

多数決制を取り入れた

リアルタイムネットワークセッション

— Improvisession から Democracyrant へ —

中村 文隆, 長嶋 洋一

神戸山手女子短期大学

ネットワーク上での即興セッション支援システム“Improvisession”を発展させた“Democracyrant”システムを設計し、RMCP(Remote Media Control Protocol)を用いて Windows NT 上に実装して実験を行った。セッションの音楽構造として小節構造を持たず、プレーヤーごとに異なった時間での繰り返しを基礎としたミニマルミュージック的なシステムを採用し、また、プレーヤー間のインタラクションの方法として多数決、または専制によるフレーズの消去のシステムを取り入れた。この形式は遅延の大きなネットワーク上でのセッションを実現する目的にも有効である。

Real time network session support system

With a majority decision

— Improvisession to Democracyrant —

Fumitaka Nakamura, Yoichi Nagashima

Kobe Yamate College

We developed a network session support system “Democracyrant” based on “Improvisession” system. The concept of this system is a “minimal music server” on the network. Players submit incoherent musical sequences to the server, and votes to them. The sequences will be erased by a majority decision by the players. This type of a network session will overcome the difficulty of large delay on the wide area network.

1 はじめに

我々はこれまでに、ネットワークで相互接続された Indy ワークステーション群を用いた即興セッション演奏支援システム Improvisession(長嶋、中村、片寄、井口,1997)を開発し、実験を行っている。Improvisession は GUI として OpenGL/Motif を、音楽データの伝送に早稲田大学の後藤らによる RMCP (Remote Music Control Protocol) [2]を用い、マウスによる Graphical な操作を主体とした「演奏」を行ってセッションを実現するシステムであった。

ネットワークを通じた音楽セッションの試みは、後藤らによるジャズセッションシステムなどがあり、更に、遠隔地間では避け得ない通信遅延に対する試みも提案されている[8]。

後藤らが指摘しているように、物理的な距離が増大すると原理的に除去し得ない遅延が発生する。

情報伝達速度は光速を超えることができないため、伝達に使用する情報処理機器の処理時間が仮に 0 であっても 2 点間の距離を光速で割った時間だけの遅延は必ず発生する。加えて、現実には通信に使用する機器の処理時間、ネットワークのバンド幅の問題を無視することができず、国内でも WAN 間通信では数十 ms から 100ms 程度の遅延が常識的である。

こうした問題に対し、後藤らは「RMCP: Remote Media Control Protocol ——時間管理機能の拡張と遅延を考慮した遠隔地間の合奏——」[9]において、予め決められたコード進行の繰り返しに基づいたセッション形態=Remote GIG を提案している。上述のように遠隔地間では情報の遅延を避けることは原理的に不可能であるから、このように音楽構造の方で遅延を吸収しうるような形態を考案していくことは不可避的であると思われ、その点で Remote GIG の提案は興味深いものである。

今回我々は、同じ問題に対する解答のバリエーションとして Democracyrant¹を提案する。このシステムは各プレーヤーから一定のシーケンスを受け取り、それを繰り返し演奏する Democracyrant サーバと、プレーヤーが操作する Democracyrant クライアントの2つからなる。

本論文では Democracyrant の構想について説明し、実際に Windows NT 上に実装したものをを用いて神戸山手女子短期大学音楽科の講義²において行った実験について報告する。

システム開発にあたっては、MIDI 情報とセッション情報伝送に RMCP Version 3 を中村が Windows NT 用に移植したものをを用い、ユーザーインターフェース部分は MFC による GUI を用いた。

2 Democracyrant システム

2-1 Democracyrant の構成

Democracyrant はセッション情報を収集・管理し、告知するサーバと、プレーヤーが操作し個々のセッション情報を送信するクライアントからなる。クライアントの数は標準的な MIDI 機器で利用可能な最大数 16 を上限としている。

セッション情報の詳細については 2-2 で述べるが、Democracyrant 特有のセッション情報を RMCP の exclusive メッセージで、MIDI の演奏情報を RMCP の smidi メッセージで送信している。

図 1 に全体のシステム構成を示す。Democracyrant サーバは、UDP の broadcast が届く範囲内に 1 つだけ存在しなければならない。WAN 間の接続については TCP パケットを用いた中継機構を用意する必要

¹ Democracyrant は、Democracy と Tyrant からなる造語である。この名称は、セッションの発展を決定するために用いられる方式として、多数決と独裁を併用することに由来している。

