3.2.1 概要

図 6 に本システムの状態遷移図を示す．本システムはサーバーアプリケーションと，クライアントアプリケーションの **2** つシステムを利用した **TCP/IP** プロトコルを用いたサーバー・クライアントシステムである．開発環境は **Microsoft Visual Basic6.0** を用いて構築した．クライアントアプリケーションは **3** つのウィンドウから構成される．クライアントアプリケーションを起動すると，まずスタートウィンドウが表示される．ここではサーバーアプリケーションとの接続を行う．**"OK"** ボタンをクリックすると自動的にメインウィンドウとサブウィンドウが表示される．メインウィンドウでは主に発言の入力・閲覧を行う．サブウィンドウでは，自分を除く参加者全員の入力状態が表示される．

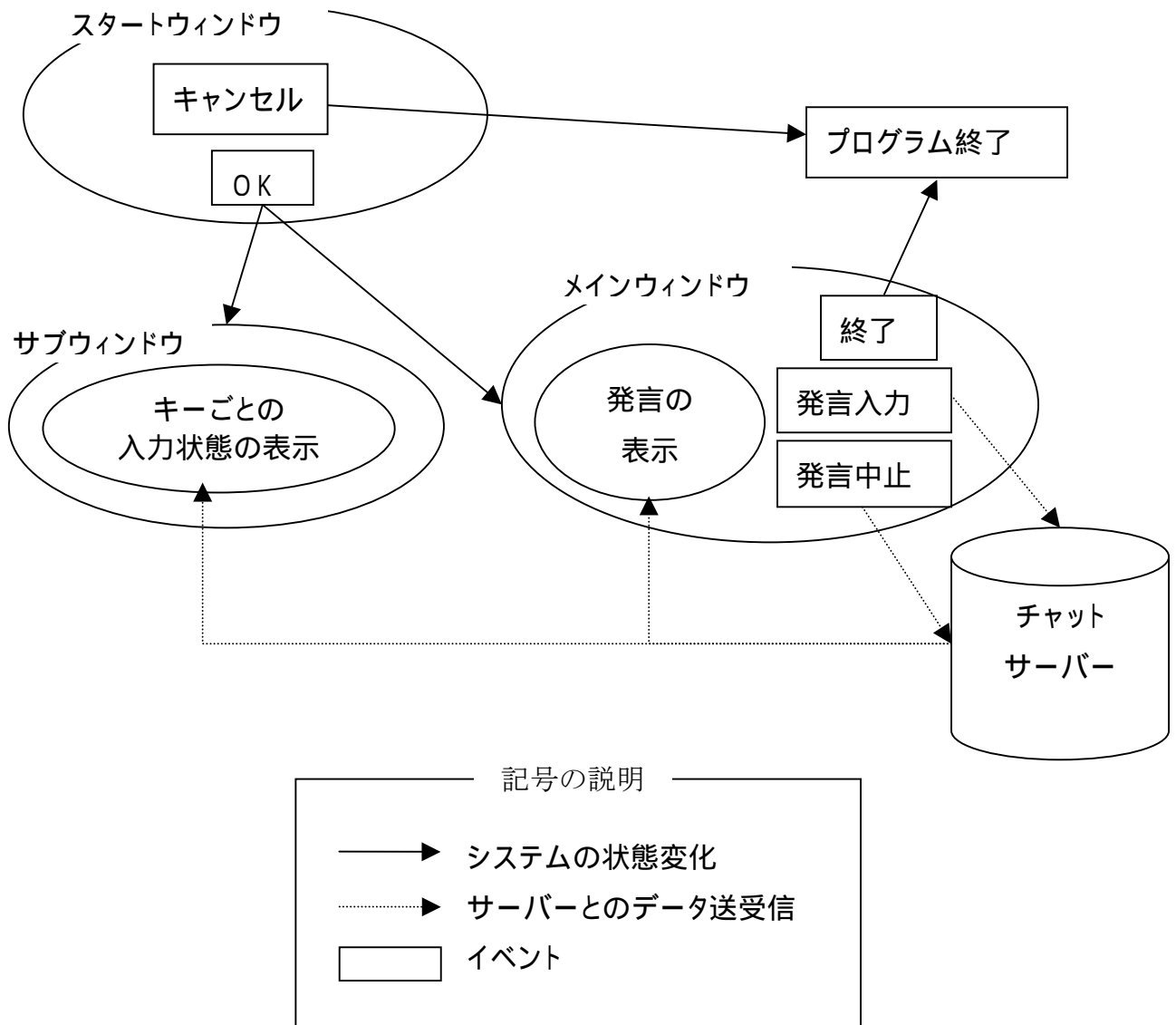


図 3. 3 システムの状態遷移図



## 3.2.2 特徴

メインウィンドウ画面を図 3. 4 に示す。メインウィンドウでは、発言表示欄と発言記入欄、それに参加者リストの 3 つのフォームが用意されている。発言表示欄では、参加者らの発言履歴が表示される。発言を入力する場合は、発言記入欄に発言を入力し、最後にリターンキーを押すと発言は送信される。現状のチャットシステムのほとんどがリターンキーを押す事により、発言を送信するため、この仕組みはチャット経験者にとって違和感なく利用可能である。参加者リストには、現在サーバーに接続されている参加者のニックネームが表示される。

