

解説

計算機による音楽演奏†



田口友康†

1. はじめに

機械仕掛けによる自動演奏装置はヨーロッパで過去数百年にわたって製作され、今もかの地の博物館に数多く保存されている。それらの中には美しい音色で音楽を奏するだけでなく、人形の精巧な動作や美術工芸的な美しさで人目を楽ませるものも多い。他方、前世紀末から一時期流行した自動ピアノのように、今日のレコードプレーヤに相当する装置もあった(Bowers¹⁾, Roehl²⁾, 梶谷³⁾, 高澤・三好⁴⁾。これらの装置の「演奏データ」は金属の円筒や円盤、時代を下っては紙ロールに記録されたもので、装置と一体か、あるいは掛けかえ型であっても装置の製作者から提供されるものであった。そこでこのような過去の自動演奏装置はオートマトン型(オートマトン=自動人形)と呼んでよかろう。

計算機を用いた音楽演奏は1950年代に始まった。そしてハードウェア、ソフトウェアの技術と電気音響技術の進歩の上に、計算機援用の音楽演奏は急速に拡大してきた。その特徴はいうまでもなくソフトウェアによる駆動であり、物理的には演奏データの大量かつ高速な蓄積・検索ならびに編集・加工の機能である。これらの機能を用いて

実演奏データの即時取り込みとその編集
多楽器、多音源の演奏制御
記号的データの入力による演奏
新規な楽音、非楽音の創出
疑似音場の形成

人と機械との相互交渉環境の提供、などが現在活発に研究されつつある。このような今日の自動演奏はインテリジェンス型と呼べよう。以下では計算機援用の音楽演奏を計算機演奏と呼ぶことにする。はじめに音楽演奏と楽譜の問題にふれよう。

2. 演奏と楽譜

1948年にP. Shaefferによって創出されたミュージック・コンクレート(musique concrète)は、共通の記述手段としての「楽譜」に表せない音楽といつてよいであろう。その音楽は自然界の音(風、波、騒音、人声など)を加工して音の造形とするもので、音世界におけるモンタージュである。加工の手段は、初期においては録音テープの逆送り、素材音の部分的な切りだし、残響の付加、テープ送り速度の変動によるピッチ昇降、多重録音などであった。

第二次大戦後に重要な意味をもちはじめた電子音楽(electronic music)はK. Stockhausenらを事実上の先駆者として展開してきたが、ミュージック・コンクレートとは対照的に、純粋に電気音響的手段による音が用いられた。そうすることの芸術的理由は決して一つではなく、作曲者により、また同一人の作品の間でも違うこともあると思われるが、少なくとも初期における一つの思想は、音楽的思考と音楽的素材の統合であった。固定的な音階から離脱し、音色や音強のプロフィールを広範かつ絶対的に精密に制御することによって「音のスペクトルを作曲する」ことはその手段であり目的でもあった。必然的に作曲と演奏が不可分な音楽となるわけで、楽譜を使うとしてもその意味は伝統的なものとは質を異にするものであろう。

この二つの音楽ジャンルの作品は、どちらも録音テープの形にでき上がるので、しばしばテープ音楽と呼ばれてきた。テープ音楽の技術的基盤は音の電気的加工であるから、計算機と電気音響技術の進歩にとともに、計算機を内包した作曲手段へ傾斜するのは自然の成り行きである。デジタル信号処理技術の浸透とともに、素材音が自然音であるか人工音であるかの違いは技術的に意味をもたなくなった。こうして音源のあらゆる可能性のもとで音そのものを作曲するという分野が計算機音楽の一つの勢力となった。J.-C. Risset⁵⁾の次の言葉はこの状況をよく表している：いく人かの

† Music Performance with Computer by Tomoyasu TAGUTI
(Department of Applied Mathematics, Faculty of Science,
Konan University).

† 甲南大学理学部応用数学科

MIDI ユーザによると、楽音合成の問題は解決されたか、副次的である。C. Yavelov にとって、音設計が作曲に対するのは字体設計がテキスト処理に対するようなものである。したがって今や彼らは作曲の問題に専念できる。しかし、J.-B. Barrière が強調しているように、音の制作は作曲なのであって、なお多くの研究が必要である……。(注: Risset は楽音合成研究の先駆者の一人であり、また最近では 1986 年にオランダのハーグで開かれた第 12 回国際計算機音楽会議に作品 *Sud* (南フランスの海岸で採取した自然音をもとにした官能的なコラージュ) を出展して賞賛を得た。Barrière も同系統の作曲家。一方 Yavelov は MIDI シンセサイザをパーソナル・コンピュータで制御する音楽システムを製作、宣伝することに精力的に関与している。)

