Vol.2019-GN-107 No.10
2019/3/19

動的な出力壁面選択を可能とするt-Room用音声出力制御法の開発

東　寛士[1]　和田　理[1]　片桐　滋[1]　大崎　美穂[1]

概要：コンピュータ支援共同作業用のシステムの一つ，t-Room の研究が継続的に行われている．t-Room は，モノリスと呼ばれる視聴覚メディア制御ユニットである壁によって部屋を構成し，遠隔地にあるその実装（部屋）を仮想的に重ね合わせることによって遠隔地における共同作業者に高い臨場感を提供することを目指している．本研究では，その既存の実装が持つ欠点，即ち映像と音像との不整合を解消するため，遠隔地 t-Room 内の音源位置に応じて動的に音像生成モノリスを選択できる t-Room 用音声通信方式を開発する．その実装においては特に，通信用計算機のコア数に応じて処理を並列化することで，通信処理に伴う遅延量の低減を目指す．また，音像生成モノリスの変更に伴うクリック雑音の発生を抑制するため，変更時に出力音をフェードイン・フェードアウトさせる機能も実装する．評価実験を通して，実装結果が設計通りに正確かつ低遅延（20ms 程度）で動作することを示す．

キーワード：遠隔共同作業支援システム，t-Room

Development of t-Room's Sound Output Control Method Dynamically Selecting Output-Target Monoliths

HIROSHI AZUMA[1]　OSAMU WADA[1]　SHIGERU KATAGIRI[1]　MIHO OHSAKI[1]

Abstract: One of the recent computer-supported cooperative work systems, t-Room, has been vigorously investigated. t-Room forms a room with multi-media walls called monoliths and aims at creating high realistic sensation for distant cooperative workers by virtually overlaying its distant implementations. To resolve the drawback of the existing implementation, mismatch between visual and acoustic images, we develop a new sound control method that dynamically selects a target monolith on which to generate a sound image, based on a sound source location in the t-Room. In particular, we implement the method, based on our computation thread design that maximizes the computational efficiency mainly determined by the usage of multiple (computer) cores. In addition, we implement sound fade-in/out functions in a process that switches monoliths to avoid harsh clicking noise and unnatural sound pressure changes. Our experimental evaluations show that our implemented method naturally switches sound locations among monoliths and operates only with about 20ms of processing delay.

Keywords: Remote cooperative work support system, t-Room

1. はじめに

音声・映像通信を用いて遠隔地どうしの空間を仮想的に重ね合わせ，その利用者に高臨場感を提供することを目指す [1][2][3] 遠隔共同作業支援システム「t-Room」[4] が提案されて以来，その様々な改良が進められている（例：[5]）．t-Room は，スピーカー・カメラ・等身大ディスプレイからなる視聴覚メディア制御ユニット（モノリスと呼ばれる）を壁面と見立て，複数のモノリスによって空間を構成する．原則として接続される t-Room は同形状であり，音声・映像のデータは，仮想的に重ね合わされるモノリス

[1] 同志社大学大学院
Graduate School of Science and Engineering, Doshisha University

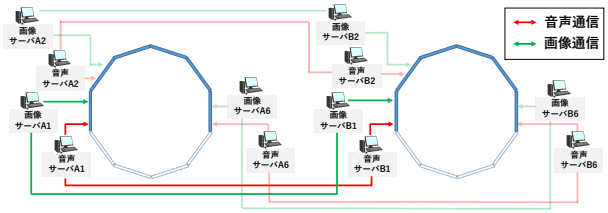

図 1 t-Room における音声・映像通信.
Fig. 1 Sound/image data transmission in t-Room.

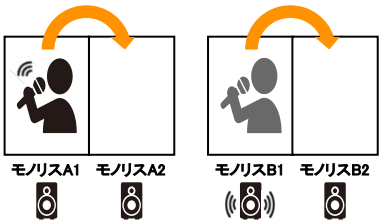

図 2 音源移動時の壁面の例.
Fig. 2 Mismatch between visual and sound images.

どうしの間で送受信されてきた（**図 1**）．　空間を重ね合わせるという t-Room の特徴は，カメラとディスプレイを対峙するように設置することによって実現される．しかしこの設置法は，空間を共有できるという際立った特徴をもたらす一方で，不適切な映像生成を避けるために利用者の位置をモノリス付近に制約し，空間の十分な利用を困難にもしてきた．また，等身大ディスプレイを壁とする構成は，スピーカー出力音の反射を大きくし，音響的エコーの問題を深刻にもしてきた．そして，こうした問題を解決するため，映像データに関しては，（撮影および映像再生の対象である）映像オブジェクトの抽出とその変換によって利用者位置の制約を緩和する試み [6] や，利用者がピンマイクを携帯することで音響的エコーを回避する試みなどが導入されてきた．

確かにピンマイクの利用は，収音ゲインを調整することで比較的容易に音響的エコーを抑制し，音声再生における深刻な質の低下の回避を可能とする．しかし，ピンマイクの位置，あるいはそれを携帯する利用者などの音源位置に応じて出力音の再生位置，即ち音像位置も適切に移動できなければ，映像と音像との乖離を引き起こし，臨場感を低下させる（**図 2**）．こうした状況の中で我々は，ピンマイクを用いる音声[*1]入力を前提として，その移動（位置の変化）に伴って，出力スピーカ，言い換えれば音像生成モノリスを切り替える機能の実現を目指してきた．本稿では，その機能を実現するための新しい音声通信方式の概要と，t-Room 用に開発されてきた既存の音声通信用のサーバ [7] を基盤としてその方式を実装した新しいサーバ群の動作検証結果を報告する．