² 神戸山手女子短期大学音楽科における「デスクトップミュージック I、II」(中村)。従来の「コンピュータと音楽」(長嶋)に続き、音楽科のコンピュータミュージック講義として今年度から開講された。

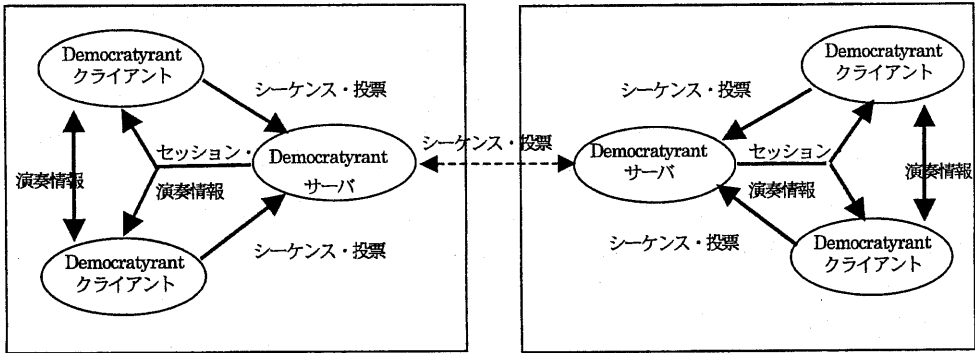


図1 : Democracyrant システム概念図

があるが、現在のバージョンでは未実装である。RCMP Gateway の利用も考えられるが、Democracyrant のコンセプト上(2-2-2 参照)、RCMP exclusive メッセージのみを選択的に中継させる必要がある。

2-2 Democracyrant パケット

Democracyrant パケットは

- 1) DC_Sequence
- 2) DC_Broadcast
- 3) DC_Join
- 4) DC_Vote

の4種類で、RCMP exclusive メッセージとしてカプセル化され、broadcast される。それぞれのパケットの内容と用途を次に述べる。なお、以下では Democracyrant を DC と略記する。

2-2-1 DC_Sequence パケット

DC_Sequence パケットは最大 20 音までの MIDI ノート・ナンバー、MIDI ベロシティ、MIDI チャンネル、音長 (ms 単位)、繰り返し時の間隔 (ms 単位) を含み、DC クライアントから送信される。

DC サーバはこのパケットを受け取り、該当する MIDI チャンネルに参加者登録があるかを調べ、エンタリがあれば登録する。

登録されたシーケンス情報は DC サーバのシーケンス部で解析され、サーバから RCMP smidi パケットとして送信される。DC クライアントはこの情報を受けて MIDI 音源を鳴らす。DC サーバ自身は MIDI 情報は捨てている。

2-2-2 DC_Broadcast パケット

DC_Broadcast パケットは現在の参加者数、参加者氏名リスト、チャンネル毎の音色情報、現在のセッションリーダー、占有チャンネルリスト、投票集計情報を含み、DC サーバから送信される。このパケットは1秒ごとに送信され、情報をアップデートする。DC クライアントはこれを受け取り、現在のセッション情報をプレーヤーに提示する。また、DC クライアントは起動時に 3 秒間 DC_Broadcast パケットを待ち、その間に DC_Broadcast パケットがこなければ DC サーバが起動していないものと判断して異常終了する。待ち時間の間に DC_Broadcast を受信すれば、占有チャンネルリストの情報からプレーヤーが参加可能なチャンネルを割り出し、提示する。

特殊な使い方として、参加者数を負の数に設定した DC_Broadcast パケットは参加者情報の再送信を DC クライアントに促す「リセット」パケットとして働く。DC クライアントは「リセット」パケットを受け

取った場合にはDC_Join パケットに参加要求情報を設定して再送信しなければならない。Democracyrant セッションを WAN 間に拡大する場合、サーバ間で中継されるのは DC_Broadcast パケットのみである。

2-2-3 DC_Join パケット

DC_Join パケットはプレーヤーの名前、占有したい MIDI チャンネル、送信したホストの IP アドレスの情報を含む。DC クライアントから送出される。DC サーバはこのパケットを受信し、占有したい MIDI チャンネルが占有されていない場合は参加者情報を更新する。占有したい MIDI チャンネルが既に使用されていれば更新は行われない。更新が行われた場合、所定のオフセットを占有 MIDI チャンネルに加算した DC_Join パケットが ACK として DC サーバから送出される。