さて、伝統的な音楽の場合、演奏家は楽譜をどのように音化しているのだろうか。五線譜に表された音楽と演奏楽音との差違は、1920~1930 年代にアイオワ州立大学の Seashore⁶⁾ と彼の共同研究者たちによってはじめて分析された。彼らは、声楽とバイオリン演奏におけるピッチ変動やビブラートの速さと深さ、ならびにピアノを含めての演奏速度のゆれなどのようすを実測によって示した。Seashore の思想は次に要約される: 楽譜は不完全であり、譜面どおりの演奏では機械的、非芸術的な音楽しか聞かれない。「音楽芸術の媒体は、主として、固定した規則正しい固い音程や、均一な強さや、固定したリズムや、純音や、完全なハーモニーなどからの芸術的な逸脱 (artistic deviation) にある。」(訳文は梅本⁷⁾ からの引用。) これ以後、この種の分析は音楽心理学における一つの主題となり、最近では Shaffer⁸⁾、Gabrielsson⁹⁾、Clarke¹⁰⁾、池内¹¹⁾、小川¹²⁾、中島¹³⁾、太田¹⁴⁾、¹⁵⁾ などが演奏における記譜音価からのずれを分析している。これらは演奏者がどのように音を作るかを調べるものだが、一方では、このような楽譜からのずれが聴き手にどう聴かれるかという問題がある。これには中村・難波¹⁶⁾、Sloboda¹⁷⁾、Gabrielsson¹⁸⁾、Senju-Ohgushi¹⁹⁾ などの研究がある。森²⁰⁾ は、等時間的発音が必ずしもイン・テンポ演奏とは聴かれないことを報告した。(以上についてのやや詳しい解説が田口²¹⁾ にある。) Sundberg²²⁾ は規則による表情づけを研究している。最近の成書に Deutsch²³⁾ がある。

どのような機式の計算機音楽でも、機械的でない計算機演奏を実現するためには、演奏家の演奏を調べる

ことは大いに役立つと思われるが、計算機音楽の分野でその研究はあまり行われていない。聴取者にどのように聴かれるかという問題もおおむね心理学の問題としてとどまっている。これらの問題が、音楽芸術と心理学、音響学、情報科学にまたがる学際領域の問題として分析されるのは今後待たなければならない。

3. 計算機演奏研究の概観

楽音合成の方法については本特集の中村勲氏執筆を参照されたい。

計算機プログラムによる楽音合成は Mathews²⁴⁾ によって 1950 年代に始められ MUSIC I-V が発表された。これらは 1960-70 年代に他の研究機関へ移植され、その研究者たちによって機能追加を含む多くの版 (MUSIC 4 B, MUSIC 4 F, MUSIC 4 BF, MUSIC 7, MUSIC 360, MUS 10, MUSIC 10, MUSIC 11 など) へ書き直された。その他、Abbott による INV, Berg による PILE が知られている。(同じく楽音合成の解説を見よ。)

ハードウェア・シンセサイザによる合成と、それを制御するソフトウェアの研究は 1970 年代に入ってから、Mathews-Moore²⁵⁾ のアナログ・シンセサイザのための実時間システム GROOVE (5. 参照) に始まった。そして、Buxton²⁶⁾、²⁷⁾ の SSSP (5. 参照)、Chabade らの PLAY, Rolnick の SYN 4 B, Abbott の 4 CED, Loy の MUSBOX, Moorer の FMX, Scaletti の IMSPLAY, Moore²⁸⁾ の Cscore/Cmusic (5. 参照) などのプログラム言語が現れた。(以上の諸言語の概略については中村²⁹⁾ を見られたい。) このほか、汎用言語をベースにした次のようなものがある(カッコ内はベースとする言語): Schottstaedt³⁰⁾ (SAIL), Collinge³¹⁾ (FORTH), Friberg³²⁾ (LISP), Vandenheede³³⁾ (PROLOG), Anderson³⁴⁾ (FORMULA), Dannenberg³⁵⁾ (C), Desain³⁶⁾ (LOGO), Boynton³⁷⁾ (LISP), Orlarey³⁸⁾ (LISP), Greenberg³⁹⁾ (LOGO), Duthen⁴⁰⁾ (LISP), Pope⁴¹⁾ (SMALL-TALK)。一方 Schmidt⁴²⁾ は自然言語による音楽記述言語を論じている。Rodet-Cointe⁴³⁾ の FORMES や Dannenberg⁴⁴⁾ の Arctic (この二つは 5. 参照) は高位の実時間音楽記述言語として注目を集めている。

汎用の作曲・演奏環境の下では、多種の音源を操作する方法論、高位の言語、ユーティリティが重要性を増す。Decker⁴⁵⁾ のモデュラな楽音合成ソフトウェア、Vandenheede⁴⁶⁾ の音色のハイブリッドの探求、

Malherbe ら⁴⁷⁾の木管系の重音奏法音の解析とデータベース, Barrière ら⁴⁸⁾の楽器音の音色ネットワークを使う音色制御, Ashley⁴⁹⁾の知識ベースによるFM音源の音色設計, Lo⁵⁰⁾の音色の内挿法などである。また音ファイルについて, Roth ら⁵¹⁾, Hebel⁴²⁾がある。

音の空間再生の研究は電気音響の分野で行われてきた。1959年にLeakey⁵³⁾の2チャンネル・ステレオの聴取実験があり, 1962/1970年Schroeder^{54), 55)}による残響の研究がある。1971年Chowning⁵⁶⁾は固定した複数のスピーカを用いて音源の疑似的な距離感と移動感を表現する方法を論じた。1973年Gerzon⁵⁷⁾は高さ感を与える音再生を, 1975年Bernfeld⁵⁸⁾は多チャンネル系における音定位を, 1977年Borenus⁵⁹⁾は劇場における移動音像を取り扱った。1983年Kurozumi-Ohgushi⁶⁰⁾は両耳間相関係数と音源の広がり感・距離感の関係を導き, 音像の広がり制御の方法を示した。計算機音楽の中では1979年Moorer⁶¹⁾が残響を, 1983年Moore⁶²⁾が空間処理のモデルを, 1984年Kendall ら⁶³⁾が空間聴を論じた。またGalante⁶⁴⁾, Kendall ら⁶⁵⁾, Martel⁶⁶⁾によるソフトウェアの研究がある。