[*1] ここで用いられる音データはヒトが発する音声とは限らないが，テレビなどの出力音を音声と呼ぶ習慣に準じてこの用語を用いる．

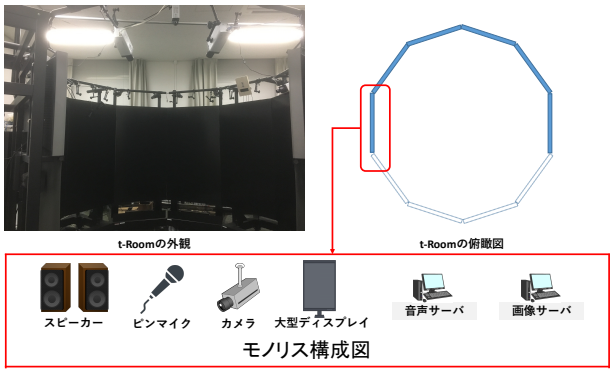

図 3 t-Room のハードウェア構成例.
Fig. 3 An example of device configuration for t-Room.

2. t-Room

本節では，t-Room システムの概要について述べる．t-Room[4] はモノリスと呼ばれるスピーカー・カメラ・大型ディスプレイによる視聴覚メディア制御ユニットを構成要素とし，対応する入出力・通信用サーバによって制御されている．以降の t-Room の構成は**図 3** のように，正十角形の 6 辺にモノリスを壁面と見立てて配置させ，t-Room 中央の俯瞰位置に t-Room 空間の俯瞰画像を撮影するためのカメラを設置しているものとする．

t-Room を制御する各サーバは**図 4** のように，階層構造 [5][8] を形成している．メディア制御層のサーバがモノリス毎の視聴覚メディア入出力・通信などを行い，メディア制御補助層の各サーバがメディア制御層の補助や，遠隔地との通信を行う．また，これらのサーバを地点毎に t-Room として管理する t-Room 制御サーバや，地点間の t-Room の接続補助などを行うセッション制御サーバを用いる．

本研究における提案音声通信方式に関連して動作するサーバは音声サーバ，音声入力制御サーバ，音声出力制御サーバ，オブジェクト抽出サーバであるため，以降ではこれらのサーバについて述べる．

3. 提案音声通信方式

開発当初以来，現在に至るまで用いられてきた t-Room 用音声通信方式では，前述のように人物などの音源移動を伴う共同作業において音像と映像が異なる壁面において生成される可能性があった．これにより，共同作業の臨場感を低減させる可能性があった．

この問題を改善させるため，本研究ではまず初めに，人物などの音源位置に応じて動的に音声の出力モノリスを選択し，選択されたモノリスの音声サーバへ音声を伝送する方式を提案する．特に，この方式の実現のため，音声出力制御サーバを新たに導入する．出力モノリスの選択は，音声出力スピーカーの切り替えを意味する．従って，単に音

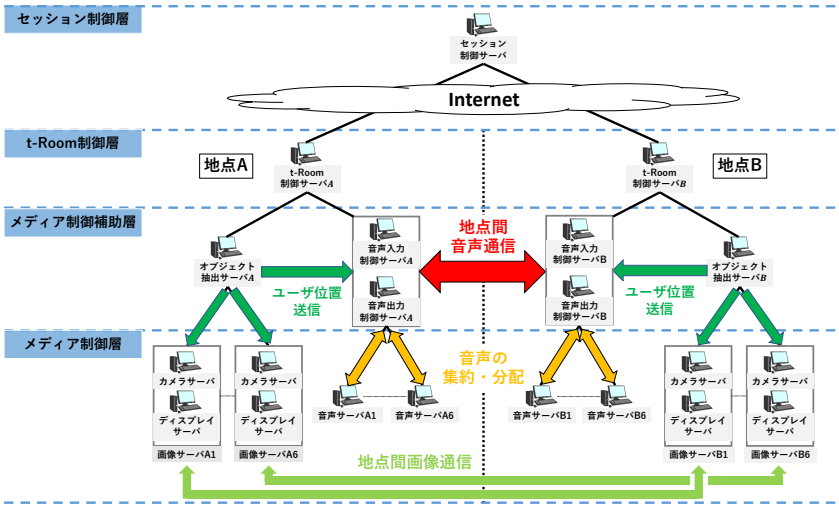

図 4 t-Room を制御するサーバによる階層構造.
Fig. 4 Hierarchical structure of servers in t-Room.

