

同期型電子会議へのスムーズな途中参加支援のための一方式

川口明彦[†] 加藤善大^{††}, 石原進^{†††},
酒井三四郎^{†††} 水野忠則^{†††}

継続中の同期コミュニケーションへのスムーズな途中参加を実現するための一方式として、ユーザ不在期間中の発言データを時間・サイズおよび重要度によって区分けしたブロックに分け、これらの組合せによって不在期間のダイジェストを作成・提供するブロック化ダイジェスト方式を提案する。本方式はユーザが同期コミュニケーションへ途中参加後すぐに発言できるようなダイジェストデータの提供のみならず、その再利用も目的としている。本稿ではブロック化ダイジェスト方式の詳細、同期・非同期電子会議システム DYNAMITE 上での実現と、同期コミュニケーションへの途中参加実験およびダイジェストの比較実験について報告する。実験に基づいた評価の結果、本方式は比較的頻繁に会話内容が変わる会議であっても、会議へのスムーズな途中参加支援のために有効であることを確認した。

A Digest Making Method for Helping Users to Join to Teleconference from Halfway

AKIHIKO KAWAGUCHI,[†] YOSHIHIRO KATO,^{††} SUSUMU ISHIHARA,^{†††}
SANSHIRO SAKAI^{†††} and TADANORI MIZUNO^{†††}

This paper describes a digest making system for helping users to join to teleconference from halfway. A Blocked Digest Method is utilized to make digests which is made of several unit of digest-blocks. The digest-blocks are made by classifying users' speech data to 2 or more priorities according to the intervals of speeches and the amount of applause and so on, and segmente them. This method has a feature, that the system can reuse the digest-block to make other new digests. We have implemented this method and merged into the DYNAMITE (that is a sync- async mobile teleconference system.), and we have evaluated it. As the result of evaluation with questionnaires, we have made sure that this method is effective for helping users to join naturally to the realtime communications.

1. はじめに

近年めざましい発展をとげているモバイルコンピューティング技術により、ユーザはいつでもどこでもコンピュータを使ったコミュニケーションが利用可能となりつつある。ユーザの行動の制限がなくなる一方で、ユーザの時間的・地理的な都合やその通信コストの高

さから、モバイル環境下では同期コミュニケーションへの常時参加が制限される可能性が高い。また、モバイル環境のみならず、固定通信環境においても遠隔地にいる多人数がネットワークを介して同期的なコミュニケーションを行う場合、従来のコミュニケーションシステム^{1)~4)}ではユーザの都合や通信環境の違いなどにより、すべてのユーザが同時刻にコミュニケーションに参加することは難しい。

これらの問題を解消するコミュニケーションシステムの1つとして、コミュニケーションへの時間的に自由な参加・退出が可能である電子会議システム DYNAMITE(DYNamic Adaptive Mobile TELEconference system ⁵⁾がある。DYNAMITEでは、会議途中に会議に参加あるいは再参加するユーザ(非同期ユーザ)のためにすべての会議発言がサーバに蓄積される。非同期ユーザは、マルチメディアデータベースに保存された会議中の各発言を随時参照できる。また DYN-

[†] 静岡大学大学院情報学研究科

Graduate School of Information, Shizuoka University

^{††} 静岡大学大学院理工学研究科

Graduate School of Science and Engineering, Shizuoka University

^{†††} 静岡大学情報学部

Faculty of Information, Shizuoka University

現在、株式会社サイボロニクスジャパン

Presently with Cyberonix Japan, Inc.

現在、静岡大学工学部

Presently with Faculty of Engineering, Shizuoka University

MITEでは、ユーザの要求に応じて、会議中の一部の発言を抽出したダイジェストの提供も可能である。しかしながら、会議参加後に過去の発言を参照できるとはいえ、ユーザが過去の会話の流れを理解することなく会議に途中参加した場合、リアルタイムで進行している会議に参加・合流するタイミングを失いかねない。

関連研究には DYNAMITE と同様に同期・非同期型の電子会議システムである ASSIST⁶⁾ や会議中の任意の発言に対して意味的な構造化を試みる試験システム Jabber⁷⁾ があるが、いずれも途中参加者に対して特別に情報提供を行うものではない。

以下本稿では、2章で途中参加直前までの会議の内容の要約を自動作成しユーザに提供することで、途中参加ユーザのスムーズな会話への参加を支援する方法を提案する。またこのための具体的な手法として、3章でブロック化ダイジェスト方式を提案する。4章で本方式を利用した会議への途中参加実験による提案方式の検証について述べる。

2. ダイジェストによる非同期会議支援

DYNAMITE ではすべての会議中の発言情報を蓄積している。非同期状態のユーザが同期会議に途中参加する際に、不在期間中の会議内容のダイジェストを作成・ユーザに提供することで、ユーザは短時間でその内容を把握し会議へと合流することができる。この際ダイジェストは会議同様にマルチメディアデータによって提供される。これは文字や絵だけでは伝わらない会話の微妙なニュアンスなどを伝えるのに効果的であるからである。

しかしながら、同期コミュニケーションへ途中参加するユーザにダイジェストを提供する場合、以下のような問題がある。

- ダイジェスト参照中に同期会議が進行する
ユーザがダイジェスト参照している間にも会議が同時進行してしまい、ユーザはその間の会議内容を知ることができない。このため、ユーザがダイジェストを参照して途中参加したとしても、参加直前の情報不足により、参加後すぐに発言できるとは限らない。
- ダイジェストデータの再利用性
複数のユーザがダイジェストを要求した場合、たとえそれが同時刻における要求であっても、ユーザごとにダイジェストを作成していたのではサーバに負荷がかかり、ダイジェスト結果を迅速に提供できないことも考えられる。特にダイジェスト作成に高度な処理が必要な場合は顕著である。

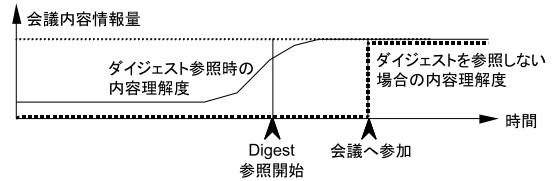


図1 スムーズな途中参加のための会議内容理解
Fig. 1 Understanding the past conference information for smoothly joining the conferences from halfway.