本システムの特徴は以下の 2 点である。

- (1) 開始時間に基づいた表示方法
- (2) 入力状態の表示機能の実装

### (1) 開始時間に基づいた表示方法

対面対話では、誰かが発言を始めると、その人は発言者となり、他の人は聞く側になる。つまり対面対話では、発言権は発言を開始した時点で確保される。一方、従来のチャットシステムでは、発言は全て完了形で表示され、発言完了時間に基づいて表示される。つまり従来のチャットシステムでは、発言権は発言完了時点で確保される。このように、チャットは対面対話とは発言権を異なる取り方で行うため、うまくタイミングが共有できないと考えらる。本研究では発言開始時点で発言権を確保する機能を実装した。

本システムでは発言の入力を始めると、発言表示欄では"・・・書き始めています・・・"と表示され、発言欄が確保される。これより、従来のチャットでは発言完了時間の順番で表示されるのに対して、本システムでは、発言開始時間の順番に発言が上から下へと表示される。また、続いて他の参加者が発言を入力し始めても、発言欄は確保されたままであり、キータイピングが遅い人でも余裕を持って発言を入力可能である。

発言開始時間での発言欄確保により、発言開始後に発言を中止したい場合でも、発言欄は既に確保されている。このような場合に対処すべく、本研究では"発言中止ボタン"を実装した。もしある参加者が発言を入力途中で止めたい場合に、"発言中止ボ

タン”を押すことにより，発言表示欄には"□□□途中で止めました□□□"とシステム側で自動表示される．

また，"・・・書き始めています・・・"や"□□□途中で止めました□□□"は実際に参加者が入力した発言ではなく，システムが自動的に表示した文字列であるので，参加者の発言と区別するために，色を変えて目立たない色で表示した．

## (2) 入力状態の表示機能の実装

プロトタイプシステムの評価より，打鍵が有り無しのみでの発言タイミング共有は，あまり効果が期待できない．そこで本システムでは，相手の入力した文字情報を含む発言過程のタイミング情報を共有可能な機能を実装した．これにより，参加者は発言者の発言の入力状態を随時確認可能であるため，相手が入力途中でもその発言に対する次の発言を入力し始める事ができる．

サブウィンドウ画面を図 3. 5 に示す．他の参加者がサーバーに接続されると，サブウィンドウ内に上から順にニックネームが表示される．各ニックネーム横のテキストボックスに，そのニックネームで入室した参加者の現在の発言記入欄の状態が表示される．表示はその参加者の発言記入欄に文字が入力される度に随時更新される．



図 3. 4 メインウィンドウ

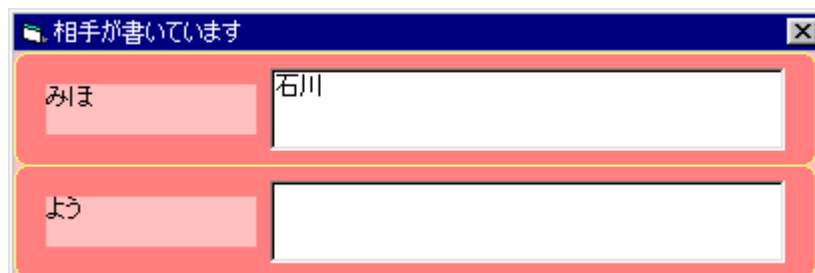


図 3. 5 サブウィンドウ

# 第 4 章

## 評価実験

本章ではシステムの評価について述べる。始めに実験概要について述べ、続いて実験で得られたデータの分析結果と考察について述べる。最後に本システムの機能について、対話内容からの分析と考察を述べる。

### 4.1 概要

本システムの特徴でもある、(1) 開始時間に基づいた表示方法 (2) 入力状態の表示機能の実装による 2 つの要因の効果を分析するため、以下の 4 つのシステムを作成した (表 4. 1)。

表 4. 1 評価システム

	発言表示順序	入力状態の表示
システム 1	完了順	無し
システム 2	完了順	有り
システム 3	開始順	無し
システム 4	開始順	有り

対話データは、1 対話 20 分間収録し、全てのシステムで 3 人対話によるデータを収録した。被験者は全員、ほぼ毎日計算機を利用しており、ある程度速くタイピングできる者である。評価には、定量的データと定性的データを用いて行い、全ての対話の発言履歴と、被験者によるアンケートにより評価を行った。