DC クライアントは起動時に DC_Broadcast パケットを待ち、空きチャンネルを取得してプレーヤーに提示する。2 人以上のプレーヤーが同時に同じチャンネルで参加要求を出した場合、先にパケットが到着した方がチャンネルを占有し、ACK 情報によって登録完了を知る。DC クライアント側では、登録情報送信後 5 秒間 DC_Broadcast と DC_Join パケットを監視し、登録依頼したプレーヤー名がセッション情報に含まれていることと、自分の IP アドレスが入った ACK オフセット付き DC_Join パケットを受信したことを持って登録完了とみなし、セッションに入る。条件が満たされなければ新たに DC_Broadcast パケットを受信し、その情報をもとに再度登録作業に入る。

DC クライアントはまた、終了時に所定のオフセットを加算した MIDI チャンネルを持った DC_Join パケットを送信し、セッションから抜けることを宣言する。DC サーバはこれを受けてセッション情報を更新する。

2-2-4 DC_Vote パケット

DC_Vote パケットは投票者 MIDI チャンネル、投票対象 MIDI チャンネル、意見 (Opinion)、の情報を含み、DC クライアントから送信される。DC サーバはこれを受信し、票数を集計する。現在の実装では Opinion はシーケンスに対するリコールのみで、特定のチャンネルに対するリコール票がその時点で参加者数の過半数を上回るとそのチャンネルのシーケンス情報は消去される。これが Democratic なシーケンス消去である。

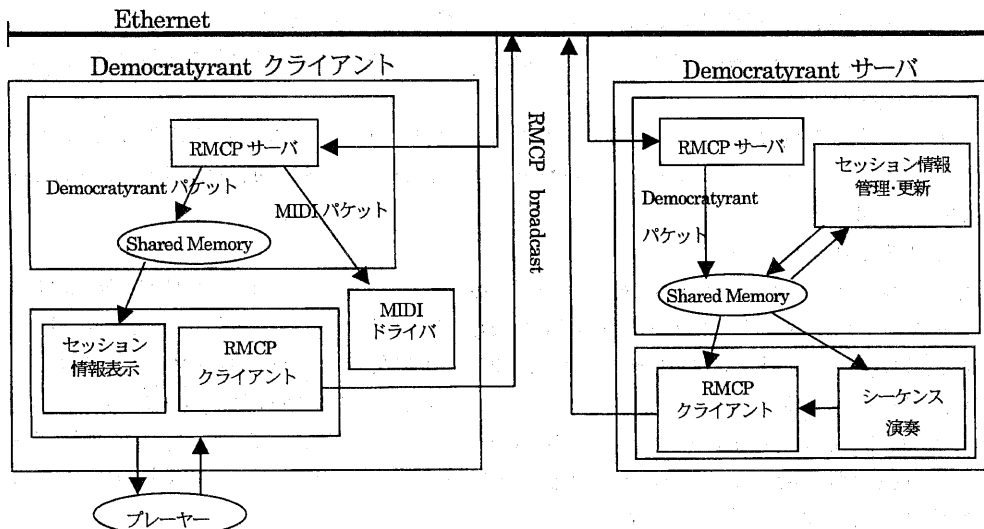


図 2 : Democracyrant のデータの流れ

特殊なケースとして、DC サーバで乱数を用いて更新されるセッションリーダーからリコール票が届いた場合には無条件で対象チャンネルのシーケンスが消去される。これが Tyrant によるシーケンス消去である。この意味で、セッションリーダーのことを Tyrant とも呼ぶ。

セッションリーダー＝Tyrant の決定は DC サーバで行い、10 秒ごとに乱数を発生させ、1/3 の確率でリーダーの更新が行われる。更新も乱数を用い、参加者の中から等確率で新しいリーダーが選ばれる。この情報は DC_Broadcast パケットを通じて各プレーヤーに送信される。

以上が Democracyrant で使用されるパケットである。図 2 にデータの流れを示す。DC サーバ、クライアントとも、送信と受信を別個のプログラムで担当し、両者の間の情報共有には shared memory を用いている。

2-3 Democracyrant サーバ

図 3 に Democracyrant サーバの操作画面を示す。サーバの機能及び役割は

- 1) クライアントからの参加要求・脱退要求を受け付け、参加者情報を管理する
- 2) 必要に応じて参加者情報のリセットを行う
- 3) クライアントからのシーケンス登録を受け付け、登録されているシーケンスを演奏する
- 4) セッションリーダーを選出する
- 5) クライアントからのリコール票を受け付け、シーケンスの削除を管理する
- 6) 全てのシーケンス情報を一括削除する
- 7) 特定プレーヤーのシーケンスを削除する