3. ブロック化ダイジェスト方式

ブロック化ダイジェスト方式は前述のダイジェスト作成・提供における問題点を考慮し、次の2つを目的としている。

- 同期コミュニケーションへのスムーズな途中参加を考慮したダイジェスト作成・提供
- ダイジェストの再利用性

同期コミュニケーションへのスムーズな途中参加本方式の目指すところは、ダイジェストを提供することにより、なるべく短時間でユーザ不在期間の会議の流れを把握し、同期会議参加直後に会話に参加できる状態にすることである。

同期会議参加直後に会話に参加するためには、会議全体の流れに加え、参加直前の会話の内容を理解していることが重要と考える。そこで図1に示されるように、ユーザ不在期間の会議内容の理解度が同期会議参加直前に向かって段階的に増加するようにダイジェストを作成する。

ダイジェストの再利用性

以前の DYNAMITE⁵⁾ では、ユーザがダイジェストを要求するたびに会議の全発言に対する解析を行い、その中から発言を抽出することでダイジェストを作成していたため、サーバに負荷がかかるという問題点があった。

本方式では会議時間をある期間ごとのブロックに分割する。ブロック中の発言にはその発言の重要度に対応した優先度を付与し、ブロック内をさらに優先度ごとに分割し格納する。ユーザの到着時には、そのユーザの要求するダイジェスト参照量を、実際の再生時間を考慮しつつブロックを選択することでダイジェストを作成する。

これにより、あらかじめ作成されたダイジェストのブロックを組み合わせることによってダイジェストを提供できる。このため、同期会議参加時刻が異なる複数のユーザに対して、図1のような理解度曲線を与えるダイジェストを容易に提供可能である。すなわち、

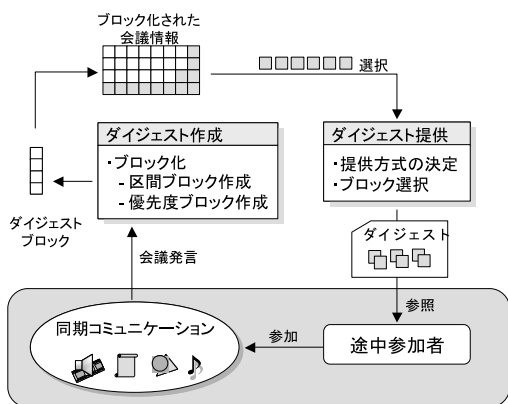


図2 ブロック化ダイジェスト方式の流れ
Fig. 2 The flow of making a digest.

会議へ途中参加する直前の会話の内容は多く理解させてスムーズに会話に入れるようにし、途中参加の時刻よりも十分に前の会話の内容は概要が分かる程度のダイジェストとする。

ブロック化ダイジェストの作成から提供までの流れを図2に示す。以後、本方式のブロック化によるダイジェスト作成および提供について述べる。

3.1 ブロック化によるダイジェストの作成

本方式では会議時間または発言量を元に、会議をある一定期間ごとにブロックとして分割する。このように分割されたブロックを区間ブロックと呼ぶ。また、区間ブロック内の発言に対し優先度の抽出を行い、その優先度に対応したブロックを作成する。これを優先度ブロックと呼ぶ。

3.1.1 区間ブロック分割

区間ブロックは会議発言のデータサイズを指標として分割する。ただし、発言が少なく、ある一定期間を経過してもブロックが作成されない場合には、指定のデータサイズに満たない場合でも1つのブロックとして分割する。

3.1.2 優先度ブロック分割

ブロック内に設定される優先度は1を最高として少なくとも4以上の優先度付けを行う。また各優先度に対し、システムまたはユーザがブロック中の全データサイズの何%を抽出するかを指定し、指定量の発言を抽出する。優先度の段階数および抽出割合は会議ごとに指定可能であるが、筆者らのプロトタイプでは4段階の優先度を考えることにしている。

DYNAMITEではユーザの発言と同時に会議ログをとっており、動画、音声、テキスト、ホワイトボード、拍手などの発言情報ははじめとして、会議開始時間、ログイン時間、ユーザ情報、その他を格納してい

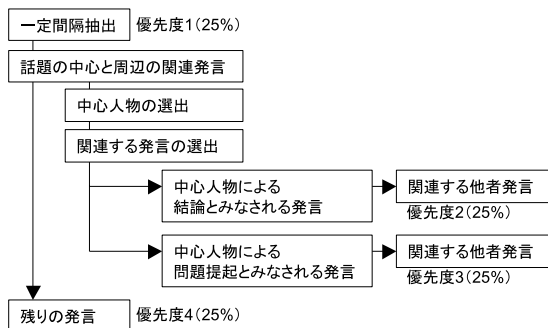


図3 優先度抽出の基本概念
Fig. 3 The basic concept of selecting priorities.