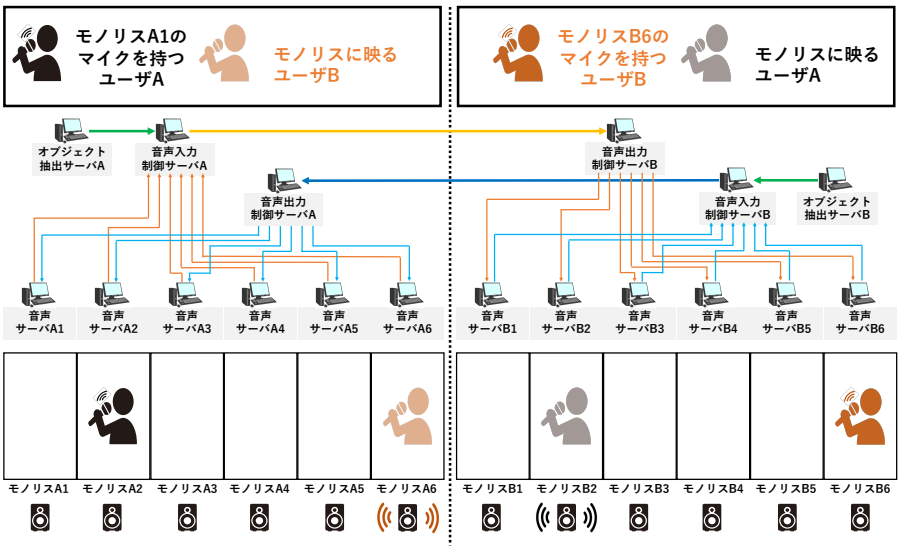

図 5 提案音声通信方式に用いるサーバ構成例.
Fig. 5 Server configuration of proposed sound transmission method for t-Room.

声データのオン・オフ的な切り替えを行うと切り替え時にクリック雑音を発生させてしまう恐れがある．従って提案音声通信方式では，このクリック雑音を抑制させるために，音声の出力モノリスが変更される際に出力音声のフェードイン/アウトを行うような実装を行なう．この実装のため，ある地点で入力された音声を一旦接続先の全てのモノリスへ送信し，音像を生成すべき位置に応じてモノリスあるいは出力スピーカーの選択を制御する仕様とした．

また，先行研究の t-Room の音声通信用サーバ [7][9][10] ではマルチスレッドによって音声の入出力や送受信を並列処理で行うような実装をしてきた．このマルチスレッドは通信相手のモノリスの数によって変化するが，本実装を進めるにあたり人物数が増えるにつれて音声がモノリスに入力されてから遠隔地のモノリスで音声が出力されるまでにかかる遅延時間が増大する問題が生じることがわかった．この問題については，通信自体にかかる不可避な遅延時間を短縮することは困難であるが，マルチスレッドをサーバのコアに割り当てることによって通信以外の処理にかかる遅延時間の短縮を行う．

本提案音声通信方式のサーバ構成を**図 5** に示す．ここからは実装した提案音声通信方式に用いるサーバの処理について具体的に述べる．

4. 提案音声通信方式の実装

4.1 オブジェクト抽出サーバ

t-Room 空間の中心付近に人物などの物体が存在する場合，地点内の全てのモノリスのカメラから同一の物体の映像が入力され，遠隔地の全てのモノリスのディスプレイから同一の物体の映像が出力される可能性があった．この問題を解決するために，オブジェクト抽出サーバの開発が進

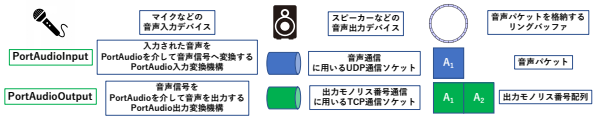

図 6 サーバ内部構成図に用いるアイコンの説明.

Fig. 6 Explanation of icons used to illustrate t-Room's server structure/behavior.

められている [6]．オブジェクト抽出サーバは，t-Room の俯瞰位置に設置されたカメラで得た t-Room 空間の俯瞰画像から人物などの物体の位置を推定する．推定された物体位置から画像サーバが映像を出力するべきモノリスを選択し，物体が単一のモノリスのディスプレイから出力される．

提案音声通信方式では，人物などの音源位置に応じて音声が出力されるモノリスを選択するため，オブジェクト抽出サーバによって推定された人物などの物体の位置を利用し，音声を出力するモノリスの番号（以降では音声出力モノリス番号と記す）を決定する．

本来は上記のように音源位置を推定し，音声を出力するモノリスを決定するが，後述の動作検証実験を効率的に行うため，一定時間毎に音声出力モノリス番号を変更する仕様に変更した．動作確認実験では，この仕様変更されたオブジェクト抽出サーバを用いた．

4.2 音声サーバ

まず**図 6** に，サーバの構成を図解する際に用いるアイコンをまとめて紹介する．以降の図解は，このアイコンを用いて行う．音声サーバは，先行研究で開発された t-Room の音声通信・入出力処理を行う音声伝送サーバ [7] を改変したサーバである．先行音声伝送サーバは各モノリスにおける音声の入出力を遠隔地の対応するモノリスとの間で音声の入力モノリスと出力モノリスを固定して音声パケットの送受信を行なっていた．本研究で開発した音声サーバは音声の入出力処理，音声入力制御サーバへの音声パケットの送信，音声出力制御サーバから音声パケットの受信を行う．各音声サーバ毎に 1 つのマイクなどの音声入力デバイスの音声の入力処理を行うため，地点毎に音声サーバの数だけ人物は共同作業に参加することが可能である．

ここでの音声パケットには以下の要素が格納されている．また，音声信号は標本周波数 44.1kHz，標本数 128 の音声信号が格納された short 型配列である．

- 音声信号.
- 音声入力モノリス番号.
- 音声出力モノリス番号.
- 音声入力時刻.