また、各機能ごとの影響を検証するため、相手の入力状態が見える／見えない、発言表示順序が発言開始順／発言完了順の 2 要因 2 水準の分散分析を行った (表 4.2)。

表 4. 2 要因と水準

	水準①	水準②
要因 <b>A</b> : 相手の入力状態が	見える <b>A①</b>	見えない <b>A②</b>
要因 <b>B</b> : 履歴表示方法が	発言完了順 <b>B①</b>	発言開始順 <b>B②</b>

## 4.2 定量的データの分析

定量的データについて、発言履歴の発言数、及び発言の長さの変化を調査した。本研究では次の仮説に基づき分析を行った。

- ・仮説 1：タイミング情報を共有することによって、発言数はより多くなる

タイミング情報が共有されない場合、参加者は発言する時に 1) 相手の反応を伺わずに発言する 2) 相手の反応を伺ってから発言する 2 パターンが考えられる。1) のパターンの方は、タイミング情報を共有する機能が有ったとしても、無い時と同じように相手のことを考えずに発言すると考えられ、本システムによる効果は期待できない。しかし、2) のパターンの方は、タイミング情報を共有することによって、相手の状態が把握でき、相手が発言を入力途中状態であっても、次の発言を準備可能である。そのため、お互いが発言完了を待ってから発言を打ち始めるといった無駄な時間が省けると考えられる。よって、2) のパターンの方は、タイミング情報を共有可能なシステムでは、共有しない場合よりも同時間内での対話において、発言数が増加する事が期待できる。以上の考察より 1) と 2) のパターンの方が混合して存在する場合、タイミング情報を共有することによって、発言数はより多くなると期待できる。

- ・仮説 2：発言権を確保する事によって、キー入力を急がなくても良いので、1 発言の長さはより長くなる

従来のチャットでは、リターンキーを押した順に発言が表示される。そのため参加者は他の誰よりも先に表示しようと争い、1 発言を短くし、発言する傾向があると考えられる[5]。しかし、発言欄を発言入力開始時に確保することにより、この争いからは解消され、ゆっくりと発言を入力できる。結果として、タイミング情報を共有する事によって、共有しない時よりも、1 発言の長さは長くなると期待できる。

以上の仮説を検証するため、始めに、発言数と発言の長さの平均（以降、発言の長さとする）の相関をチェックした。図 4. 1 に全対話における、被験者ごとの発言数と発言の長さの相関関係のグラフを示す。また表 4. 3 に相関係数を示す。

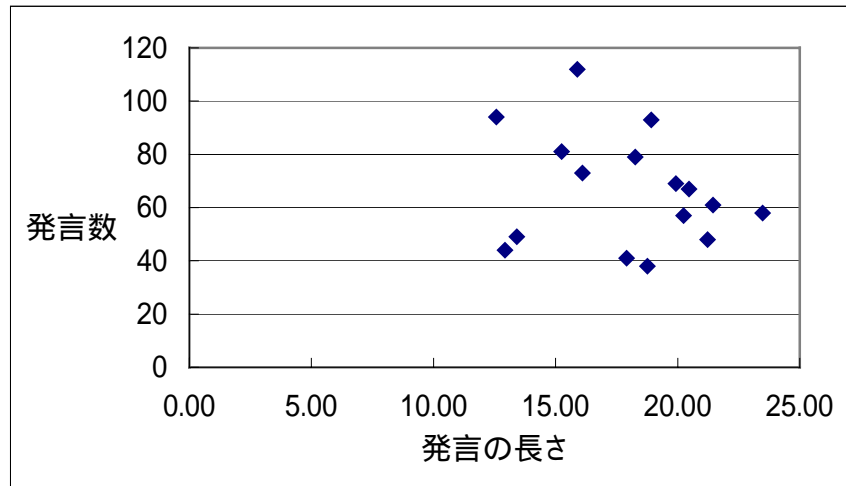


図 4.1 相関図

表 4.3 相関係数

相関係数	-0.23
------	-------

計算の結果から，発言数と発言の長さの相関が低いことが確認された。

続いて，被験者間における，発言数と発言の長さの平均，標準偏差を求め（表 4. 4, 表 4. 5），2 要因 2 水準による分散分析を行った（表 4. 6, 表 4. 7）。

表 4. 4 発言数の平均値と標準偏差

平均	B 完了順	B 開始順
A 見えない	16.83	25.33
A 見える	24.42	22.00
標準偏差	B 完了順	B 開始順
A 見えない	6.32	10.01
A 見える	10.40	8.61