音律に関しては, 非平均律あるいは微分音律の問題を最近Computer Music Journal が特集で取り上げた(Keisler⁶⁷⁾, Carlos⁶⁸⁾, Barlow⁶⁹⁾, Polansky⁷⁰⁾, Kirck⁷¹⁾)。今後, 多様な音律のもとでの作曲・演奏研究に新たな展開が出てくると思われる。

我が国の計算機演奏の研究は欧米に比べてずっと遅れて始まった(小嶋⁷²⁾)。1979年からの玉木ら⁷³⁾の電子オルガン自動演奏, 1980年小嶋⁷²⁾のピアノ自動演奏(CRS), 1981年梶谷⁷⁴⁾のリコーダ・バイオリン・チェロのロボット合奏, 1981年小川⁷⁵⁾の多種音律オルガンの製作と自動演奏, 1984年高澤⁷⁶⁾のピアノ自動演奏(COMSCORE), 1985年三好ら⁷⁷⁾の音楽情報システムとピアノ自動演奏(PSYCHE-EUROPA), 1985年大原ら⁷⁸⁾の楽譜視奏ロボット, 1986年今井ら⁷⁹⁾のピアノ音楽採譜・模奏, 1986年田口⁸⁰⁾のピアノ自動演奏(MUSE, 5. 参照) 1987年平田ら⁸¹⁾のICOTone などである。1985年に音楽制作業務向けの言語MUSICAL⁸²⁾が発表された。

近年はMIDIインタフェースをもつ楽器が急速に普及し, これをパーソナル・コンピュータで制御するシステムの研究が世界的に流行している。市販のソフトウェアも多種類出まわるようになったが, これについてはYavelov⁸³⁾, 坂崎⁸⁴⁾を見られたい。

4. 演奏制御

音楽は実時間にそって展開される芸術の一つである。自然で美しい演奏を実現するためには, 音の時間的な発展が精度高く制御できなければならない。表-1は音楽がもついろいろな時間単位を示している。

表-1 音楽がもついろいろな時間単位の大きさ(秒)

可聴音の上限周波数(18kHz程度)	10 ⁻⁴
楽音(基本波の周波数4kHz~20Hz程度)	10 ⁻³ ~10 ⁻¹
リズム(脈搏, 呼吸, 歩行の次元)	10 ⁰
動機, 走向, 楽句	10 ¹
楽章, 楽曲	10 ² ~10 ⁴

ここで, 10⁻¹秒の領域(音響振動から律動への遷移領域)に注意したい。楽音を機械的に一定なピッチ, 音色, 強さから変えて発音することや機械的に一樣な律動から変えて発音することは, 好い演奏のための必要条件といわれている。アタック変化やビブラートなどである。このような細やかな表情に特有な時間スケールはこの領域に入る。また前打音やトリルのような装飾音の音符の持続時間や, テンポのゆれを作るための音符長さの微妙な変動量もこの領域に入る。

さて, 一つの演奏が実現されるためには, 作曲から始まる音楽的思考の流れの中で, さまざまな心理的フィードバックが存在する。これについてLoy⁸⁵⁾による図-1のような二元的な見方が考えを整理するのに参考になる: 静的(static)とは出力が入力だけに依存する状態, 動的(dynamic)とは出力が入力だけでなく過去の出力にも依存する状態をさす。前者は計画, あるいはインプット間の関係づけの作業であり, 後者は制御, あるいはインプットとアウトプットの関係づけの作業である。一方, 因果的(causal)とは単調な時間発展の状態を, 非因果的(non-causal)とは不連続のないしは行きつ戻りつの状態をさす。たとえばレコード再生は静的で因果的, 普通の作曲活動は動的で非因果的である。リハーサルも同様に動的で非因果的といえる。本番の演奏は因果的である。しかしこのとき, 奏者が目で楽譜を追う作業は静的であり, 奏者が自身の演奏を聞きながら環境に合わせる作業は動的である。

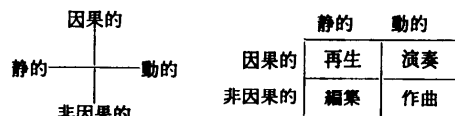


図-1 制御理論からみた音楽活動

この視点から実時間 (= 因果的) 演奏システムを分類することができる。静的なシステムとはタスクの時間発展をあらかじめ定められたとおりに制御するもので、ここでは各タスクをそれぞれの最終時刻に間に合わせるだけが問題にされ、タスク間の連続性を取ることはシステムを使用する側に委ねられる。これに対して、動的なシステムはフィードバック要素をもち、環境に反応しながらタスクの連続性を保持する機能をもつ。

上の定義を使うと、テープ音楽を始めとして、シーケンサ利用の音楽や音符リスト音楽 (普通の音楽ソフトウェアによるもの) は、制作の過程は動的 (かつ非因果的) であるが、その再生は (計算機を実時間的に使うとしても) 静的である。動的な実時間演奏システムとは、演奏時に即興的な作曲や演奏の表現要素が取り込めるシステムということになる。なお、Anderson-Kuivila⁸⁶⁾、Loy⁸⁷⁾ が実時間のタスク・スケジューリングを論じている。

静的な演奏で、緩急を任意に指定する方法について Jaffe⁸⁸⁾ は、一つの時間尺度を他の時間尺度へ写す時間マップ関数とその時間微分であるテンポ関数を用いることを提案した。田口⁸⁹⁾ は記譜音価を演奏音価へ変換する方法を提案した。両者は概念的には同値であるが、階層的に作用させるときの方法に違いがある。