音声サーバの内部構成例を**図 7** の音声サーバ A1 で示す．音声サーバ A は以下のスレッドを持つ．

- 音声入力・送信スレッド.
- 音声受信スレッド.

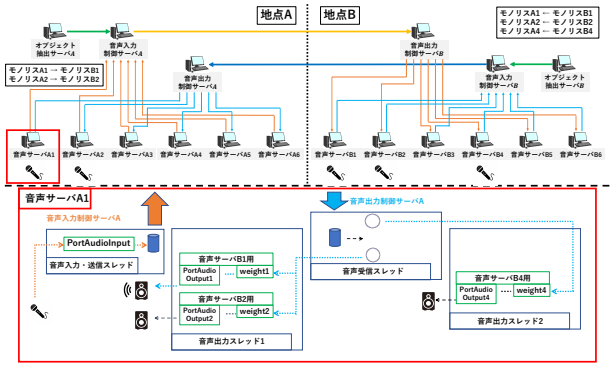

図 7 音声サーバの内部構成例.

Fig. 7 Inner structure of t-Room's sound server.

- 音声出力スレッド.

音声入力・送信スレッドにおいて，モノリス A1 のピンマイクなどの音声入力デバイスから入力された音声を PortAudio 入力機構によって音声信号に変換し，音声入力モノリス番号，音声入力時刻とともに音声パケットに格納し，音声入力制御サーバ A の対応する音声受信スレッドへ音声パケットを送信する．

音声受信スレッドにおいて，音声出力制御サーバ A の音声サーバ A1 用音声送信スレッドから音声パケットを受信し，音声入力モノリス番号に応じて対応するリングバッファへ音声パケットを格納する．このリングバッファは，地点 B の人物が音声を入力しているモノリスの半数毎に作成する．この場合，モノリス B1，モノリス B2 用のリングバッファ，モノリス B4 用のリングバッファを作成する．

音声出力スレッドにおいて，音声受信スレッドのリングバッファ毎の音声の出力処理を行う．この場合，音声出力スレッド 1 でモノリス B1，モノリス B2 で作成された音声パケットに格納された音声信号の出力処理を行い，音声出力スレッド 2 でモノリス B4 で作成された音声パケットに格納された音声信号の出力処理を行う．出力処理を行う際，この音声サーバのモノリス番号と音声パケットの音声出力モノリス番号が一致する音声信号のみ出力し，一致しない場合には信号値 0 の音声信号を PortAudio 出力機構によって出力する．

音声出力モノリス番号が変更され，音声が出力されなくなる場合には 1 から 0 へ単調に減少する重み係数を，一定数の音声パケットの音声信号値にかけて出力する．逆に音声が出力開始する場合には 0 から 1 へ単調に増加する重み係数を一定数の音声パケットの音声パケットの音声信号値にかけた音声信号を出力する．モノリス B1 で**図 8** のような波形の音声を入力している人物がモノリス B1 付近からモノリス B3 付近へ移動する場合を考える．この場合，音声の出力モノリスはモノリス A1 からモノリス A3 へ変更される．このとき音声をフェードイン/アウトさせることによって，**図 9** の時刻 10[ms] のように急激に音圧変化が

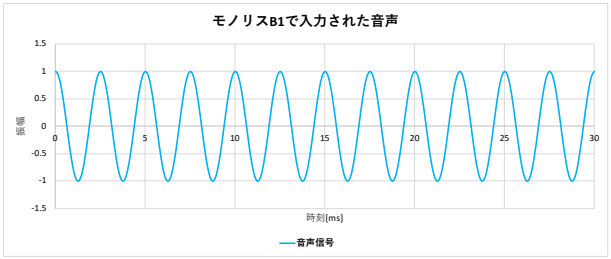

図 8 モノリス B1 で入力された音声の波形例.
Fig. 8 An example of sound wave input by monolith B1.

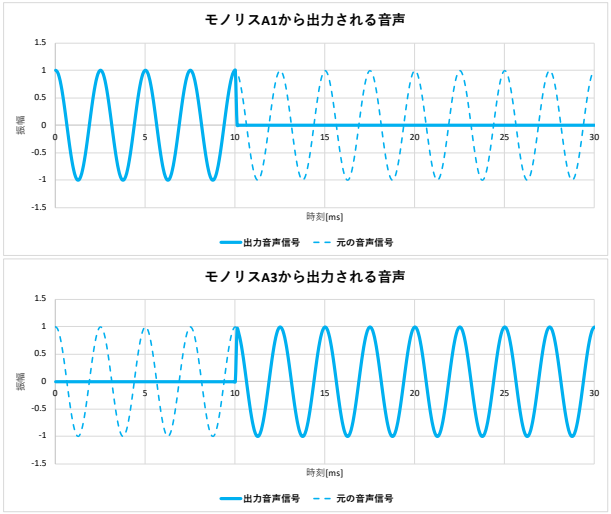

図 9 フェードイン/アウトなしの出力音声の波形例.
Fig. 9 An example of output sound wave without fade-in/out function.