表 4. 5 発言の長さの平均値と標準偏差

平均	B 完了順	B 開始順
A 見えない	17.82	16.48
A 見える	17.70	17.98
標準偏差	B 完了順	B 開始順
A 見えない	5.97	4.71
A 見える	6.04	4.09



表 4. 6 発言数における分散分析結果

	平方和	自由度	平均平方	F 比
要因 A	54.19	1	54.19	0.62
要因 B	111.02	1	111.02	1.26
A * B	357.52	1	357.52	4.07*
誤差	3869.73	44	87.95	1.00
		47		

\*は有意水準 5%で有意

表 4. 7 発言の長さにおける平均の分散分析結果

	平方和	自由度	平均平方	F 比
要因 A	5.70	1	5.70	0.19
要因 B	3.38	1	3.38	0.11
A * B	7.84	1	7.84	0.26
誤差	1332.47	44	30.28	1.00
		47		

分散分析の結果，発言数に関して，交互作用があることがわかった．発言の長さに関して有意差は見られず，仮説 2 を検証できなかった．

発言数において，交互作用が見られたので，水準ごとに再度分散分析を行った（表 4. 8）．

表 4. 8 発言数における水準ごとの分散分析結果

	平方和	自由度	平均平方	F 比
要因 A	54.19	1	54.19	0.62
B 水準	345.04	1	345.04	3.92*
B 水準	66.67	1	66.67	0.76
要因 B	111.02	1	111.02	1.26
A 水準	433.50	1	433.50	4.93**
A 水準	35.04	1	35.04	0.40
A * B	357.52	1	357.52	4.07**
誤差	3869.73	44	87.95	1.00
		47		

\*は有意水準 10%で, \*\*は 5%で有意

分析の結果, 履歴表示方法が発言完了順の時, 相手の入力状態が見えた方が見えな  
いよりも発言数が多い傾向があり, 入力状態が見えない時, 発言開始順の方が完了順  
よりも発言数が多いという結果が得られた.

これより, 各機能が単一に存在する場合に, 発言数は増加し, タイミングはうまく  
共有されたと考えられる. しかし両機能が同時に使用される場合, 有意な差は得られ  
なかった. これより以下の事が考えられる.

発言開始順による履歴表示方法は, 発言欄を確保すると共に, 発言開始タイミ  
ングを共有している. 相手の入力状態が見える機能は発言の文字情報と共に発言開始時  
を含めた発言過程のタイミングを共有しており, 両機能とも同じ効果を持つと考えら  
れる. しかし実験の結果, 両方の機能を同時に使用した場合において, 期待された効果  
は得られなかった. これは, これより, 同じ機能が異なるウィンドウ上に表示される  
ため, 利用者の注意が分散するからだと考えられる. 複数のタイミングを共有する機  
能を多様に実装すれば, それだけ情報量は増加し, 利用者は各情報に注意を分散しな  
ければならないため, 単に情報量を増加する事が有用でないことを示唆している.

### 4.3 定性的データの分析

定性的データとして，被験者に 1 対話終了ごとにアンケート調査を行った．表 4.9 にアンケート項目を示す．項目は全て 5 段階で得点付けを行い，1（全くそうでない）から 5（その通りである）まで設定した．全ての項目について，得点付けとその理由を書く記入欄を設けた．質問の性質上，質問番号 3 はシステム 2 とシステム 4 のみに回答し，質問番号 4 はシステム 1 を除く全てのシステムに回答する．

始めに各アンケート項目の相関をチェックするため，全アンケート結果を基に各項目の相関を求めた．相関係数の算出法として，スピアマンの順位相関係数を求めた（表 4.10）．

スピアマンの順位相関係数とは，2 つの変数間における各データの順位の一致度を表わす測度である[8]．今回のアンケート調査では，各項目に対して得点付けを行ったため，スピアマンの順位相関係数を用いた．

表 4.9 アンケート項目

質問番号	質問内容
1	落ち着いて発言を入力することができたか
2	話の流れに追いつけないことがあったか
3	相手の打ち込み状態が見える機能はあったほうが良いか
4	相手がタイピング中なので，自分のタイプを止めて待とうとした事があったか
5	発言タイミングは取りやすかったか
6	システムは使いやすかったか