る。発言の優先度抽出はこの会議ログを元に行われる。次に各優先度における発言抽出の基本的な考え方を述べる。また図3にその概略を示す。

優先度1

会議の内容を理解するというより意味の流れを把握することを重視し、最も選択される可能性のある優先度1では一定間隔で発言を抽出する方法をとる。これまでのダイジェスト参照実験により、ごく少量の発言抽出量で会議内容を理解するためには発言間の関連を考慮して抽出を行うよりも、一定間隔で抽出するほうがユーザが会議内容の流れを判断するためには良いことが分かっている⁵⁾。

優先度2および3

優先度2,3は優先度1における会議の流れの理解に加え、重要部分の内容理解を目的とする。そのための方針として、区間ブロック中における会話の中心人物と他者とが活発に意見を交換している場所を優先的に抽出することとした。これは、通常の会議であれば、議長や発言の活発なユーザのようなコミュニケーションの中心となるユーザの発言を軸に会話を進行していくことが考えられるからである。

中心人物の選出は、それぞれのユーザの「発言時間」、「発言回数」、「指名されて発言された回数(被指名回数)」、「拍手された回数(被拍手回数)」、「すべてのユーザに対する発言数(広報発言)」を指標として、これらに重み付けを行い総合点の高いユーザを中心人物とする。

次に、中心人物を基点として他者との意見がやりとりされている場所を選出する。これはユーザ間の発言遷移や、中心人物がだれに対して発言しているかを調査することで判断できる。

話題の結論は会話中の決定事項と結び付く可能性が高い。そこで、一連の会話中で中心人物のある発言に注目した場合、その発言より以前の他者発言数、各発

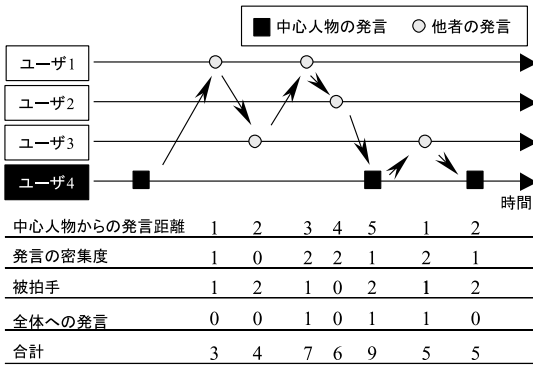


図4 中心人物の結論の抽出例

Fig. 4 An example of determining a core person.

言の時間的密度, 被拍手回数, 広報発言を指標として総合点の高い中心人物の発言を話題の結論と見なし(図4), これを優先度2とする。目標とした割合の発言が選ばれるまでこの手順を繰り返す。優先度3では優先度2と同様の手法を用い, 中心人物による一連の会話の開始発言を問題提起を行った発言と見なし発言抽出を行う。

優先度4

これまでの発言抽出では選択されなかった発言を選択する。

3.2 ダイジェストの提供

ダイジェストの提供は, 優先度ごとに分割されたブロックを途中参加ユーザの希望するダイジェスト抽出量だけ選択し, これをダイジェストとしてユーザに提供する。

しかしながら2章で述べたとおり, ユーザはダイジェストの参照中に同時進行する会議の内容を知ることができない。したがってユーザはこの空白時間が長いほど会議への途中参加に重要である直前の会議内容を失うことになる。そこで, 本方式ではダイジェストの提供を図5に示すように2段階に分けることで空白時間を短縮する。

以下, 2段階によるダイジェスト提供方法の詳細について述べる。

3.2.1 2段階によるダイジェスト提供方法

ユーザの要求するダイジェスト参照時間を T_{req} とする。1段階目ダイジェストの作成対象期間(ユーザの不在期間)を T_1 とし, 期間 T_1 を対象としたダイジェストを D_1 とする。このとき D_1 を参照するのに必要な時間は T_2 であり, 同時に2段階目のダイジェスト作成対象期間が T_2 , その参照時間が T_3 となる。また, ユーザがダイジェストを参照し始める時刻を t_a , 会議に参加できる時刻を t_s とすると, 時刻 $t_a + T_2$

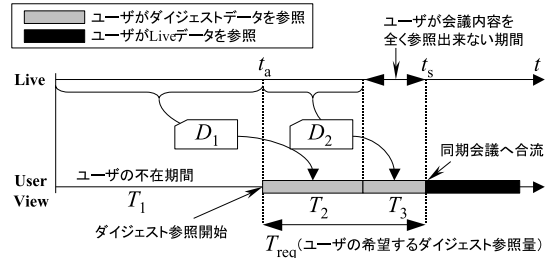


図5 2段階によるダイジェストの提供

Fig. 5 Producing digests with two steps.

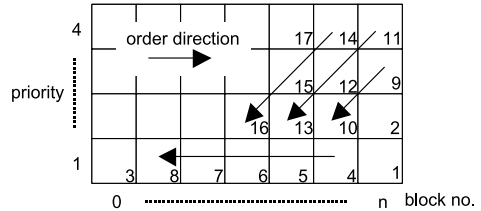


図6 ブロック選択方針

Fig. 6 The policy of selecting blocks.