即興性のない計算機音楽 (つまり静的な演奏) は演劇的性格を欠いた音楽であろう。Appleton⁸⁹⁾ は計算機演奏における即興の必要性を、Chadabe らの見解を交えながら論じている。Chadabe⁹⁰⁾ は演奏時の (システムに支援された) 即興的な作曲を 1970 年代から研究している一人である。彼の方法は、演奏プログラム中に、実行時に奏者 (作曲家兼演奏者) がそれに介入できる仕組みを入れておくものである。具体的にはいくとおりの機能を定義しておいて、それを実行時に選択したり、機能のパラメータに値を入れる。ここで、機能の応答の仕方には奏者にも予測できない (たとえば乱数による) ものが含まれていてよい。Chabot⁹¹⁾ も同様であるが、彼らは機能選択とパラメータ設定の仕事は奏者による即興性、機械の応答のうち奏者に予測できない部分を機械の即興性と定義している。

ところで伴奏は、ソリストの演奏に合わせなければならぬから、動的でなければならない。Dannenberg⁹²⁾⁻⁹⁴⁾、Vercoe⁹⁵⁾ は適応的な動的自動伴奏システムを発表している。伴奏すべき楽譜をあらかじめ入力しておき、演奏時に歌手や奏者などの共演者の表

現 (テンポの変動など) を実時間で電気音響的に自動計測し、そのデータを分析しながら追従するものである。リハーサルをとおして奏者の演奏法の特徴をある程度分析・記憶し、本番に利用するという適応機能をもたせてある。

5. 音楽言語

3. に引用した 6 つの言語 GROOVE, SSSP, Cscore/Cmusic, FORMES, Arctic, MUSE の概要を紹介する。

1) GROOVE²⁵⁾ は、1968 年に Bell Laboratories の Mathews と Moore によって開発が始められたソフトウェアシステムで、アナログ・シンセサイザを制御する。計算機は当時の中型機 DDP-224 で、主外部記憶に磁気ディスクを、コンソールにタイプライターと CRT ディスプレーをもつほか、GROOVE 用の入出力としていくつかの AD 変換器と DA 変換器を備えていた。

このシステムでは、アナログ・シンセサイザの周波数、振幅などの制御変数をすべて時間の関数として把握することを設計の基本に置き、これらの関数を対話的に生成、蓄積、編集、実行させる機能をもたせている。関数は、基本的な周期関数と利用者供給の関数を FORTRAN ふうの算術式 (GROOVE 式) で組み合わせることで新しく定義できるが、その定義式に実時間入力の値との算術演算を含められるのがこのシステムの一つの特徴である。後者は、実時間入力機器 (ポテンショメータなど) に命名された固有の関数名を算術式の中に書けばよい。新しく定義された関数は必要ならばファイルに蓄積できる。関数の実行が演奏である。実時間入力の関数名を含む関数を実行すると、刻刻の入力値が算術式の中に取り込まれて関数が評価される。このように実時間入力によって関数値が確定する関数もファイルに蓄積できる。蓄積された関数は、GROOVE 式によってさらに新しい関数を定義するのに使える。

2) SSSP^{26), 27)} (Structured Sound Synthesis Project) は、1977 年にトロント大学の Buxton らによって開発が始められた対話的作曲・演奏環境を提供するシステムであり、UNIX のもとで動く PDP 11/45 を用いて、ハードウェア・シンセサイザを制御することを目的とした。(1982 年に SSSP の研究資金は打ち切られ、翌年ホスト計算機はスクラップ化された。SSSP のソフトウェア遺産は Helicon System 社へ

引き継がれた.)

このシステムでは、計算機支援の作曲・演奏作業を音色パレット、スコア、オーケストレーション、演奏の四つの機能に分け、それぞれの機能を独立に支援するとともに、それらを統合する環境を提供することをねらいとしている。音色指定のない音符の任意の構造体をスコアと呼び、階層的に定義される Mevent (musical event) という文法的対象で表現する。ここで最下層の Mevent は一つの音符であり、最上層は楽曲全体を表す。Mevent に適当なパラメータを与えることで引用時に変形して使うことができる。(一つのマスタコピーから、パラメータによって修飾した異なるインスタンスが導ける。)

Mevent はスコアファイルに置かれるが、その実体はC言語の構造型で記述する。この構造型は、システムを統括するためのシンボルテーブルへのポインタ、ならびにそのスコアの総時間、音強、前後のスコアとの時間的關係、子 Mevent リストへの双方向ポインタなどを定義する型の集合である。一方、音色はオブジェクトファイルというファイルに置かれるが、その実体も構造型で記述される。この構造型は、シンボルテーブルへのポインタおよび発振器に関する情報(台数、楽音合成方式など)を定義するための型の集合である。

3) Cscore/Cmusic^{28),62)} は、1980年代初期からカリフォルニア大学サンディエゴ校の CARL=Computer Audio Research Laboratory で Moore (Bell Labs. から移った) によって開発が始められた作曲と実時間演奏システム用の音楽記述言語である。システム全体がC言語で構築されており、Cscore はスコアファイルのプリプロセッサ、Cmusic は楽音合成プロセッサである。Cscore の出力が Cmusic へ入力として渡される。マクロ定義ファイルの取り込み機能を活用している。Cmusic は MUSIC V の機能拡張を伴うC言語版であり、音の空間処理機能をもつことに特徴がある⁶²⁾。なお CARL については Dolson⁹⁶⁾、湯浅・志村⁹⁷⁾ が参考になる。