表 4. 10 相関係数

各項目間の相関係数	
項目 1 と 2 の相関係数	-0.312
項目 1 と 3 の相関係数	0.357
項目 1 と 4 の相関係数	<b>0.448</b>
項目 1 と 5 の相関係数	<b>0.406</b>
項目 1 と 6 の相関係数	0.301
項目 2 と 3 の相関係数	-0.094
項目 2 と 4 の相関係数	0.074
項目 2 と 5 の相関係数	<b>-0.409</b>
項目 2 と 6 の相関係数	-0.148
項目 3 と 4 の相関係数	0.282
項目 3 と 5 の相関係数	<b>0.478</b>
項目 3 と 6 の相関係数	<b>0.497</b>
項目 4 と 5 の相関係数	0.340
項目 4 と 6 の相関係数	0.157
項目 5 と 6 の相関係数	<b>0.490</b>

各項目間の相関算出結果，項目 1 と 4 の相関は比較的強い相関を示した．しかし項目 4 は先に述べた通り，システム 1.では回答をしてない．そのため次に述べる分析方法が他の項目とは異なるため，今回は別々に扱った．また項目 5, 6 と他の項目との相関が比較的高いが，項目 5 と項目 6 はシステム全体の感想を聞く，他の項目の包括的な項目である．これらの項目は被験者の直感的な意見を調査する項目であるため，別々に扱った．

次に、システム毎の各項目の得点の平均値、標準偏差を求めた（表 4. 11）。

表 4. 11 アンケート結果

平均	項目 1	項目 2	項目 3	項目 4	項目 5	項目 6
システム 1	3.42	2.58			2.67	3.50
システム 2	4.00	1.92	3.08	3.33	3.67	3.75
システム 3	3.50	2.17		3.00	3.00	3.00
システム 4	3.83	1.58	3.83	3.67	3.92	4.00

標準偏差	項目 1	項目 2	項目 3	項目 4	項目 5	項目 6
システム 1	1.08	1.08			1.30	1.17
システム 2	1.04	1.08	1.38	1.44	0.65	0.62
システム 3	1.17	1.27		1.81	1.04	0.85
システム 4	1.27	0.67	1.19	1.37	1.31	0.74

更に定量的データによる分析と同様、2つの機能による効果を検証するため、質問項目 3 と 4 を除くものに対して、2 要因 2 水準の分散分析を行った。質問項目 3 と 4 についてはウェルチの t 検定により、システム間の有意差を検証した。表 4. 12～4. 19 に各質問項目ごとの各要因における分散分析の結果を示す。

表 4. 12 質問項目 1 (落ち着いて発言を入力することができたか) の結果

平均	<b>B 完了順</b>	<b>B 開始順</b>
A 見えない	<b>3.42</b>	<b>3.50</b>
A 見える	<b>4.00</b>	<b>3.83</b>
標準偏差	<b>B 完了順</b>	<b>B 開始順</b>
A 見えない	<b>1.08</b>	<b>1.17</b>
A 見える	<b>1.04</b>	<b>1.27</b>

表 4. 13 項目 1 の分散分析結果

	平方和	自由度	平均平方	F 比
要因 A	<b>2.52</b>	<b>1</b>	<b>2.52</b>	<b>1.81</b>
要因 B	<b>0.02</b>	<b>1</b>	<b>0.02</b>	<b>0.01</b>
A*B	<b>0.19</b>	<b>1</b>	<b>0.19</b>	<b>0.13</b>
誤差	<b>62.82</b>	<b>45</b>	<b>1.40</b>	<b>1.00</b>
		<b>48</b>		

表 4. 14 質問項目 2 (話の流れに追いつけないことがあったか) の結果

平均	<b>B 完了順</b>	<b>B 開始順</b>
A 見えない	<b>2.58</b>	<b>2.17</b>
A 見える	<b>1.92</b>	<b>1.58</b>
標準偏差	<b>B 完了順</b>	<b>B 開始順</b>
A 見えない	<b>1.08</b>	<b>1.27</b>
A 見える	<b>1.08</b>	<b>0.67</b>

表 4. 15 項目 2 の分散分析結果

	平方和	自由度	平均平方	F 比
要因 A	<b>4.69</b>	<b>1</b>	<b>4.69</b>	<b>3.99*</b>
要因 B	<b>1.69</b>	<b>1</b>	<b>1.69</b>	<b>1.44</b>
A*B	<b>0.02</b>	<b>1</b>	<b>0.02</b>	<b>0.02</b>
誤差	<b>52.82</b>	<b>45</b>	<b>1.17</b>	<b>1.00</b>
		<b>48</b>		

\*は有意水準 10% で有意

表 4. 16 質問項目 5 (発言タイミングはとりやすかったか) の結果

平均	<b>B 完了順</b>	<b>B 開始順</b>
A 見えない	<b>2.67</b>	<b>3.00</b>
A 見える	<b>3.67</b>	<b>3.92</b>
標準偏差	<b>B 完了順</b>	<b>B 開始順</b>
A 見えない	<b>1.30</b>	<b>1.04</b>
A 見える	<b>0.65</b>	<b>1.31</b>