から $t_a + T_2 + T_3$ までの間(期間 T_3), ユーザは会議の内容をまったく知ることができない。

そこで本方式では, ユーザの要求するダイジェスト提供目標を満たす適切な T_2, T_3 を決定し, これをダイジェスト提供目標量として優先度ブロックの選択を行い, ユーザに提供する。ブロックの選択は図1のような理解度を得るよう, 図6に示す番号順に従い会議参加直前の会議情報を優先して選択する。

会議参加直前の情報欠損を最小限におさえるためには T_3 を短くする必要がある。しかし同時に参加直前の情報をより多く知るためには D_2 を大きくしなければならない。一方, T_3 を短く設定することで空白時間を短くすることができるが, この結果 D_2 が減り, 参加直前の会議情報が減少してしまう。このように空白時間の短縮と参加直前の会議情報の重視は相反する特性を持つ。実装においてはこれらの適切な妥協点を見いだして T_2 および T_3 を決定する必要がある。

具体的には, ユーザが我慢できる限度の空白時間を仮に T_w とし, 空白時間がこれよりも小さくなり, かつ D_2 ができるべく大きくなるように T_2, T_3 を決定する。ただし, ユーザからのダイジェスト要求時間が空白限度時間 T_w より小さい場合には, ユーザがダイジェスト参照中の空白時間を許容できるとして, 1段階のみのダイジェスト提供を行う。

3.2.2 最適な T_2, T_3 の決定

期間 T_1 中で実際に会話が行われていた時間の割合を発言存在率 $R (< 1)$, 話速変換による発言再生時間

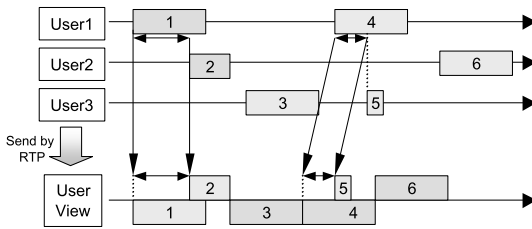


図7 ダイジェスト発言の送信順序制御

Fig. 7 A control of sending summarized speech data.

の短縮率を $C (< 1)$ ，期間 T_2 中の全発言のうちダイジェストとして選択される発言の割合を d_2 とする．期間 T_2 でも期間 T_1 と同じ発言存在率であると仮定すると，期間 T_2 中に予想される会議発言時間は $T_2 R$ であるから，ダイジェスト D_2 の参照時間 T_3 は

$$T_3 = T_2 R C d_2 \quad (1)$$

と計算できる．

本方式では T_2 期間中に少なくとも半分以上の会話を知る必要があると仮定し $d_2 > 0.5$ とする．これは T_2 の期間がダイジェスト支援期間の後半部分にあり，抽出される発言の割合は概して大きいという前提に基づく．したがって，式 (1) より，

$$d_2 = \frac{1}{R \times C} \frac{T_3}{T_2} \geq 0.5 \quad (2)$$

また，期間 T_3 が空白時間となるので，

$$T_3 \leq T_w \quad (3)$$

式 (2)，(3) を満たすような T_3 値を適切な空白時間 T_3 の目標値，また $T_2 = T_{\text{req}} - T_3$ を T_2 の目標値とする．

3.2.3 ダイジェストの送信

DYNAMITE では多人数同時発言が認められているため，2人以上の発言が時間的に交差する状況がある（交差発言）．このような交差発言がダイジェストとして選択された場合，その交差状況を再現しつつ短時間で全発言を送信する必要がある．そこで図7に示すように，ダイジェスト中でも会話の重なりを再現し，かつ発言間の空白時間をゼロにするようにダイジェストの送信を行う．

4. 実装と評価

4.1 実装

実際にブロック化ダイジェスト方式を DYNAMITE システム上で実現した．また，これに合わせ DYNAMITE システムを大幅に変更した．また，会議クライアントにダイジェスト参照用のビューア（会議進行状況の確認や，単一発言の再生が可能）を組み込んだ．利用できるメディアは音声・テキスト・ホワイトボー



図8 会議画面

Fig. 8 A screenshot of a teleconference.

ド・拍手であり，現段階ではビデオは未実装である．チャット・ホワイトボード・制御データは TCP によるユニキャスト通信，ライブの音声発言の送信は RTP を利用したマルチキャスト通信，ダイジェストの音声発言の送信は RTP を利用したユニキャスト通信をそれぞれ用い，音声コーデックは DVI Codec 32 Kbps とした．実装言語は Java であり，マルチメディアライブラリとして Java Media Framework⁹⁾，音声短縮ライブラリとして Chrono Player¹⁰⁾を使用した．

さらに，会議サーバには途中参加シミュレータを追加実装した．途中参加シミュレータは，終了した過去の会議における発言やホワイトボードの書き込みを，指定した時刻から忠実に再現・再生する．これを用いることにより，ユーザはすでに終了している会議にあたかも途中参加するような体験ができ，それまでの会議内容のダイジェストを参照することが可能である．

4.2 会議への途中参加実験

ブロック化ダイジェスト方式を組み込んだ DYNAMITE システムで実際に会議を行った（図8）．会議は2人の途中参加者を含んだ8人で行われ，約1時間程度を要した．途中参加者には，会議中盤以降に途中参加およびダイジェストの参照をしてもらい，感想を聞いた．

4.2.1 会議条件

会議の条件を表1に示す．会議という名目上，1人の参加者を調停役として指名したが，各会議発言に発言権・書込権などの制限を設けず比較的自由に発言を行う雰囲気の中で会議を行った．拍手は他者の発言に対し「重要である」「賛同できる」と判断した場合に実行してもらった（拍手ボタンをクリックすることで実行できる）．途中参加者はほぼ同時刻に途中参加

し、それぞれ 90 秒と 180 秒のダイジェストを要求してもらった。途中参加者 1 は 90 秒のダイジェストを要求したが、これはダイジェスト要求量が前述した空白時間 ($T_w \leq 90$) の限度設定値内であるため、1 段階のみのダイジェストが作成・提供された。また、途中参加者 2 は 180 秒のダイジェストを要求した。このダイジェスト要求量は限度設定値 90 秒以上であるため、2 段階のダイジェスト提供が行われた。また、実際のダイジェストは音声発言のみを参照してもらった。

発言の優先度抽出の割合を表 2 に示す。優先度 1 は会議の流れを把握するには会議全体の 18% 程度の発言を参照する必要があるという報告⁵⁾に基づき 20% とし、その他の割合は単純に残りの発言をほぼ 1/4 ずつ抽出するよう設定した。

4.2.2 会議結果

会議結果は表 3 のようになった。

表 1 会議条件
Table 1 The terms of conference.