4) FORMES^{43,98)} は、1980年初期からフランスの IRCAM (Institut de Recherche et de Coordination Acoustique/Musique) で Rodet と Cointe によって開発が始められたオブジェクト指向の作曲・演奏環境を提供するシステムであり、Lisp 言語を使って VAX 11/780 上に構築された。(オブジェクト指向プログラミングの音楽分野への応用については Lieberman⁹⁹⁾

がある。)

FORMES の設計方針は、特定の音や音符を個々に記述するのではなく、音楽的に高位な対象のインスタンスとして実現しようという考え方に立つ。(ある意味で SSSP と同様の考え方である。) このことによって、特定の楽音合成技術から独立させるという意図もある。(IRCAM においては、4X, CHANT などのハードウェア・シンセサイザが動いており、音楽的な動機に応じてそれらを自由に使いたいという目的があった。)

この言語の中心は、プロセスと呼ばれる時間依存型の一種のオブジェクトである。プロセスの一般形は {ルール, モニタ, 環境変数, 子プロセス} から成る。プロセスはメッセージによって駆動される。一つのプロセスに含まれる子プロセスたちは直列か並列に動作するように定義される。この定義から、プロセスの階層構造が形成される。ルールはプロセスが駆動されたとき、そのプロセスがなすべき動作を記述する。(たとえばエンベロープの刻々の値の計算。)環境変数は静的変数で、ルールの実行時にデータを供給する。モニタは子プロセスたちの開始・終了時刻のスケジュール、および自身と関係する外部のプロセスとの連係を取る。プロセスは親プロセスから渡された環境変数値によって自身の環境変数値を修飾して用いる。

プロセスの実行が演奏である。実行は実行したいプロセス(どの水準の“サブ”プロセスであってもよい)へ play メッセージを送ることで実現する。(階層的に)駆動されるそれぞれのプロセスは終了条件(たとえば、プロセスに定義された持続時間を消費したという事象)によって自身を終了させ、それを呼び出した親プロセスへ制御をかえす。全体の演奏は play をかけたプロセスの終了条件が満たされたとき終わる。

5) Arctic⁴⁴⁾ は、1984年からカーネギーメロン大学の Dannenberg らが提案している言語で、音楽の実時間制御を、手続的ではなく、宣言的に記述する音楽言語である。(論文は彼らの基本的構想を提示しているだけで、音楽演奏のための具体的な機能にはほとんど触れていない。)

制御対象はプロトタイプと呼ばれる高位の関数で、それは既定義のプロトタイプたち(システム定義の原始プロトタイプが始めにある)の間のある種の算術演算、時間軸の尺度変化と移動、条件つき繰り返しなどによって定義される。(高位の関数とは、その関数値自身が別の関数を表すという意味である。)

プロトタイプは starting time という引数をもつ。この引数に実時刻へ割り当てることによって、実時間軸上の関数が実現される。これをインスタンスと呼ぶ。どのプロトタイプも、そのほかに少なくとも二つの引数 duration factor と terminate をもつ。前者はプロトタイプでの定義区間幅を実時間幅に変換する係数を、後者は非同期的事象によってこのプロトタイプが打ち切られる条件を与える。

原始プロトタイプの例として、単位区間上で定義された周期1の正弦関数、定値関数、ランプ関数などをあげ、これらを用いて音の波形を表す関数を定義する方法を例示し、さらにそれぞれが特有の振幅エンベロープをもつ部分音(倍音)を重ね合わせて一つの楽音を合成するプロトタイプにも言及している。

6) MUSE⁹⁰ (Music in Structured Expression) は筆者によるピアノ系楽器のための演奏言語である。ピアノ演奏の制御変量は、打鍵強度、持続時間、ペダル操作であり、その精密な制御が演奏の質を決めると考えられる。具体的には、局所的なものとして (i) 和音の響きを得るための構成音間の音強配分や必要に応じた非同時打鍵、(ii) 裝飾音の処理(拍進行に対して前打音は先行させ主音は遅らせるとか、トリル内部での緩急をつけるための音価配分)、(iii) スタッカートからレガートゥッシュモまでのさまざまな階調の継切(アーティキュレーション)とそれに付随する音尾の処理(音鍵の戻り速度)などがあり、一方大域的な制御には、(iv) 拍子リズムやフレージングを表現する強弱や緩急の変化のような経時(水平)的に大域的なものと、(v) 声部間の強弱や緩急の対比のような同時(垂直)的に大域的なものがある。このほか、(vi) 必ずしも打鍵とは同期しないペダル踏み込みとその深さの制御も必要である。

このような音楽的に多様な質の制御を統一的に記述するために、音符の列を任意の深さをもつ一般化された直並列構造に表現する機能と、演奏表現を音楽の意味に即した形でパラメータ化し、それらを上記の直並列構造の任意の部分構造に付与できる機能をもつ言語として MUSE を設計した。パラメータの種類は、拍属性、強弱属性、緩急属性、継切属性、時間シフト属性、ペダル深さ属性、音鍵戻り速度属性である。付与された属性の解釈は、属性ごとに上位構造のものを下位構造へ継承させる。これによって、時間的秩序の中で速度の揺らぎを与えることや、一音一音の音の個性を大域的環境に整合させることが比較的容易にで