表 4. 17 項目 5 の分散分析結果

	平方和	自由度	平均平方	F 比
要因 A	<b>11.02</b>	<b>1</b>	<b>11.02</b>	<b>8.38*</b>
要因 B	<b>1.02</b>	<b>1</b>	<b>1.02</b>	<b>0.78</b>
A*B	<b>0.02</b>	<b>1</b>	<b>0.02</b>	<b>0.02</b>
誤差	<b>59.18</b>	<b>45</b>	<b>1.32</b>	<b>1.00</b>
		<b>48</b>		

\*は有意水準 5%で有意

表 4. 18 質問項目 6 (システムは使いやすかったか) の結果

平均	<b>B 完了順</b>	<b>B 開始順</b>
A 見えない	<b>3.50</b>	<b>3.00</b>
A 見える	<b>3.75</b>	<b>4.00</b>
標準偏差	<b>B 完了順</b>	<b>B 開始順</b>
A 見えない	<b>1.17</b>	<b>0.85</b>
A 見える	<b>0.62</b>	<b>0.74</b>

表 4. 19 項目 6 の分散分析結果

	平方和	自由度	平均平方	F 比
要因 A	<b>4.69</b>	<b>1</b>	<b>4.69</b>	<b>5.82*</b>
要因 B	<b>0.19</b>	<b>1</b>	<b>0.19</b>	<b>0.23</b>
A*B	<b>1.69</b>	<b>1</b>	<b>1.69</b>	<b>2.09</b>
偶然	<b>36.27</b>	<b>45</b>	<b>0.81</b>	<b>1.00</b>
		<b>48</b>		

\*は有意水準 5%で有意

「話の流れに追いつけないことがあったか」に関しては相手の入力状態が見えない場合の方が見える場合より、得点が高い傾向を示した。これは、相手の入力状態が見えないため、発言のタイミングが上手く取れず、発言をする機会を失ってしまう場合があるためだと考えられる。

「発言タイミングは取りやすかったか」、「システムは使いやすかったか」に関しては、相手の入力状態が見える方が有意に高い点を示した。これは相手の入力状態が見えることにより、発言タイミングが取りやすく、システムの的にも使いやすい事が考えられる。つまりこれは発言タイミングが取りやすくなれば、チャットシステムは使いやすくなる事を示唆していると考えられる。

質問項目 **3**(相手の打ち込み状態が見える機能はあったほうが良いか)について、ウェルチの **t** 検定を行った結果を示す (表 **4. 20**)。

表 **4. 20** 質問項目 **3** の検定結果

	平均	標準偏差
システム <b>2</b>	<b>3.08</b>	<b>1.38</b>
システム <b>4</b>	<b>3.83</b>	<b>1.19</b>
自由度	<b>21.56</b>	
<b>t</b> 値	<b>-1.36</b>	

検定の結果、システム **3** と **4** における項目 **3** の有意差は見られなかった。すなわち、相手の入力状態が見える機能は、発言表示方法が異なるシステム間で差は無く、一定の評価が得られたことになる。被験者の意見として、「相手の入力状態が見えるのは好ましいが、自分の入力状態は見られるのは好ましくない」という意見が多数あった。しかしこの問題はトレードオフの関係にあり、両立は困難である。続いて、質問項目 **4** (相手がタイピング中なので、自分のタイプを止めて待とうとした事があったか) について、システム **2**, **3**, **4** のそれぞれについて、ウェルチの **t** 検定を行った結果を示す (表 **4. 21**)。



表 4. 21 質問項目 4 の検定結果

「システム 2 と 3」

	平均	標準偏差
システム 2	3.33	1.44
システム 3	3.00	1.81
自由度	20.92	
t 値	0.48	

「システム 2 と 4」

	平均	標準偏差
システム 2	3.33	1.44
システム 4	3.67	1.37
自由度	21.95	
t 値	-0.56	

「システム 3 と 4」

	平均	標準偏差
システム 3	3.00	1.81
システム 4	3.67	1.37
自由度	20.50	
t 値	-0.97	

検定の結果、全てのシステム間で有意差は見られなかった。すなわち質問項目 4 について各システム間での違いは見ることはできなかった。

## 4.4 発言履歴から見るシステムの影響

前節までは、評価実験に基づいた発言数と発言の長さ、アンケート調査の結果について述べてきた。本節では、本研究において、開始時間に基づいた表示方法、相手の入力状態が見える機能が、チャット対話そのものにどのような影響があったのかを考察する。本研究で構築したチャットシステムの効果として挙げた、発言タイミングの取りやすさについて発言の擬似的重複の点から検討する。また、履歴上での発言開始順の表示について履歴上での発言順序の入れ替えの点から検討する。