会議内容	「究極の未来の端末」をテーマとしたフリーディスカッション
会議時間	73 分
参加者	大学生, 大学院生 6 人 (内 2 人が途中参加)
音声短縮率	0.66 (1.5 倍速)
利用メディア	音声, テキストチャット, ホワイトボード (WB), 拍手
途中参加者 1	会議開始から 46 分後に 90 秒のダイジェストを要求
途中参加者 2	会議開始から 43 分後に 180 秒のダイジェストを要求

表 2 各優先度抽出方式における実験での平均抽出率
Table 2 The average producing rate of each priority in the experiment.

	抽出方式	目標抽出率	実際の抽出率
優先度 1	一定間隔	20%	22.4%
優先度 2	中心人物 (結論)	25%	25.9%
優先度 3	中心人物 (問題提起)	25%	25.3%
優先度 4	残存発言	30%	26.6%

表 3 会議結果

Table 3 The results of the conference.

	発言数	発言時間 (秒)	拍手数	ホワイトボード書き込み数	CHAT 発言数	中心人物として選択された回数
User1 (調停役)	114	1,146	284	391	35	12
User2	51	357	309	75	45	3
User3	59	314	389	66	20	0
User4	39	245	304	169	19	0
途中参加者 1	17	97	0	0	11	0
途中参加者 2	13	136	1	15	0	0
総数	293	2,296	1,286	702	145	—
サイズ (KB)	9,185	—	45	229	11	—

作成された区間ブロック数と 1 区間ブロックあたりの平均発言数は以下ようになった。約 5 分間ごとに 1 つの区間ブロックが作成された。

- 区間ブロック数: 15 (優先度ブロック数: 60)
- 1 区間ブロックあたりの平均発言数: 19.5

4.2.3 途中参加者へのダイジェスト提供

図 9 に途中参加者 1, 2 のダイジェスト要求時において選択されたブロックを示す。この図から、スムーズな途中参加を考慮し、会議への参加直線付近の会議情報を重視してブロックが選択されていることが分かる。ユーザの要求より 7 秒多い 97 秒のダイジェストが選択されたが、これは 10% 程度の誤差であり許容範囲であると考えられる。

また、途中参加者 2 は会議開始から 43 分後に 180 秒のダイジェストを要求した。途中参加者 2 のダイジェスト要求時刻までの総発言時間は 1265 秒、総会議時間は 2,555 秒、発言存在率は 0.49 であり、結果として要求より 22 秒多いダイジェスト参照量となった。これも要求量と大きく違っているわけではない。ダイジェストの参照時間について

会議終了後に途中参加者にダイジェスト参照時間量について質問をしたところ、途中参加者 1, 2 両者ともに「特に気にならない」との意見であった。

しかしながら、180 秒のダイジェストを参照した途中参加者から「会議の要点を得ていたように思えるが、

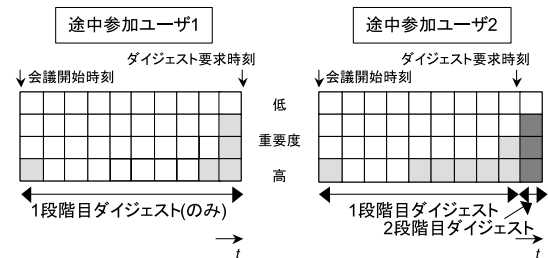


図 9 ダイジェスト要求時の選択ブロック
Fig. 9 Selected blocks in demanding digests.

音声のみを 180 秒も聞き続けるのは精神的につらい」との意見を得た。今回のダイジェスト参照実験では音声発言のみ提供しており、各音声発言に関連するホワイトボード描画やチャット、拍手などは再現していない。このため変化の乏しい画面を見つめながら音声のみを聞くという精神的苦痛と、参照中の発言が会議中で占める位置が不明であること、また、発言に関連する他発言との関係が明確にならないことが参照者にダイジェストへの不信感を与えていると考えられる。

今後はダイジェストの音声発言に関係するホワイトボード書き込みや、拍手、チャットなどを複合的に利用することで、会議内容理解を深める工夫が必要である。

ダイジェストの参照内容について

ダイジェストが内容的に問題ないかという質問をしたところ、特に 90 秒のダイジェスト参照者からは「内容はなかなか良い。会議の目的と現時点での話題が把握できた。会議参加がスムーズに行えた」という感想を得た。これは本方式の参加直前の会議内容を重視するという特徴が現れた結果である。

しかしながら、途中参加者が納得して聞いていたダイジェストの内容が、本当に会議の内容を反映するものであるとは断定しがたい。これを検証するために次節で述べるダイジェストの比較実験を行った。

4.3 ダイジェストの効果と信頼性の検証

以下の 2 点の確認を行うために実際に電子会議を行い、アンケートにより評価した。

(i) ダイジェストの効果

本研究で提示する「スムーズな」会議への途中参加が可能であるかを検証した。また、ダイジェストの提示方法による違いの検証をした。

(ii) ダイジェストの内容の信頼性

ダイジェストは実際の会議の一部の発言を抽出したものであり、実際の会議そのものとは異なる。ダイジェストを聞いてユーザが理解した内容が、実際の会話の内容を適切に反映しているかどうかを確認した。