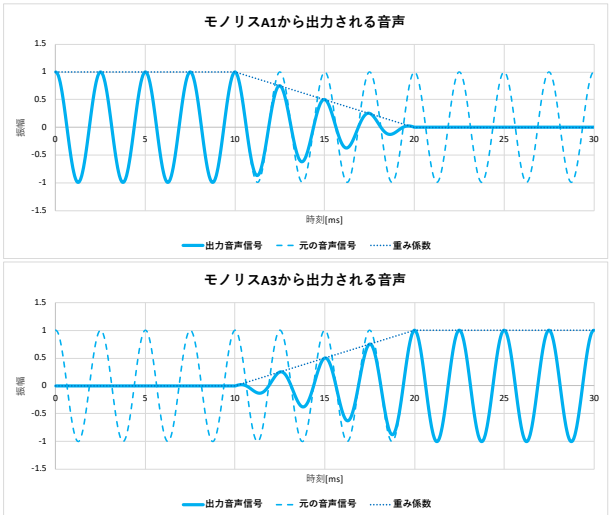

図 10 フェードイン/アウトありの出力音声の波形例.
Fig. 10 An example of output sound wave with fade-in/out function.

生じる可能性があった音声の出力が，**図 10** のように滑らかに音圧変化させることが可能となり，前述のクリック雑音の抑制を可能とした．

また，実際の移動を伴う共同作業を想定すると，フェードインが完了する前に異なるモノリスへ移動する場合や，フェードアウトが完了する前に再度移動前のモノリスへ人物が移動する場合も考えられる．この場合，その時点でのフェードインに用いていた重み係数の値を引き継いで 0 へ短調に減少する重み係数を設定することで，フェードインを中断してフェードアウトを行うことを可能としている．フェードアウトの場合も同様である．

また，音声サーバは Mac OS 上で開発しており，音声の入出力は PortAudio 入出力機構において音声処理ライブラリ PortAudio と CoreAudio によって行う．出力時には PortAudio 出力機構を複数作成することによって多チャンネル音声を 1 チャンネル音声として合成させて出力する．

音声サーバは論理コア数 4 の計算機上で動作させることを想定して，これらのスレッドの総数が 4 以下で作成するよう実装した．具体的に，その音声サーバのピンマイクなどの音声入力デバイスを使用する人物が存在する場合に音声入力・送信スレッドを 1 つ作成し，そうでない場合は作成しない．音声受信スレッドは遠隔地の音声サーバのピンマイクなどの音声入力デバイスを使用する人物が 1 人以上存在する場合に 1 つ作成し，そうでない場合は作成しない．音声出力スレッドは遠隔地の音声サーバのピンマイクなどの音声入力デバイスを使用する人物が 2 人以下存在する場合にその人数分作成し，そうでない場合は 2 つ作成する．これによって各スレッドを各コアに割り当て，処理効率の最大化を図っている．

4.3 音声入力制御サーバ

音声入力制御サーバは本研究において新設した提案サーバである．地点内の人物が音声を入力している全ての音声サーバから受信した音声パケット毎に，オブジェクト抽出サーバから受信した音声出力モノリス番号を格納し，遠隔地の音声出力制御サーバへ送信する．音声入力制御サーバの内部構成例を**図 11** の音声入力制御サーバ A で示す．音声入力制御サーバ A は以下のスレッドを持つ．

- 出力モノリス番号受信スレッド．
- 地点 A の音声サーバ用音声受信スレッド．
- 音声出力制御サーバ B 用音声送信スレッド．

出力モノリス番号受信スレッドにおいて，オブジェクト抽出サーバ A で音声出力モノリス番号が変更された際に TCP 通信によって音声出力モノリス番号配列を受信し，このスレッドが持つ音声出力モノリス番号配列に格納する．

地点 A の音声サーバ用音声受信スレッドのそれぞれにおいて，対応する地点 A の音声サーバから音声パケットを受信し，このスレッドが持つリングバッファに格納する．

音声出力制御サーバ B 用音声送信スレッドにおいて，地点 A の各音声サーバ用音声受信スレッドのリングバッファから音声パケットを取り出し，出力モノリス番号受信スレッドの音声出力モノリス番号配列から得た対応する音声出力モノリス番号を音声パケットに格納し，音声出力制御

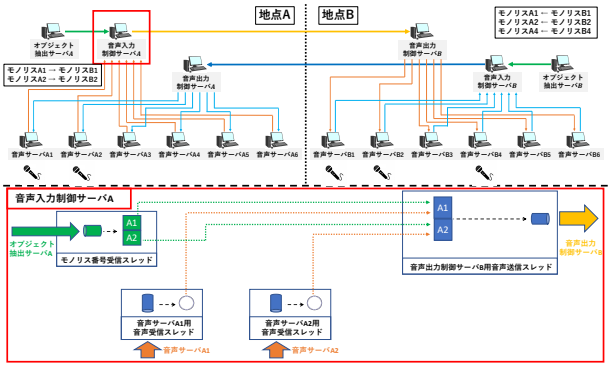

図 11 音声入力制御サーバの内部構成例.

Fig. 11 Inner structure of proposed sound input control server.