### 4.4.1 発言の擬似的重複

本研究では、相手の入力状況が見えることで発言状況を共有し、発言タイミング取りやすくする機能を付加した。発言タイミングが取りやすくなることが期待されることから、相手の発言状況が見えるものと見えないものとは、擬似的な重複箇所の減少が推測される。擬似的な重複とは、1章で述べたように、あるチャット参加者達が同時に発言した場合に、その発言間の意味的な関連が取れなく、対面対話での重複を想定させるものである。

擬似的重複と判断できる発言間インターバルがどれくらいかは明確ではない。そこで今回は、発言間インターバル 3 秒以下の発言を抽出し、[9]での擬似的重複の分類に基づいて、更に発言内容を加味した上で擬似的重複を判定した。表 4. 22 に擬似的重複の回数について、システム毎の全発言数における割合を示す。

表 4. 22 擬似的重複の割合

システム番号	1	2	3	4
擬似重複の割合	4.3%	6.4%	8.3%	2.7%

表 4. 22 より、システム毎に若干の差はあるものの、それほど大きな差は見られなかった。しかし、開始時間に基づいた履歴表示方法、相手の入力状態が見える機能のどちらかのみ付加されたシステム（システム 2, 3）は、特に付加された機能をもたないもの（システム 1）よりも、擬似的重複の割合が増加する結果になった。ここから、付加した機能は単一ではなく、発言の開始を明確にし、相手の発言状況が共有さ

れる機能の両方が組み合わさった場合に、発言タイミングの計り易さについて、有効に働く可能性が示唆される。

#### 4.4.2 履歴上での発言順序の入れ替わり

システム 3 および 4 において、発言は発言開始時間に基づいて、上から下へと発言履歴に表示されるため、発言の開始が先であっても、発言完了時では後になり、履歴表示上において、発言開始と発言完了の入れ替わりが起こる場合がある (表 4. 23)。そこで今回の実験において、発言開始と発言完了の入れ替わりがどれくらい起こったかを表 4. 24 に示す。

表 4. 23 表示順序による入れ替わりの例

「発言開始順の場合 (システム 3 の履歴)」

		開始時間	完了時間	発言内容
24	C	40:08	40:28	やっぱり関西出身の人は関西で関東より北の人は関東を好む傾向にありますよね。 関西は面白い人が多く、人情味がある感じがします。 そうかもしれないですね。 私にとって関西は未知の世界です。 なんとなく関西のほうが人がよさそうな気がしていいんだけど、結局関東なんですよ。
25	A	40:35	41:37	
26	B	40:35	40:40	
27	B	40:43	40:54	
28	C	41:03	41:19	

「発言完了順の場合」

		開始時間	完了時間	発言内容
24	C	40:08	40:28	やっぱり関西出身の人は関西で関東より北の人は関東を好む傾向にありますよね。 関西は面白い人が多く、人情味がある感じがします。 そうかもしれないですね。 私にとって関西は未知の世界です。 なんとなく関西のほうが人がよさそうな気がしていいんだけど、結局関東なんですよ。
26	B	40:35	40:40	
27	B	40:43	40:54	
28	C	41:03	41:19	
25	A	40:35	41:37	

表 4. 24 発言開始と発言完了の入れ替わりの割合

	システム 3	システム 4
入れ替え	17.3%	13.3%

表 4. 24 より，発言履歴表示上において，発言の開始順と完了順が入れ替わっていた発言が存在し，しばしば起こっていることがわかる．また，システム 3 とシステム 4 を比較すると，若干ではあるがシステム 4 の方が発言の開始，完了の入れ替わった発言割合が減少している．これは，発言開始時のタイミングを共有する事よりも，発言開始時から発言完了時までを含むタイミングを共有する事の方が，タイミング情報として明確であるためだと推測できる．

また，開始時間に基づいた履歴表示方法であるシステムでは，履歴上に発言箇所が確保されても，後にその発言を取りやめることのできる機能が付加されている．この機能の使用割合はシステム 3 で 4%，システム 4 で 2% とほとんど使用される機能ではなかった．この機能を使用した発言者をみた場合，全ての実験において，特定の被験者のみが使用していることがわかった．その理由として，1) 使いづらいため，特定の人しか使用しなかった．2) 他の発言者よりも発言の完了が遅い人が，特定の人だったの 2 つの可能性が考えられる．

# 第 5 章

## まとめと今後の課題

### 5.1 まとめ

本研究では、タイミング情報を共有するチャットシステムを構築した。タイミング情報を共有するために、発言開始時間に基づいた表示方法、相手の入力状態が確認可能な機能の実装を行った。発言数に関しては、履歴表示方法が発言完了順の時、相手の入力状態が見えた方が見えないよりも、発言数が多いと有意な差を示した。また入力状態が見えない時、発言開始順の方が完了順よりも、発言数が多いと有意な差を示した。