きる。

構造化の考え方は SSSP や FORMES と同様であるが、汎用性を省き、ピアノ演奏の表現に特化した点にこの言語の特徴がある。16ビット系のパーソナルコンピュータ用のコンパイラをC言語で実現した。

6. おわりに

我が国で計算機音楽のシンポジウムが情報処理学会により初めて開かれたのは1984年であった¹⁰⁰。しかし欧米では1974年に第1回の計算機音楽会議が開かれるなど研究者層の厚さと蓄積¹⁰¹は大きい。(この会議は後に国際計算機音楽会議 International Computer Music Conference と名前を変え、欧米において毎年開催されている。)幸い我が国でも、近年この分野に活発な研究活動が芽生えている¹⁰²。多くの方の活躍を期待したい。

本稿は昭和62年度文部省科学研究費 No. 62580027 による研究の一部として執筆した。

参考文献

(CMJ は Computer Music Journal, ICMC××は Proceedings of International Computer Music Conference 19××の略記。)

- 1) Bowers, Q. D.: Encyclopedia of Automatic Musical Instruments, The Vestel Press, New York (1979).
- 2) Roehl, H. N.: Player Piano Treasury, The Vestel Press, New York (1980).
- 3) 梶谷: オーケストラ楽器の自動演奏, 音響誌 41 (6), pp. 396-401 (1985).
- 4) 高澤, 三好: ピアノと電子楽器の自動演奏, 音響誌 41 (6), pp. 402-406 (1985).
- 5) Kendall, G. S. et al.: Essays on the 1986 International Computer Music Conference, CMJ 11 (2), pp. 35-48 (1987).
- 6) Seashore, C. E.: Psychology of Music, McGraw-Hill/Dover Publications (1938/1967).
- 7) 梅本: 音楽心理学, 誠信書房, 3ページ (1966).
- 8) Shaffer, L. H.: Performances of Chopin, Bach, and Bartok—Studies in Motor Programming, Cognitive Psychology 13, pp. 326-376 (1981).
- 9) Gabrielsson, A. et al.: Performance of Musical Rhythm in 3/4 and 6/8 Meter, Scandinavian J. Psychology 24, pp. 193-213 (1983).
- 10) Clarke, E. F.: Structure and Expression in Rhythmic Performance, in Book: Musical Structure and Cognition (P. Howell (ed.)), Academic Press (1985).
- 11) 池内他: リズム並びにテンポのゆらぎの数量化に関する研究—あるギター曲のメロディーを例

- にとった場合, 音響誌 40(4), pp. 228-234 (1984).
- 12) 小川, 三好, 五十嵐: ピアノ演奏のコンピュータ分析とその自動演奏への応用, 情報処理学会プログラミングシンポジウム報告集, pp. 159-168 (1986).
 - 13) 中島他: 楽譜に記された時価と演奏家の実現する長さとの系統的なくい違いについて, 音響誌 43 (7), pp. 478-487 (1987).
 - 14) 太田, 田口: ピアノ演奏における緩急法の分析-局所メトロノーム速度によって, 甲南大学紀要理學編 33 (2), pp. 125-150 (1986).
 - 15) 太田, 田口, 馬道: ショパンのワルツとマズルカの演奏リズムについて, 音楽音響研 MA 87-14 (1987).
 - 16) 中村, 難波: 演奏音を媒介とした演奏者と聴取者の間のコミュニケーションについて, 大阪大学教養部研究集録第 30 輯, pp. 3-16 (1982).
 - 17) Sloboda, J.A.: The Communication of Musical Metre in Piano Performance, *Quart. J. Experimental Psychology* 35 (A), pp. 377-396 (1983).
 - 18) Gabriellsson, A.: Interplay between Analysis and Synthesis in Studies of Music Performance and Music Experience, *Music Perception* 3 (1), pp. 59-86 (1985).
 - 19) Senju, M. and Ohgushi, K.: How are the player's ideas conveyed to the audience?, *Music Perception* 4 (4), pp. 311-324 (1987).
 - 20) 森, 菅, 田口: 音楽テンポに関する心理学的一研究, 甲南大学紀要文学編 63, pp. 23-38 (1987).
 - 21) 田口: 自動演奏と演奏モデル-演奏の芸術性をめざして, 102), pp. 42-53.
 - 22) Sundberg, J. et al.: Musical Performance-A Synthesis-by-rule Approach, *CMJ* 7 (1), pp. 37-43 (1983).
 - 23) Deutsch, D. (ed.): *The Psychology of Music*, Academic Press, 1982; 寺西他監訳: 音楽の心理学 (上・下), 西村書店 (1987).
 - 24) M. V. Mathews: *The Technology of Computer Music*, M. I. T. Press (1969).
 - 25) Mathews, M. V. and Moore, F. R.: GROOVE - A Program to Compose, Store, and Edit Functions of Time, *Comm. ACM* 13 (12), pp. 715-721 (1970).
 - 26) Buxton, W. et al.: An Introduction to the SSSP Digital Synthesizer, *CMJ* 2 (4), pp. 28-38 (1978).
 - 27) Buxton, W. et al.: The Use of Hierarchy and Instance in a Data Structure for Computer Music, *CMJ* 2 (4), pp. 10-20 (1978).
 - 28) Moore, F.R.: The Computer Audio Research Laboratory at UCSD, *CMJ* 6 (1), pp. 18-29 (1982).
 - 29) 中村: 音楽・楽音・コンピュータ, 102), pp. 54-64.
 - 30) Schottstaedt, B.: A Composer's Idea of a Language, *CMJ* 7 (1), pp. 11-20 (1983).
 - 31) Collinge, D.J.: A Language for Computer Music Performance, *ICMC* 84, pp. 217-220.
 - 32) Friberg, A. and Sundberg, J.: A Lisp Environment for Creating and Applying Rules for Musical Performance, *ICMC* 86, pp. 1-3.
 - 33) Vandenheede, J.: Musical Experiments with PROLOG II, *ICMC* 86, pp. 5-10.
 - 34) Anderson, D.P. and Kuivila, R.: A Model of Real-time Computation for Computer Music, *ICMC* 86, pp. 35-41.
 - 35) Dannenberg, R.B.: The CMU MIDI Kit, *ICMC* 86, pp. 53-56.
 - 36) Desain, P. and Honing, H.: LOCO-Composition Microworlds in LOGO, *ICMC* 86, pp. 109-118.
 - 37) Boynton, L. et al.: MIDI-LISP-A Lisp-based Music Programming Environment for the Macintosh, *ICMC* 86, pp. 183-186.
 - 38) Orlarey, Y.: MLOGO-A MIDI Composing Environment for the Apple IIe, *ICMC* 86, pp. 211-213.
 - 39) Greenberg, G.: Computers and Music Education-a Compositional Approach, *ICMC* 86, pp. 349-351.
 - 40) Duthen, J. and Potard, Y.: Le-loup, an Object-Oriented Extension of Le-lisp for an Integrated Computer Music Environment, *ICMC* 87, pp. 151-158.
 - 41) Pope, S. T.: Smalltalk-80-based Music Toolkit, *ICMC* 87, pp. 166-173.
 - 42) Schmidt, B.L.: A Natural Language System for Music, *CMJ* 11 (2), pp. 25-34 (1987).
 - 43) Rodet, X. and Cointe, P.: FORMES-Composition and Scheduling of Processes, *CMJ* 8 (3), pp. 32-50 (1984).
 - 44) Dannenberg, R.B. et al.: Arctic-A Functional Language for Real-time Systems *CMJ* 10 (4), pp. 67-78 (1986).
 - 45) Decker, S.L. et al.: A Modular Environment for Sound Synthesis and Composition, *CMJ* 10 (4), pp. 28-41.
 - 46) Vandenheede, J. and Harvey, J.: Identity and Ambiguity-the Construction and Use of Timbral Transitions and Hybrids, *ICMC* 85, pp. 97-102.
 - 47) Malherbe, C. et al.: Functional Integration of Complex Instrumental Sounds in Musical Writing, *ICMC* 85, pp. 185-192.
 - 48) Barrière, J.B. et al.: Models of Continuity between Synthesis and Processing for the Elaboration and Control of Timbre Structures, *ICMC* 85, pp. 193-198.