の 7 つである。

このうち、6) と 7) は機能上、Tyrant 的な性格をもつが、6) はむしろ全体に対するインザッツのような意味合いで装備されている機能であり、セッションの 1 クールにおいては通常使用しないのが望ましい。7) についてもクライアント PC 全体のハングアップのように、不可避的な事故に対処するための機能である。

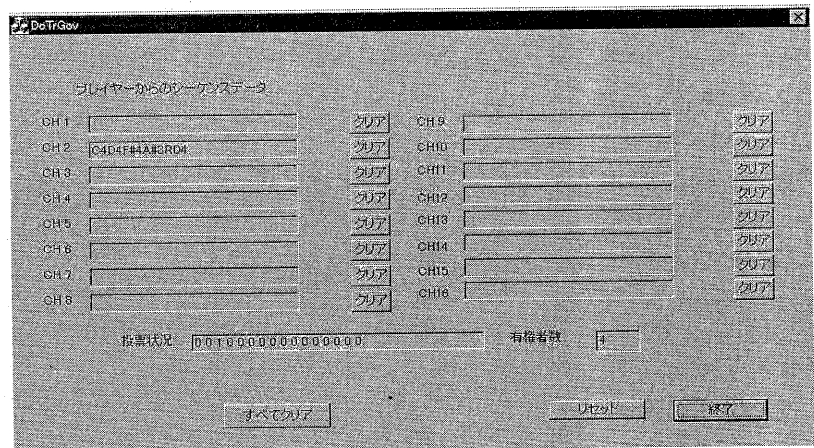


図 3 : Democracyrant サーバ画面

2-4 Democracyrant クライアント

各プレーヤーは Democracyrant クライアントを起動し、DC サーバからの DC_Broadcast 情報に基づいて参加可能なチャンネルを選択する。図 4 が起動時の登録画面であり、この画面でプレーヤーが参加可能なチャンネルと音色（サーバ側で指定され、固定）を選択し、プレーヤーの名前を入力して登録を行う。プレーヤー同士の識別は選択した音色によってなされる。

セッションへの参加登録が完了すると、図5のセッション画面を用いてセッションに参加する。クライアントでの操作は

1. 一定間隔で繰り返されるシーケンス情報の編集・サブミット
2. 各シーケンス情報への投票
3. マウス操作によるリアルタイム演奏情報の3種類である。

これらのうち、3は LAN 内でのみリアルタイムの「合いの手」として RMCP MIDI パケットで送信されるべき情報である。。このリアルタイム演奏情報に関しては遅延に関する対策をしないため、セッションを WAN 間に拡張する場合にはこのパケットが中継されてはならない。

「セッション情報」のセクションは、現在のセッション参加者数、プレーヤーの参加チャンネル、現在のセッションリーダー、の情報が表示されている。「あなたのチャンネル」と「セッションリーダー」

が等しい時、そのプレーヤーはセッションリーダー = Tyrant であり、その間、「参加者一覧」セクション中の [やだ!] ボタンをクリックするとその音色のシーケンスがサーバから消去される。「セッションリーダー」が自分のチャンネルと異なる場合は通常の投票操作となり、参加者の過半数がリコール票を投じたシーケンスが削除される。

「シーケンスフレーズ」セクションでは、スライダーを用いて音高を設定し、[追加]、[休符]、[クリア]の各ボタンで「あなたのフレーズ」欄にシーケンスを登録していく。間隔、ペロシティ、繰り返し間