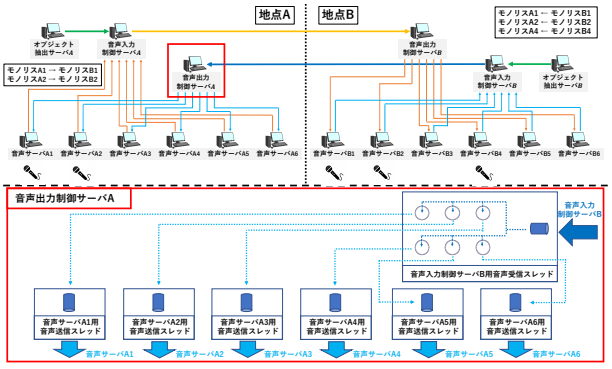

図 12 音声出力制御サーバの内部構成例.

Fig. 12 Inner structure of proposed sound output control server.

サーバへ順次送信する.

音声入力制御サーバは論理コア数 8 の計算機上で動作させることを想定して，これらのスレッドの総数が 8 以下で作成するよう実装した．具体的に，地点内の音声サーバのピンマイクなどの音声入力デバイスを使用する人物が存在する場合に出力モノリス番号受信スレッドを 1 つ作成し，そうでない場合は作成しない．地点内の音声サーバ用音声受信スレッドは，ピンマイクなどの音声入力デバイスを使用する人物が存在する地点内の音声サーバ毎に作成し，人物が存在しない場合は作成しない．遠隔地の音声出力制御サーバ用音声送信スレッドは，地点内の音声サーバ用音声受信スレッドが作成される場合に 1 つ作成し，そうでない場合は作成しない．これによって各スレッドを各コアに割り当て，処理効率の最大化を図っている.

4.4 音声出力制御サーバ

音声出力制御サーバは音声入力制御サーバと同様に，本研究において新設した提案サーバである．遠隔地の音声入力制御サーバから受信した遠隔地の各音声サーバで作成された音声パケットを，音声パケット毎に地点内の全ての音声サーバへ送信する．音声出力制御サーバの内部構成例を**図 12** の音声出力制御サーバ A で示す．音声出力制御サーバ A は以下のスレッドを持つ.

- 音声入力制御サーバ B 用音声受信スレッド.
- 地点 A の音声サーバ用音声送信スレッド.

音声入力制御サーバ B 用音声受信スレッドにおいて，音声入力制御サーバ B から受信した各音声パケットを，入力モノリス番号に応じて地点 A の音声サーバ毎に用意したリングバッファに格納する.

地点 A の音声サーバ用音声送信スレッドにおいて，音声入力制御サーバ B 用音声受信スレッドの対応するリングバッファから音声パケットを取り出し，地点 A の対応する音声サーバに音声パケットを送信する.

音声出力制御サーバは論理コア数 8 の計算機上で動作させることを想定して，これらのスレッドの総数が 7 以下で作成するよう実装した．具体的に，遠隔地の音声サーバのピンマイクなどの音声入力デバイスを使用する人物が存在する場合に遠隔地の音声入力制御サーバ用音声受信スレッドを 1 つ作成し，そうでない場合は作成しない．地点内の音声サーバ用音声送信スレッドは，地点内の音声サーバ毎に作成し，遠隔地に人物が存在しない場合は作成しない．これによって各スレッドを各コアに割り当て，処理効率の最大化を図っている.

5. 動作確認

5.1 実験概要

提案音声通信方式を用いた音声通信によって，最大の負荷となる状況においても意図した通りに動作が行えているかを以下の項目から確認する.

(1) フェードイン/アウトが行えているか.
(2) 音声の出力モノリス切り替えにかかった遅延.
(3) 通信処理遅延が許容範囲内であるか.

上記の確認項目について，(1) は遠隔地で出力された音声の波形を目視し，また試聴することによってクリック雑音の抑制が行えているかを確認する．(2) は人物移動によって音声が出力されるモノリスが変更された場合を想定し，音声出力モノリス番号を変更した時刻から実際に音声が出力される音声サーバでフェードイン/アウトを開始するまでにかかった遅延時間を確認する．(3) は音声がモノリスに入力されてから遠隔地のモノリスで出力されるまでにかかった遅延時間である．ITU-T の勧告 G.114[11] によると，多くの音声通信アプリケーションにおいてほとんどの利用人物が 0〜150[ms] の遅延時間を許容することが可能とされている．各サーバの処理効率最大化を図った実装によって，この許容範囲を達成しているか確認する.

上記 3 項目を確認するにあたり，最大負荷とは各地点にモノリスが 6 台ずつ存在し，かつ共同作業の参加者がそれぞれ最大の 6 人が存在する場合を意味する．また，この動作確認では波形の確認や，効率的にデータ集計を行うために，人物が各モノリスのピンマイクから音声を入力する

表 1 入力音声.
Table 1 Input sound sources.