アンケート調査による結果、本論文で提案した、発言表示方法が開始順であり、相手の入力状態が見える機能を持つシステムを用いた場合、従来のシステムと比べ、(1) 発話タイミングが取りやすく、会話が円滑に進む (2) システムが使いやすいという 2 点に関して有意な結果が得られた。

### 5.2 今後の課題

評価実験より、複数のタイミング情報を異なる位置で提示すると、利用者の注意が分散してしまう可能性があることが示唆された。よって、本システムにおけるサブウィンドウでの相手の入力状態の表示ではなく、直接履歴上で相手の入力状態が見える機能を実装することにより、タイミング情報を共有する効果が一層期待できる。

チャットシステムの問題は 1 章で述べたように、タイミング情報の欠乏以外にも様々な問題がある。そこで最終目標のシステムとしては、複数の問題に対応したシステムが望ましいと考え、本システムに有用な付加機能を考察した。

さらに、タイミング情報の欠乏を一因として起こる、隣接する発言が意味的に対応しない問題を解決するために、関連情報の欠乏に関する問題に注目し、関連発言を色分けする機能を実装した。本機能は発言者がどの発言に関連しているかを指示し、関連付けされた発言同士は色分けする仕組みである。本機能について有用性を示すため、アンケート調査を行った。結果、ほぼ全員があつた方が良いと回答した。特に、発言関連指示機能のユーザーインターフェースに関しては、改良が必要である事が調査より明らかとなった。改良方法として、関連を自動的に付与する機能や、色分けではなく、関連発言同士が線などで直接結ばれるような、より視覚効果の高い機能を付与する事が考えられる。

# 謝辞

本研究を進めるにあたり，石崎雅人助教授には，数々のご指導・ご助言を頂き，心から感謝致します。

また本研究と関連した副テーマにおいて的確なご指導を頂いた白井清昭助教授，本論文の中間審査，本審査において貴重なご指摘を頂いた **Ho Tu Bao** 教授，下嶋篤助教授，林幸雄助教授に深く感謝致します。

最後に普段のミーティングでアドバイスを頂いた石崎研究室の学生の皆様方に感謝致します。特に植田繁雄氏には研究全般でのアドバイス，評価実験のサポート，角野正高氏には評価実験のサポート，小倉加奈代氏には，評価データの正解データ作成に協力していただきました。ここに感謝の意を込めて明記させて頂きます。

## 参 考 文 献

- [1] 細馬宏通 (2000) チャットは何を前提としているか—チャットの時間構造と音声会話の時間構造, 「身体性とコンピュータ」bit 別冊, pp.339-349, 共立出版.
- [2] 水上悦雄, 右田正夫 (2002) チャット会話の秩序—インターバル解析による会話構造の研究, 認知科学, Vol.9, pp.77-88.
- [3] MSN メッセンジャー, <http://messenger.microsoft.com/>.
- [4] Vronay,D., Smith,M., and Drucker,S.(1999) Alternative Interfaces for Chat, Proc. of the12<sup>th</sup> Annual ACM Symposium on User Interface Software and Technology (UIST99).
- [5] Smith,M., Cadiz,J.J., Burkhalter,B. (2000)Conversation Trees and Threaded Chats, CSCW 2000,ACM Press, pp97-105.
- [6] 山田裕子, 平野貴幸, 西本一志 (2002) Tangible Chat:キーボードチャットにおける触覚を利用した対話状況アウェアネスの伝達, 情報処理学会, グループウェアとネットワークサービス研究会, 43-18, pp.103-108.
- [7] Fernanda B.Viegas and Judith S.Donath(1999)Chat Circles , InCHI , Conference proceeding , pp9-16 .
- [8] 吉田寿夫, 森敏昭 (1990) 心理学のためのデータ解析テクニカルブック, 北大路書房.
- [9] 小倉加奈代, 石崎雅人 (2002) チャット対話の話題推移に関する特徴分析, 人工知能学会, SIG-SLUD-A202-3, pp.13-19.



## 発表論文

- [1] 益田武士, 石崎雅人 (2002) 「波形チャットシステムの構築」平成 14 年度電気関係学会北陸支部連合大会, pp386
- [2] 益田武士, 石崎雅人 (2003) 「時間情報を共有可能なチャットシステムの構築」平成 15 年情報処理学会全国大会 (to appear)
- [3] 益田武士, 小倉加奈代, 石崎雅人 (2003) 「時間情報を共有可能なインターネットチャットシステムの構築」, 人工知能学会, 第 37 回言語・音声理解と対話処理研究会 (to appear)