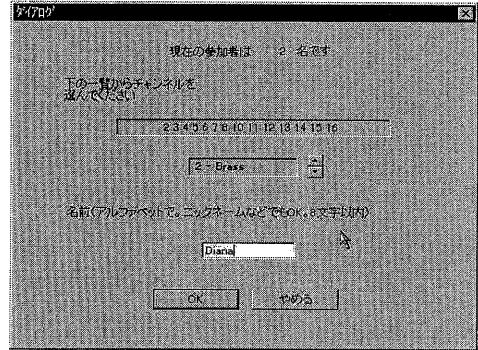


図4 : DemocracyTyrant クライアント
起動時登録画面

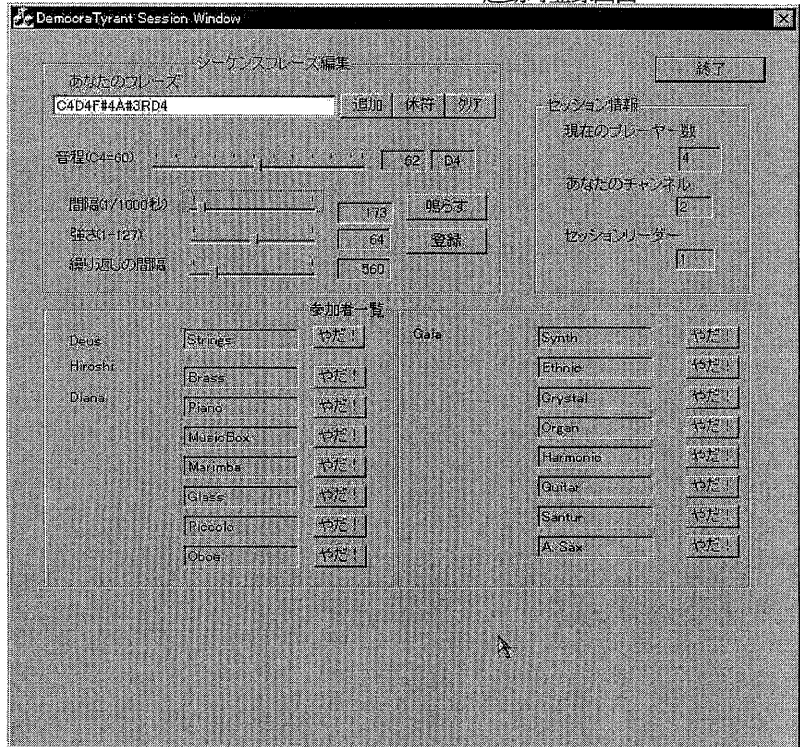


図5 : DemocracyTyrant クライアント
セッション画面

隔の各スライダーでそれぞれの値を設定し、[鳴らす]ボタンで一度だけ、[登録]ボタンでサーバに登録して繰り返し演奏をさせることができる。

シーケンス情報のパラメータは

- 音程 (0 ~ 127)
- 間隔 (30ms ~ 3000ms)
- 強さ (0 ~ 127)
- シーケンスとシーケンスの間隔 (30ms ~ 3000ms)

の4つである。

また、ウィンドウ下部の空き領域でマウスをドラッグすると左→右の方向に音高が上がり、上→下の方向にペロシティが上がる。この領域を使ってリアルタイム演奏を行う。

2-5 Democracyrant セッション

各ユーザーはお互いの出したシーケンスフレーズの「合奏」を聞き、次に出すべきシーケンスを考案し、登録を更新する。並行して、消したいと感じたフレーズにリコールの投票を行う。更に、マウス操作による「リアルタイム演奏」も行い、これらを総合したものが Democracyrant セッションとなる。

Remote GIG との主な相違点は、(1) 小節という概念を用いず、ユーザー毎に異なる周期を基本単位とすること、(2) 他のユーザーとのインタラクションの形式として、投票によるシーケンスのリコールシステムをもつこと、である。

ここで、(1) については、音楽の形式としてミニマル・ミュージックをイメージの基礎においている。周期の異なるフレーズ群を繰り返すことにより生じるズレやうねりを楽しもう、という発想である。

また、(2) については民主主義のパロディ的な色彩が濃いのが、通常のセッションにおける、聴衆や他のプレーヤーの反応、という部分がある程度取り込むための試みである。

3 実験結果

続いて2回に渡って行われた Democracyrant セッションの様子を報告する。

3-1 第一回・音楽科一年生 8名

第一回の実験は1998年7月8日、中村が担当する「デスクトップミュージック I」の講義を受講する学生を対象に行われた。参加した学生は神戸山手女子短期大学音楽科の1年生8名で、大半がピアノ専攻、フルート、声楽、作曲の専攻がそれぞれ1名ずついた。

この実験では最初のうち、自分がレッスンで練習しているクラシック曲のテーマをシーケンスとして打ち込む学生が続出した。しかし、クライアント起動時の音高、間隔のパラメータが毎回同一で、ほとんどのプレーヤーが初期値の間隔を使用したため、フレーズの「ずれ」によるミニマムの効果を得ることが難しかった。また、提出されるフレーズもダイアトニックなものが多く、それほど面白い演奏効果が得られなかったのが残念である。

リコール機能については、参加者同士が顔見知りであったこともあって、間違えた自分のフレーズに対して「私の音色の[やだ!]を押して!!」という音声によるブロードキャストメッセージでリコールが行われる、という場面もあった。全般に気心が知れた間柄であるためか、比較的リコールや Tyrant の機能を使用してフレーズの削除が行われていたセッションであった。