音声	パターン 1	パターン 2
音声 A1，音声 B1	440Hz 矩形波	440Hz 矩形波
音声 A2，音声 B2	660Hz 矩形波	無音声
音声 A3，音声 B3	880Hz 矩形波	無音声
音声 A4，音声 B4	1100Hz 矩形波	無音声
音声 A5，音声 B5	1320Hz 矩形波	無音声
音声 A6，音声 B6	1540Hz 矩形波	無音声

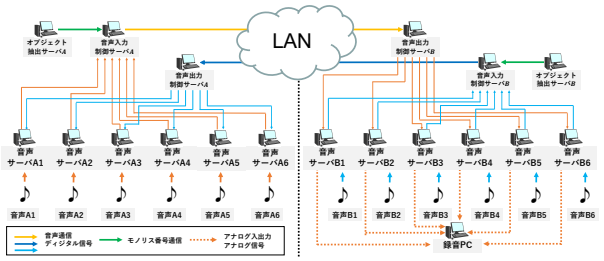

図 13 動作確認のサーバ構成.
Fig. 13 Server configuration used in our operation evaluation experiments.

のではなく，各音声サーバに対して予め用意した異なる音声を入力音声としている．音声を入力する際，各音声サーバの計算機上の音声再生ソフトで再生した音声を，仮想オーディオ入力インタフェースの Loopback Audio を用いてディジタル入力している．入力音声は**表 1** の 2 パターンの音声を用いた．

動作確認のサーバ構成を**図 13** に示す．地点 A から地点 B へ伝送される音声についての動作確認の流れは以下の通りであり，地点 B から地点 A についても同様である．この方法で 1 時間動作させた場合の各項目の集計を行なった．

(1) 地点 A の各音声サーバの計算機上で音声 A1〜音声 A6 を再生する．
(2) 地点 A の各音声サーバで入力音声と入力時刻を音声パケットに格納し，音声入力制御サーバ A へ送信する．
(3) オブジェクト抽出サーバ A で 5[s] 毎に音声出力モノリス番号を変更し，音声入力制御サーバ A へ送信する．
(4) 音声入力制御サーバ A で地点 A の各音声サーバから受信した音声パケットに，音声出力モノリス番号を格納する．
(5) 音声入力制御サーバ A から音声出力制御サーバ B へ音声パケットを送信する．
(6) 音声出力制御サーバ B で受信した各音声パケットを地点 B の全ての音声サーバへ送信する．
(7) 地点 B の各音声サーバで音声パケットを受信する．
(8) 受信した音声パケットの音声の入力時刻と出力時刻の時刻差を，通信処理遅延としてログに出力する．
(9) 受信した音声パケットの音声出力モノリス変更時刻と，実際にフェードイン/アウトを開始する時刻差を音声の出力モノリス切り替えにかかった遅延としてログに出力する．
(10) 地点 B の各音声サーバから音声を 1 チャンネル音声としてライン出力する．
(11) 録音 PC で全ての 1 チャンネル音声をまとめた 6 チャンネル音声をライン入力で録音する．
(12) 録音音声の波形を目視，また試聴してフェードイン/アウトが行えているか確認する．

表 2 音声サーバの諸元.
Table 2 Main specifications of sound server.

OS	macOS Sierra 10.12.6
CPU	Intel Core i7 @ 3GHz
論理コア数	4
メモリ	8GB
NTP デーモン	chronyd (chrony) version 3.4
オーディオインタフェース	Loopback Audio
再生ソフトウェア	iTunes

表 3 音声入力制御サーバ，音声出力制御サーバ，オブジェクト抽出サーバの諸元.
Table 3 Main specifications of sound input control server, sound output control server, and object detection server.

OS	CentOS 7.6.1800
CPU	Intel Core i7-4790 @ 3.60GHz
論理コア数	8
メモリ	7.6GB
NTP デーモン	chronyd (chrony) version 3.2

表 4 NTP サーバの諸元.
Table 4 Main specifications of NTP server.

OS	CentOS 7.6.1810
CPU	Intel Core i7-8700K @ 3.70GHz
論理コア数	12
メモリ	16GB
NTP デーモン	chronyd (chrony) version 3.2

上記の動作確認について，LAN 環境において NTP デーモンとして chrony を用いたサーバ間の時刻同期を行なった際に，1[ms] 以下の誤差で時刻同期が行えることが確認されている [12]．そのため，全てのサーバは共通の NTP サーバと時刻同期しているものとし，入力時刻などはサーバの絶対時刻を用いた．各サーバなどの諸元を**表 2**，**表 3**，**表 4**，**表 5** に示す．

5.2 確認結果

(1) フェードイン/アウトが行えているか

表 1 のパターン 2 の入力音声を用いた場合の録音音声の波形を**図 14**，**図 15** に示す．図の縦軸は振幅，横軸は時刻 [s] を表す．音声の出力モノリス切り替え時に音声のフェードイン/アウトが行えていることが確認できる．パターン

表 5　録音用計算機の諸元.
Table 5　Main specifications of computer used to record output sounds.

OS	macOS Sierra 10.12.6
CPU	Intel Core i7 @ 3GHz
論理コア数	4
メモリ	8GB
オーディオ入力インタフェース	MOTU 16A
録音ソフトウェア	Sound eXchange

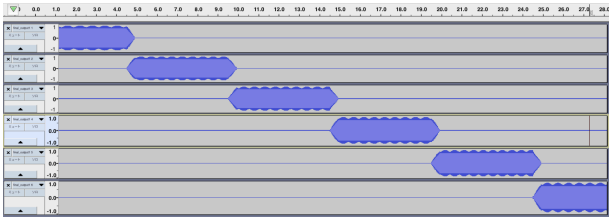

図 14　録音音声の波形.
Fig. 14　Examples of recorded sound waveforms.