4.3.1 実験方法

ブロック化ダイジェスト方式を組み込んだ DYNAMITE により電子会議を行った。会議の形式は特定のテーマを与えたフリーディスカッションである。会議は 4 回行い、時間はそれぞれ 40 分、参加者は 6 人である。それぞれの会議には調停者を 1 人指名した。

電子会議が終了した後、会議に 20 分遅れて参加するという想定で会議の参加者と非参加者にそれぞれ 120 秒のダイジェスト（会話再生速度：1.5 倍速）を参照してもらい、ダイジェスト参照後そのまま 2、3 分に

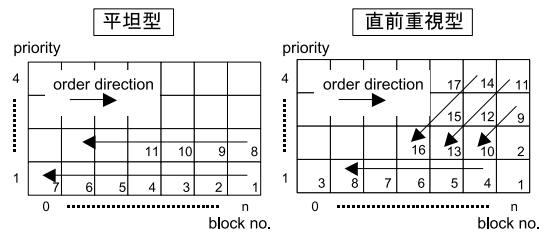


図 10 実験で利用したブロック選択方式
Fig. 10 Methods of selecting blocks used in the experiments.

わたり進行中の会議を参照してもらった。これはダイジェスト内容と参加直後の会議内容を比較してもらうためである。なお、ダイジェスト参照には参加者・非参加者ともに途中参加シミュレータを使用してもらった。

それぞれの会議ではダイジェストを 2 種類用意した（図 10）。平坦型ダイジェストはダイジェストブロックから平坦にブロックを選択したものであり、直前重視型ダイジェストは会議参加直前を重視してブロックを選択したものである。会議参加者は平坦型、直前重視型の両方を参照し、それぞれのダイジェストを参照後アンケートに記入した。非参加者は平坦型、直前重視型のどちらかを参照し、その後アンケートに記入した。なお、日程の都合により、会議参加者は会議終了から約 1 日後にダイジェストを参照した。参加者は前日の会議内容を忘れてしまう可能性があるため、会議ビューアを使用して会議発言を約 5 分間参照し（ダイジェストは利用しない）、前日の会議内容・会議進行を思い出した後、ダイジェストを参照し、アンケートに記入した。

4.3.2 アンケート内容

会議参加者および非参加者への共通の質問として以下の (1) から (5) を、会議参加者のみへの質問として (6) から (8) をそれぞれ準備し、各 5 段階評価で回答してもらった。

- (1) ダイジェストの音声発言を聞き取り可能であったか。
- (2) ダイジェストの音声発言間の「間(ま)」はどうだったか。
- (3) ダイジェストを聞いてこれまでの会議の流れがつかめるか [理由]。
- (4) ダイジェストの内容と会議参加直後の会話内容は関連しているか [理由]。
- (5) このダイジェストについて満足しているか。
(満足：分かった気になる、不満足：聞くだけ無駄)
- (6) ダイジェストの内容は途中までの会議内容と比較して違和感を感じるか [理由]。

また、平坦型、直前重視型両方に関連する質問として、

- (7) ダイジェストの参照時間はこれまでの会議内容を理

表 4 会議結果
Table 4 The result of the conferences.

	会議 1	会議 2	会議 3	会議 4
会議テーマ	乗物	住環境	乗物	住環境
会議形式	フリーディスカッション			
会議時間(分)	40			
参加者(名)	6		6	
総発言数	125	86	202	213
総発言時間(秒)	1638	1656	1140	1949
平均発言長(秒)	13.1	19.2	5.6	9.1
総拍手数	448	102	72	43
WB 書込	228	526	268	398
総チャット数	63	79	7	51

乗物：究極の乗物，住環境：究極の住環境，WB：ホワイトボード
うち 3 人が 20～25 分経過時の途中参加者

解するのに適当か [理由] .

- (8) 平坦型と直前重視型のどちらが良いか [理由] .
(この質問のみ「両方とも良い」「平坦型」「直前重視型」「どちらも良くない」の 4 項目から回答)

を用意した ([理由] は自由回答形式) .

4.3.3 会議結果と考察

会議の結果を表 4 に示し，アンケート結果の集計を図 11 に示す .

なお，アンケートは得点式・記述式を併用したが，被験者ごとに得点式の判断基準が大幅に異なる場合がある . このため，以下では主にコメントから得た内容を主眼に置き，得点式の結果を補助的なものにとらえて考察する .

会議の流れの把握

会議参加者・非参加者ともに，会議の流れはつかめたかという質問 (3) に対し，比較的「流れをつかめた」と返答した . しかしながら，コメントから判断すると「流れ」の定義を「会議全体 (不在期間) の流れ」と判断している被験者と「一連の会話の流れ」と判断している被験者に分かれた . 特に会議参加者のコメントでは，平坦型のダイジェストでは「不在期間の会話の流れ」がつかめたが，発言が断片化しており話題の飛び方が激しいとの意見を得た . これに対し，直前重視型のダイジェストは「一連の会話の流れ」が良いが，不在期間前半のほとんどが抽出されていないといった意見を得た . これは，今回の 120 秒という比較的短いダイジェスト要求量が原因と考える . 平坦型は薄く平坦にブロックを選択し，直前重視型は会議開始直後と，主に途中参加直前のブロックしか選択できなかったため，このような意見が得られたと考える .