3-2 第二回・雑多な参加者

第二回の実験は、1998年7月10日、音楽科卒業生2名、国文学科2年生1名、教養学科1年生2名、情報処理担当教員1名、それに中村、の6名で行われた。これらのうち、教養学科1年生と情報処理担

当教員は音楽に関してはほぼ素人であり、国文学科の学生はバンドでトロンボーンを吹き、レコーディングも経験している学生であった。

第一回の実験の反省に基づき、クライアント起動時には音高、間隔をランダムに設定するようにした。これによって初期値のままでもフレーズの「ずれ」が生じることが期待されたが、今回の参加者のうち教養学科1回生2名を除く4名は「ノリ」がよく、積極的に間隔を変更してフレーズを作成したため結果としては面白いというねりを聴くことができた。この時の参加者は既成曲のフレーズを打ち込むことはほとんどなく、半音進行やオクターブ以上の跳躍など、「変わった」フレーズを作成することに熱中していた様子であった。しかしその分、他のプレーヤーのフレーズに耳を傾けることが少なく、自分の「面白い」フレーズ作成に没頭した感があり、従って[やだ!]ボタンや Tyrant 機能の使用頻度も少なかった。個々の作成したフレーズは特長のあるものであったが、他者の演奏に耳を傾けなければ面白いセッションはできない、ということが実感された実験でもあった。

4 まとめと今後の課題

我々は今回、Improvisession の発展形としての Democracyrant システムを設計し、Windows NT 上に実装して実験を行った。Democracyrant 自体は遅延の大きなネットワーク経由のセッションを視野に入れたシステムであるが、今回はコンセプトを確認するため LAN 内でのセッション実験とし、その結果、ネットワーク経由でのミニマルミュージック的セッションを支援するシステムとして Democracyrant は有効に機能することが確認できた。

今後の課題として、RMCP Gateway などを用いた WAN 間でのセッション機能の実装、一音ごとの強弱・不等同隔などシーケンス編集機能の拡充、ユーザー間のコミュニケーション手段として文字によるメッセージ送信機能の追加を考えている。引続き開発と実験を重ね、報告を行いたい。

参考文献

- [1] 長嶋洋一, 中村文隆, 稲松千奈美, 渡辺卓也: マルチメディア・ワークステーションによる情報基礎教育の試み. 情報処理学会平成6年度後期全国大会講演論文集I, pp.5-6, 1994.
- [2] 後藤真孝, 橋本祐司: MIDI 制御のための分散協調システム ---遠隔地間の合奏を目指して--- 情報処理学会研究報告 Vol.93, No.109 (93-MUS-4), pp.1-8, 1993.
- [3] 長嶋洋一, 片寄晴弘, 由良泰人, 藤田泰成, 井口征士: マルチメディア生成系におけるプロセス間情報交換モデルの検討. 情報処理学会研究報告 Vol.95, No.74 (95-MUS-11), pp.63-70, 1995.
- [4] 長嶋洋一, 中村文隆, 後藤真孝, 片寄晴弘, 井口征士: ネットワーク上で相互作用するアルゴリズム作曲系を用いた音楽教育システム. 情報処理学会平成9年度前期全国大会講演論文集II, pp.273-274, 1997.
- [5] 長嶋洋一: Multimedia パフォーマンス作品"Muromachi". 京都芸術短期大学紀要 [瓜生] 第17号 1994年, pp.39-43, 1995.
- [7] 長嶋洋一: 音群技法による音楽作品のための演奏支援システム. 情報処理学会平成2年度後期全国大会講演論文集I, pp.253-254, 1990.
- [8] 後藤真孝, 日高伊佐夫, 松本英明, 黒田洋介, 村岡洋一: すべてのプレーヤーが対等なジャズセッションシステム I. システムの全体構想と分散環境での実現. 情報処理学会研究報告 Vol.96, No.19(96-MUS-14), pp.21-28, 1996
- [9] 後藤真孝, 根山亮, 菊池淑晃, 村岡洋一: RMCP: Remote Media Control Protocol - 時間管理機能の拡張と遅延を考慮した遠隔地間の合奏. 情報処理学会研究報告 Vol.97, No.67(97-MUS-21), pp.13-20
- [10] 長嶋洋一, 中村文隆, 片寄晴弘, 井口征士: "Improvisession" ネットワークを利用した即興演奏支援システム. 情報処理学会研究報告 Vol.97, No.67(97-MUS-21), pp. 25-30, 1997