- 49) Ashley, R. D. : A Knowledge-based Approach to Assistance in Timbral Design, ICMC 86, pp. 11-16.
- 50) Lo, Y. O. : Techniques for Timbral Interpolation, ICMC 86, pp. 241-247.
- 51) Roth, J. M. et al. : A Network Sound System for UNIX, ICMC 85, pp. 61-67.
- 52) Hebel, K. J. : A Machine-independent Sound Conversion/Storage System, ICMC 85, pp. 125-127.
- 53) Leakey, D. M. : Some Measurements on the Effects of Interchannel Intensity and Time Differences in Two Channel Sound Systems, JASA 31 (7), pp. 977-986 (1959).
- 54) Schroeder, M. R. : Natural Sounding Artificial Reverberation, J. Audio Eng'g. Soc. 10 (3), pp. 219-223 (1962).
- 55) Schroeder, M. R. : Digital Simulation of Sound Transmission in Reverberant Spaces, JASA 47 (2), pp. 424-431 (1970).
- 56) Chowning, J. M. : The Simulation of Moving Sound Sources, J. Audio Eng'g. Soc. 19 (1), pp. 2-6 (1971).
- 57) Gerzon, M. A. : Periphony—With-height Sound Reproduction, J. Audio Eng'g. Soc. 21 (1), pp. 2-10 (1973).
- 58) Bernfeld, B. : Simple Equations for Multi-channel Stereophonic Sound Localizations, J. Audio Eng'g. Soc. 23 (7), pp. 553-557 (1975).
- 59) Borenus, J. : Moving Sound Image in Theaters, J. Audio Eng'g. Soc. 25 (4), pp. 200-203 (1977).
- 60) Kurozumi, and K. Ohgushi, K. : The Relationship between the Cross-correlation Coefficient of Two-channel Acoustic Signals and Sound Image Quality, JASA 74 (6), pp. 1726-1733 (1983).
- 61) Moorer, J. A. : About This Reverberation Business, CMJ 3 (2), pp. 13-28 (1979).
- 62) Moore, F. R. : A General Model for Spatial Processing of Sounds, CMJ 7 (3), pp. 6-15 (1983).
- 63) Kendall, G. S. and Martens, W. L. : Simulating the Cues of Spatial Hearing in Natural Environments, ICMC 84, pp. 111-125.
- 64) Galante, F. : AAS—Acoustic Ambience Simulator System, ICMC 86, pp. 281-284.
- 65) Kendall, G. S. et al. : Spatial Processing Software at Northwestern Computer Music, ICMC 86, pp. 285-292.
- 66) Martel, A. : The SS-1 Sound Spatializer—A Realtime MIDI Spatialization Processor, ICMC 86, pp. 305-307.
- 67) Keislar, D. : History and Principles of Microtonal Keyboards, CMJ 11 (1), pp. 18-28 (1987).
- 68) Carlos, W. : Tuning—At the Crossroads, CMJ 11 (1), pp. 29-43 (1987).
- 69) Barlow, C. : Two Essays on Theory, CMJ 11 (1), pp. 44-60 (1987).
- 70) Polansky, L. : Practical Tuning—An Agenda for the Use of Computers in Experimental Intonation, CMJ 11 (1), pp. 61-68 (1987).
- 71) Kirck, G. T. : Computer Realization of Extended Just Intonation Compositions, CMJ 11 (1), pp. 69-75 (1987).
- 72) 小嶋 : コンピュータによる自動演奏 (解説), 音響誌 37(11), pp. 569-574 (1981). (他に, ピアノの自動演奏—演奏分析システム, 情報処理講義集 pp. 733-734 (1980.3), ピアノの自動演奏システムについて, DEC ユーザ協議会論文集, pp. 61-64 (1984).).
- 73) 玉木他 : コンピュータによる電子オルガンの自動演奏システムのためのソフトウェアシステム, 九州工業大学研究報告 (工学) 39, pp. 181-188 (1979). (他に, 情報処理講義集, pp. 1501-1502 (1986. 10).).
- 74) 梶谷他 : リコーダとバイオリンの自動演奏システムの試作, 音響講義集, pp. 313-314 (1981. 10).
- 75) 小川 : 多音階オルガンの実現—調和比例論の系譜の中で, 千葉大学教養部研究報告 B-14, pp. 123-134 (1981).
- 76) 高澤 : コンピュータによる音楽演奏, 100 pp. 27-39.
- 77) 三好, 笹川, 五十嵐 : 総合音楽情報システム PSYCHE, 音楽音響研 MA 84-20 (1985).
- 78) Ohteru, S. et al. : Automated High Speed Recognition of Printed Music (WABOT-2 Vision System), Proc. Int. Conf. on Advanced Robotics, pp. 477-482 (1985).
- 79) 今井他 : 人間のピアノ演奏をシミュレートするシステム, 情報処理講義集, pp. 1477-1478 (1986. 10).
- 80) 田口 : ピアノ演奏言語の一設計, 音楽音響研 MA 85-20 (1986).
- 81) 平田他 : 新世代音楽システム ICOTone, 102 pp. 101-104.
- 82) 荒木 : 楽譜情報記述言語 MUSICAL, 放送技術 (1985. 4).
- 83) Yavelov, C. : MIDI and the Apple Macintosh, CMJ 10 (3), pp. 11-47 (1986).
- 84) 坂崎 : パソコン用演奏言語の比較, 102 pp. 124-130.
- 85) Loy, G. : Designing a Computer Music Workstation from Musical Imperatives, ICMC 86, pp. 375-380.
- 86) Anderson, D. P. and Kuivila, R. : Accurately Timed Generation of Discrete Musical Events, CMJ 10 (3), pp. 48-56 (1986).