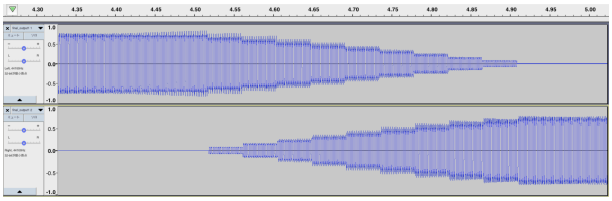

図 15　図 14 の拡大図.
Fig. 15　Enlarged view of Fig. 16.

表 6　音声の出力モノリス切り替えにかかった遅延.
Table 6　Delay observed in switching sound output Monoliths.

標本平均 [ms]	3.42128
標本標準偏差 [ms]	2.11106
最大値 [ms]	8

1 の音声を用いた録音音声を試聴してみたが，クリック雑音は発生しておらず，意図した通りにクリック雑音の抑制に成功した．

(2) 音声の出力モノリス切り替えにかかった遅延

音声の出力モノリス切り替えにかかった遅延を**表 6** に示す．標本平均，標本標準偏差から安定して切り替えが行えていると考えられる．

(3) 通信処理遅延

通信処理遅延を**表 7** に示す．前述の許容遅延を達成しており，かつ標本標準偏差は 1[ms] 未満であることから安定して低遅延の音声通信が行えていることが確認できた．

6. おわりに

本研究では，t-Room において人物などの音源が移動を伴う共同作業を行う場合においても音像と映像が生成される壁面を一致させ，遠隔共同作業の臨場感向上を目的とした．その上で，音声の出力モノリス切り替え時にクリック

表 7　通信処理遅延.
Table 7　Delay observed in sound data transmission and processing.

標本平均 [ms]	0.702565
標本標準偏差 [ms]	0.806515
最大値 [ms]	22

雑音を抑制させるために出力音声がフェードイン/アウトするような実装も行った．本提案手法を用いて，一定時間毎に出力モノリスを変更させて動作確認を行い，これらの実装機能が設計通りに正確かつ低遅延 (20ms 程度) で動作することを確認した．しかし，他研究 [6] で開発が進められているオブジェクト抽出サーバのように，ユーザ位置の推定によって音声の出力モノリスを変更する仕様で検証を行えていない．よって，今後の課題として他研究のオブジェクト抽出サーバと連携した上で，ユーザが音声の出力モノリス切り替えにかかる遅延を許容可能か，評価実験する必要がある．

参考文献

[1] Harrison, S.; ”Media Space 20+ Years of Mediated Life”, Springer Publishing Company, Inc. (2009).
[2] Fuchs, H., State, A. and Bazin, J.-C.; ”Immersive 3D Telepresence”, IEEE Computer Magazine,Vol. 47, No. 7, pp. 46-52 (2014).
[3] Cesar, P., Kaiser, R. and Ursu, M.; ”Toward Connected Shared Experiences”, IEEE Computer Magazine, Vol. 47, No. 7, pp. 86-89 (2014).
[4] Keiji Hirata, Yasunori Harada, Toshihiro Takada, Shigemi Aoyagi, Yoshinari Shirai, Naomi Yamashita, and Junji Yamato; ”The t-Room - Toward the Future Phone”, NTT Technical Review, Vol. 4, No. 12, pp. 26-33(2006).
[5] 福岡篤志, 和田理, 片桐滋, 大崎美穂; ”Linux 版 t-Room における実システム環境を用いた動作確認”, 2017 年度 情報処理学会関西支部 支部大会，(2017).
[6] 和田理，片桐滋，大崎美穂; ”t-Room における俯瞰カメラを用いた映像出力壁面選択機構”, 情報処理学会研究報告, Vol. 2017-GN-101, No. 28, (2017).
[7] 松尾雄真, 片桐滋, 大崎美穂 ; ”Mac OS のためのローカル・ラグ制御機能をもつ音声伝送サーバの実装と性能評価”, 2016 年度　情報処理学会関西支部　支部大会　講演論文集, (2016).
[8] 荻野裕也, 杉本直也, 片桐 滋, 大崎美穂; ”Linux 版 t-Room の開発：デバイス制御システムの設計と実装”, 情報処理学会研究報告, Vol. 2014-GN-90, No. 13, (2014).
[9] 竹森幸輝, 前田佳奈, 岩原正典, 片桐 滋, 大崎美穂; ”ローカル・ラグ制御機能とログ同期機能を持つ音響サーバの開発”, 情報処理学会研究報告, Vol. 2012-AVM-76, No. 3, (2012).
[10] 中谷彰皓, 片桐 滋, 大崎美穂; ”高臨場感を与える視聴覚ディスプレイのための音像制御システム”, 情報処理学会研究報告, Vol. 2015-GN-94, No. 23, (2015).
[11] ”One Way Transmission Time”, ITU-T Recommendation G. 114, (1996).
[12] 藤田 佳樹, 片桐 滋, 大崎 美穂; ”NTP を用いる複数 Monolith 間同期の実現”, 第 79 回全国大会講演論文集, Vol. 2017, No. 1, pp. 381-382, (2017).