ダイジェストと会議参加直後の会話との関連

参加者へのアンケートでは，ダイジェスト内容と会議参加直後の会話内容の関連に関する質問 (4) に対

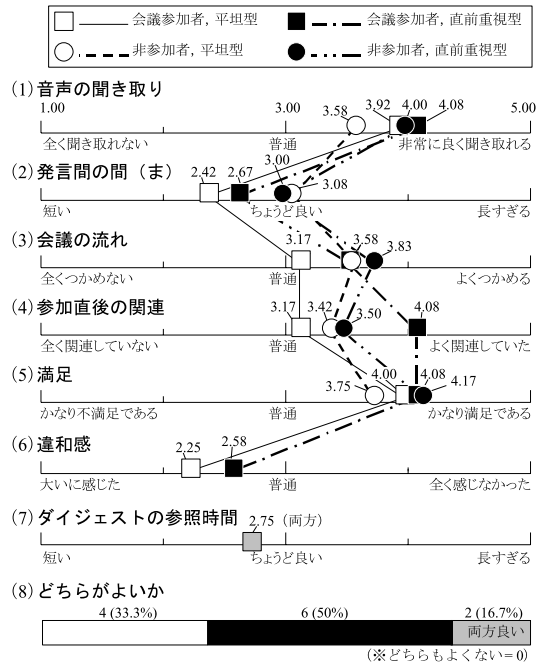


図 11 アンケート集計結果

Fig. 11 The result of the questionnaire.

し，直前重視型が若干関連度が高いという結果になった . 特に直前重視型に関するコメントでは「直前の会話がダイジェストに多く残っていて，前の話題と参加直後の関係がほぼそのままに保たれていた」「違和感無く続いていた」などの意見を多数得た . これは提案方式の特徴である参加直前の会話を重視するという点が現れた結果であり，ダイジェストと参加直後の話題がリンクしていた結果であると考え . これに対し，平坦型では参加直後の話題と深く関連してはいないものの，2 段階によるダイジェスト提供の効果により空白時間が短くなり，ある程度参加直前の話題が提供されていたため「関連は普通」との意見が多かった .

ダイジェスト内容と実際の会議との違い

会議参加者のダイジェスト内容と会議内容を比較しての違和感に関する質問 (6) に対しては，平坦型，直前重視型ともに違和感を感じる傾向にあった . 平坦型の違和感の理由として，質問，返答などの会話の流れがうまく再現されておらず，返答だけの発言がある場合があり，意味的に他の発言とつながらないため違和感を感じるとの意見を得た . また，直前重視型の違和感に関して述べると，参加直前の話題は良いが，不在期間の前半の会話内容が大幅に抜けており違和感を感じるとの意見を得た . これは，平坦型，直前重視型のブロック選択方式の違いが現れた結果であると考え .

平坦型，直前重視型の比較

会議参加者に対し，平坦型，直前重視型のどちらが良いかという質問(8)に対しては，会議ごとに意見が分かれ，どちらともいえない結果となった。これは個人の嗜好のほかに会議内容，会議の進行具合が大きく関連していると考えられる。たとえば会議2では全員が直前重視型が良いと解答しており，ダイジェストへの満足度も高く，違和感もそれほど感じていない。これは平坦型のような平坦なブロック選択方法では，i) 発言間の間隔が大きくなり，ダイジェストで選択された発言間の意味的な対応を把握しにくいことと，ii) 会議2のような個々の発言の長い会議では1つの発言の持つ意味が大きくなるため，ダイジェストによって発言が間引かれた場合，調停者からの質問と他者の解答が一致しなくなり，内容の理解の大きな妨げになることが原因と考えられる。一方，会議3のような個々の発言の短い会議では1つの発言の持つ意味は比較的小さく，ダイジェストによって発言が間引かれた場合でも内容の理解の大きな妨げになることは少ないため，平坦型と直前重視型の間に大きな差はみられなかった。

音声発言の聞き取り

質問(1)に多くの会議参加者または非参加者が音声発言を聞き取れたと返答した。1.5倍速という会話再生速度も問題なく聞き取り可能であったことが分かる。その一方で「ブツブツ音が気になる」とのコメントが多かった。「ブツブツ音」には2つの原因がある。1つ目は音声発言中にホワイトボード書き込みを行うと音声音が音飛びしてしまう現象に起因するものである。また，2つ目は音声再生(ストリーミング再生)の終わり際から2，3秒間だけ「ブツブツ」音が入る現象に起因するものである。多くの被験者から「気になる」とのコメントを得たが，特に非参加者からのコメントが多かった。非参加者は会議内容を知らないため，ノイズによって発言を逃してしまうとダイジェストの内容が途端に分からなくなってしまうことが理由と考える。

その他

音声発言に関連して表示されるホワイトボードがダイジェスト理解の大きな助けになるとのコメントを得た。しかしながら，これとは逆に「ホワイトボードの内容で混乱した」とのコメントも得ている。ホワイトボードは会議内容理解の助けとなるが，その見せ方を誤ると逆に誤解を招く危険性があることが確認できた。

4.3.4 結 論

結果としてアンケートからは，平坦型，直前重視型の優劣は一概にはつけにくいことが確認された。それ

ぞれのブロック選択方式における効果は一長一短であり，会議の種類，テーマ，参加者，会議の進行具合に大きく関係するといえる。平坦型では会議の内容を広く浅く知ることができるが，全体的なダイジェストの信頼性が比較的低く，ユーザは会話の内容が頻繁に変わる会議にスムーズに参加することが比較的困難である。逆に，直前重視型は参加直前の会議内容を重視するため，途中参加直前におけるダイジェストの信頼性が比較的高く，頻繁に会話が変わる会議であっても途中参加をスムーズに行いやすい。一般的に会議・会話は前半部分の会話よりも後半部分の会話の方が重要であるため，短時間で会議参加直前の内容をある程度把握するには直前重視型が良いであろう。