- 87) Loy, G.: On the Scheduling of Multiple Processors Executing Synchronously, ICMC 87, pp. 117-124.
- 88) Jaffe, D.: Ensemble Timing in Computer Music, CMJ 9 (4), pp. 38-48 (1985).
- 89) Appleton, J.: Live and in Concert—Composer/ Performer Views of Real-time Performance Systems, CMJ 8 (1), pp. 48-51 (1984).
- 90) Chadabe, J.: Interactive Composing—an Overview, CMJ 8 (1), pp. 22-27 (1984).
- 91) Chabot, X. et al.: A Workstation in Live Performance—Composed Improvisation, ICMC 86, pp. 57-59.
- 92) Dannenberg, R. B.: An On-line Algorithm for Real-time Accompaniment, ICMC 84, pp. 193-198.
- 93) Bloch, J. J. and Dannenberg, R. B.: Real-time Accompaniment of Keyboard Performances, ICMC 85, pp. 279-289.
- 94) Dannenberg, R. B. and Mont-Reynaud, B.: Following an Improvisation in Real Time, ICMC 87, pp. 241-248.
- 95) Vercoe, B.: The Synthetic Performer in the Context of Live Performance, ICMC 84, pp. 199-200.
- 96) Dolson, M.: Recent Advances in Musique Concrète at CARL, ICM 85, pp. 55-60.
- 97) 湯浅, 志村: テープ音楽とコンピュータ, 102), pp. 182-191.
- 98) Boynton, L. et al.: Adding a Graphical User Interface to FORMES, ICMC 86, pp. 105-108.
- 99) Lieberman, H.: Machine Tongues IX—Object-oriented Programming, CMJ 6 (3), pp. 8-21 (1982).
- 100) 「計算機と音楽」シンポジウム報告集, 情報処理学会プログラミング・シンポジウム委員会(1984).
- 101) Loy, G. and Abbott, C.: Programming Languages for Computer Music Synthesis, Performance and Composition, ACM Computing Surveys 17 (2), pp. 235-265 (1985) (佐藤訳: 計算機音楽の合成, 演奏, 作曲用プログラミング言語, コンピュータ・サイエンス '85), 共立出版 (1985. 2).
- 102) 西村 (監修): bit 別冊「コンピュータと音楽」, 共立出版 (1987. 9).

(昭和 63 年 5 月 6 日受付)