

MIDI音源の発音遅延と 音楽心理学実験への影響

Delay measurement of MIDI synthesizers
and study of its problem

長嶋洋一

Y.Nagashima

Art & Science Laboratory

nagasm@computer.org

ハードウェア的なMIDI音源（GM音源）と、ソフトウェア的なGMサウンドドライバについて、MIDI信号入力から発音開始までの遅延時間のばらつきを計測した実験について、多種のMIDI音源の計測結果を詳細に報告する。この結果から推定される音源処理アルゴリズム（MIDI入力から発音までの信号処理）に関する考察を行い、音楽心理学的実験に使用できる場合と、実験の意味を失うために使用できない場合とを整理する。また、音源アルゴリズムを現実的に改良するための提案を行う。

1. はじめに

MIDIは既に約15年の歴史のあるComputer Musicにおけるスタンダードであるが、これまで主に楽器メーカーが Black Boxとしていたこともあり、その「遅延」に関しては今も図1のような誤解が存在する。

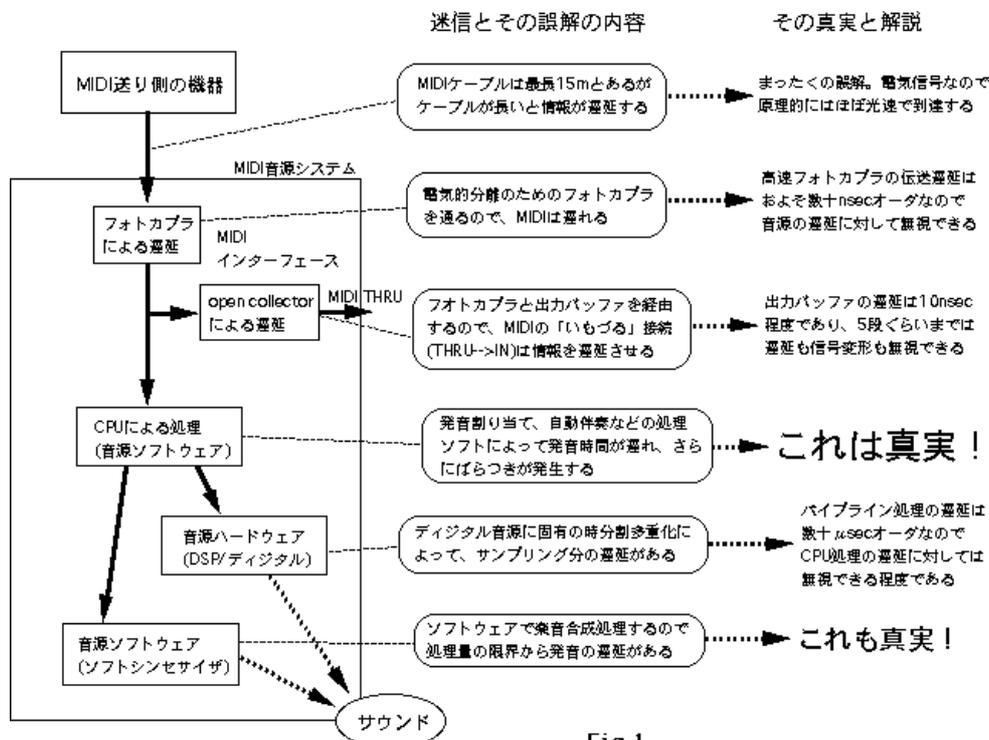


Fig.1

本稿では、市販されているMIDIシンセサイザやMIDI音源モジュール、さらにパソコン等のソフトウェアによるGM音源ドライバやGM音源モジュールについて、実際にMIDI信号を受けてから発音されるまでの発音の遅延とそのばらつきと確実性、さらにMIDI情報のトラフィックに対する振舞

いについて実験的に検証した。そして、特に音楽心理学のための実験環境として、危険性の認識を欠いて安易に利用されかねない現状に関連して、その問題点を検討した。

2. 実験計測環境の開発と二つの計測モード

分解能が4分音符単位で960とか謳われている市販のシーケンスソフトの時間管理にそのような精度が無いのは常識であり、実際のMIDI信号を処理する部分がブラックボックス内に入っているソフトウェアをこの計測に使用することは無意味である。そこでまず、必要な基準MIDIメッセージのみを正確に送出し、さらにその出力タイミングと厳密に同期したトリガ信号を発生する、図2のような計測専用ツールを開発した。