5. ま と め

本稿ではモバイル環境のための同期非同期型電子会議システム DYNAMITE 上におけるブロック化ダイジェスト方式の実現について述べ，本方式を利用した会議への途中参加実験およびダイジェストの比較実験・評価について述べた。

本方式は比較的頻繁に会話内容が変わる会議であっても，会議へのスムーズな途中参加支援のために有効であることが示された。しかしながら，ダイジェストの作成方法は会議の種類やユーザの好みに大きく依存するため一概には優劣がつけにくく，場合によって使い分けるのが良いことが明らかとなった。

今後の課題としては，会議の途中参加者に音声やホワイトボードだけをダイジェストとして提供するのではなく，ビデオやテキストなど他のメディアも何らかの形で関連づけて提供することが考えられる。また，関連している複数の発言が互いに別のブロックに分けられた場合の解決法も考慮する必要がある。

謝辞 本研究に快く話速変換モジュールを提供していただいた琉球大学の仲村貴大氏に，つつしんで感謝の意を表す。

参 考 文 献

- 1) 吉野 孝，宗森 純，伊藤士郎，長澤庸二：多人数対応電子会議システム DEMPO III の開発，情報処理学会論文誌，Vol.40, No.1, pp.150-160 (1999).
- 2) Munemori, J., Yoshino, T. and Yunokuchi, K.: Development of an Idea Collecting System and the Application to GUNGEN, *Proc. 4th International Conference on Knowledge-Based Intelligent Information Engineering Systems & Allied Technologies*, pp.237-240 (2000).

- 3) 井上裕子, 栗原主計, 才野 真, 重野 寛, 岡田謙一, 松下 温: 作業空間と会話空間を統合した仮想空間で共同作業を行う多地点遠隔会議システム: MAX, 日本バーチャル学会論文誌, TVRSJ Vol.4, No.2, pp.357-366 (1999).
- 4) Matsuura, N., Fujino, G., Okada, K. and Matsushita, Y.: Venus: A tele-communication environment to support awareness for informal interactions, *The Design of Computer Supported Cooperative Work and Groupware Systems*, pp.227-239 (1997).
- 5) Yamada, Y., Ohta, K. and Mizuno, T.: Extracting and viewing information method for mobile teleconference system, *Proc. Asia Pacific Computer Human Interaction*, pp.430-435 (1998).
- 6) 田中 充, 福宿光徳, 西堀良久, 勅使河原可海: 同期・非同期統合型マルチメディア会議システム ASSIST におけるマルチメディア議事録の開発と評価, DICOMO シンポジウム論文集, Vol.99, No.7, pp.79-84 (1999).
- 7) Kominek, J. and Kazman, R.: Accessing Multimedia through Concept Clustering, *Proc. ACM SIGCHI*, pp.19-26 (1997).
- 8) 加藤善大, 石原 進, 酒井三四郎, 水野忠則: 非同期電子会議へのスムーズな再参加を支援するための一方式, 情報処理学会モバイルコンピューティング研究会研究報告, Vol.99, No.80, pp.57-64 (1999).
- 9) <http://java.sun.com/products/java-media/jmf/>
- 10) <http://hb4.seikyoku.ne.jp/home/peervil/>

(平成 13 年 6 月 7 日受付)

(平成 13 年 9 月 12 日採録)



川口 明彦 (学生会員)

昭和 53 年生。平成 12 年静岡大学情報学部情報科学科卒業。現在、同大学大学院情報学研究科博士前期課程 (情報学専攻) 在学中。モバイルグループウェアに関する研究に従事。



加藤 善大

昭和 51 年生。平成 11 年静岡大学工学部知能情報工学科卒業。平成 13 年同大学大学院理工学研究科博士前期課程 (計算機工学専攻) 修了。在学中は電子会議システムに関する研究に従事。現在、サイボロニクスジャパン (株) に勤務し、FA 向け自律分散システムの研究・開発に従事。



石原 進 (正会員)

昭和 47 年生。平成 6 年名古屋大学工学部電気学科卒業。平成 11 年同大学大学院博士後期課程 (電子情報学専攻) 修了。平成 10 年度日本学術振興会特別研究員。平成 11 年静岡大学情報学部助手。現在、同大学工学部助教授。工学博士。モバイルコンピューティング, 無線環境用 TCP/IP に関する研究に従事。電子情報通信学会, ACM 各会員。平成 9 年度電気通信普及財団賞受賞。



酒井三四郎 (正会員)

昭和 31 年生。昭和 54 年静岡大学工学部情報工学科卒業。昭和 59 年同大学大学院電子科学研究科博士後期課程 (電子応用工学専攻) 修了。学習院大学, 新潟産業大学を経て, 平成 4 年静岡大学工学部助教授。現在, 同大学情報学部助教授。工学博士。ソフトウェア工学に関する研究・開発, 分散システムの設計・運用に従事。電子情報通信学会, 日本ソフトウェア科学会各会員。



水野 忠則 (正会員)

昭和 20 年生。昭和 43 年名古屋工業大学経営工学科卒業。同年三菱電機 (株) 入社。平成 5 年静岡大学工学部情報知識工学科教授。現在, 同大学情報学部情報科学科教授。工学博士。情報ネットワーク, モバイルコンピューティング, 放送コンピューティングに関する研究に従事。著書「プロトコル言語」(カットシステム)、「コンピュータネットワーク概論」(ピアソン・エデュケーション) 等。電子情報通信学会, IEEE, ACM 各会員。当会フェロー。