基準MIDIメッセージとして「MIDI1チャンネル、ノートナンバ60、ベロシティ100でON、その直後にOFF」という6バイトのMIDI情報を送出して、その最終バイトの送出と同期してデジタルポートに幅2.5m秒のトリガパルスを生じた。このMIDI信号を対象の音源に直接接続し、GMリセットによりそのサウンド出力に現れるピアノ音の立ち上がり点までの遅延をストレージスコープで20回計測し、平均遅延時間と最大値最小値をプロットした。この「シングルトリガモード」とともに、多量の

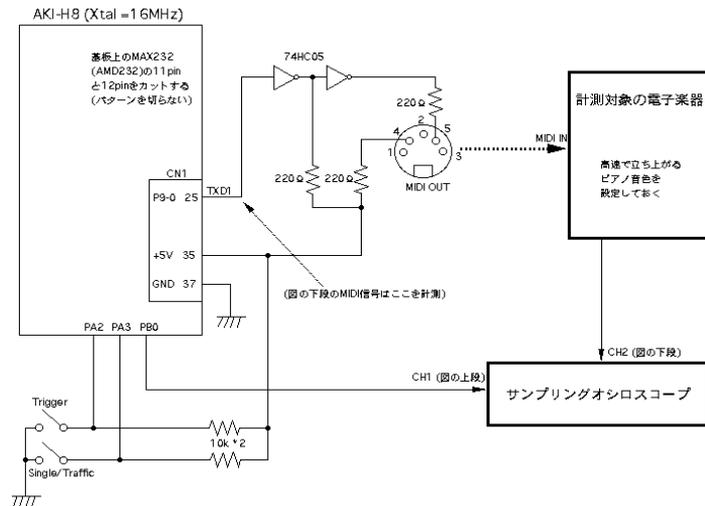


Fig.2

MIDIメッセージを受信して発音する「トラフィックモード」も設けて併せて計測した。ここでは、上記シングルトリガモードの情報と対応したトリガ信号を生成しながら、さらにバックグラウンドとして「他の二つのMIDIチャンネルで、ノートナンバ0から127までを順に、ベロシティ1でONして直後にOFFする多量のメッセージ」を流す。送出レートのチューニングにより、実際にMIDIケーブル上を流れるデジタル信号が約70%になるように調整した。これは実際の音楽の場では多数のセンサフュージョン情報をマージしたような非常に過酷な条件であるが、同時発音数は常に1音で情報としてはMIDI規格に従った条件であり、ベロシティが最小音量である多量のメッセージに埋没することなく、基準メッセージのサウンドが発音されることが求められる。図3、図4はこの両モード(上段はトリガ信号、下段はMIDIライン上の信号)である。



Fig.3

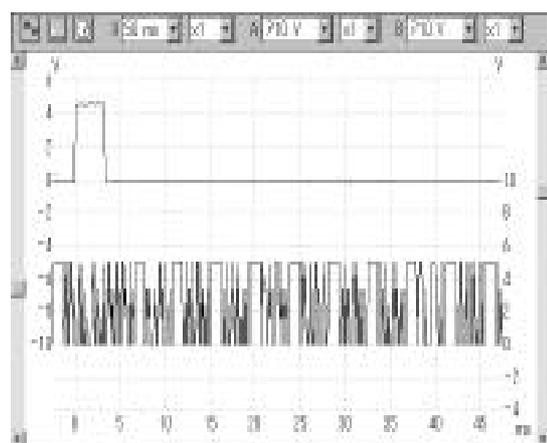


Fig.4

3. 計測対象と結果の検討

今回の実験で対象とした各種の音源 (実験の時期は1998年秋-冬なので、それぞれ最新の製品やソフトというわけではない) を以下の図5に示す。説明の便宜上、<1--><24>の記号を割り当てた。

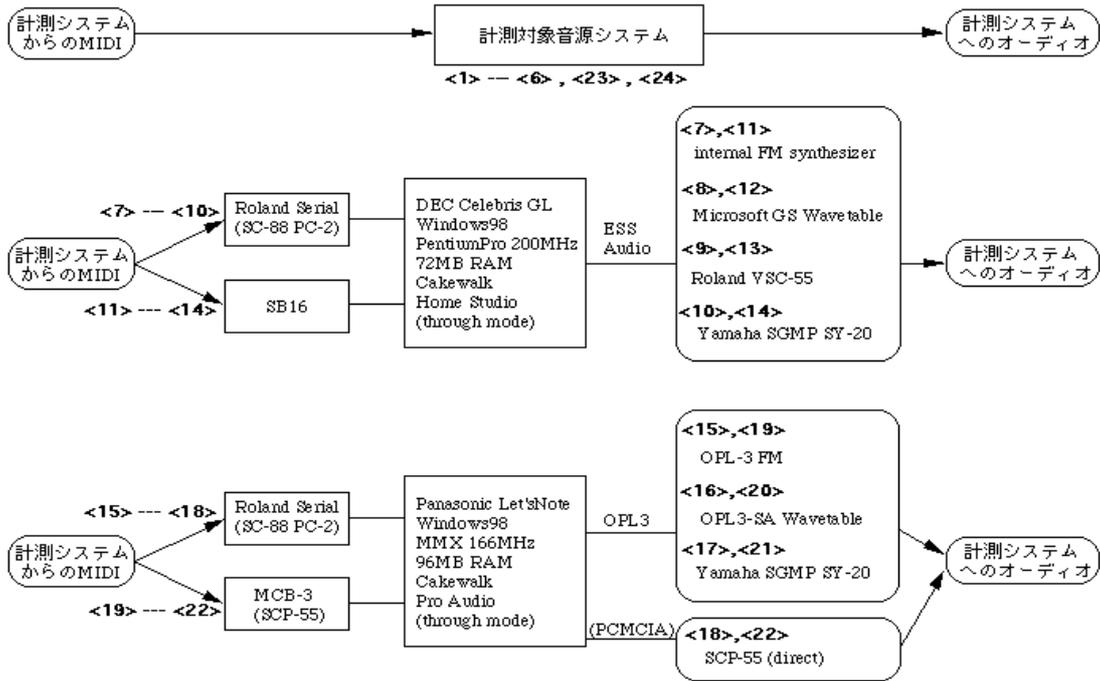


Fig.5

ハード音源としては、<1>SC-88STPro、<2>SC-55mark2、<3>05R/W、<4>GMega、<23>JV30、<22>SCP-55、パソコンのソフト音源としてはMacintosh(PowerBook2400)上のQuickTime3.0、WindowsではESSチップを用いたものとしてはFM音源、Microsoft GS Wavetableドライバ、VSC-55、OPLチップを用いたものとしてはOPL3-FM、OPL3-SA、SGMP driver、さらにSGI Indy上でのIRIX6.2のソフトGMシンセサイザ。WindowsパソコンのMIDIインターフェースについては、デスクトップ機がローランドシリアル(SC-88Pro)とSB16、ノート機がローランドシリアルとSCP-55を取り替えつつ使用した。

3.1 計測データの状況

図2の計測システムで、実際に測定対象となる音源を接続し、図3および図4に示すようなトリガ信号とMIDI信号で計測したデータ例のうちの二つを、以下に示した。

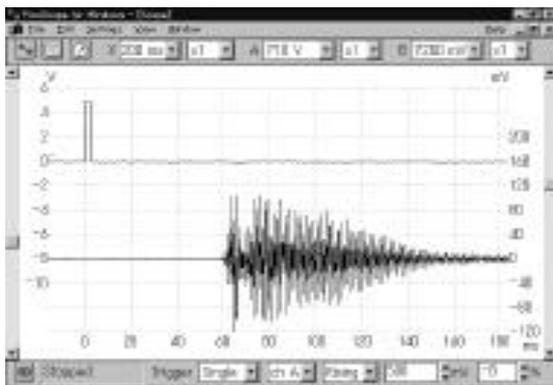


Fig.6

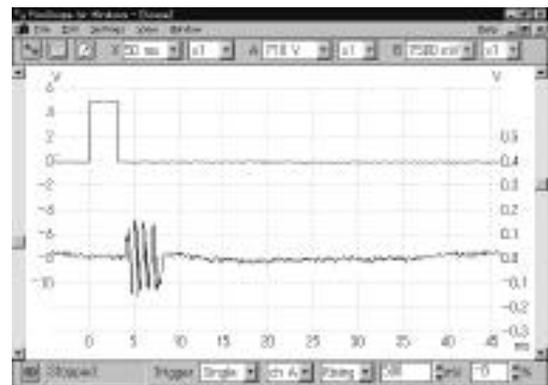


Fig.7

図6は、シングルトリガモードでの典型的な発音状況であり、この例ではトリガ信号の立ち上がりに対して約60msecの遅延で発音を開始している。また図7は、トラフィックモードでの典型的な状況であり、約4msecとかなり高速で発音を開始するが、バックグラウンドの「ベロシティ=1」の発音イベントによって同時発音数の制限に到達したためか、発音していたサウンドが強制終了させられている様子がよく判る。以下、図5に従った24タイプの音源の計測結果を、図8から図11に示す。

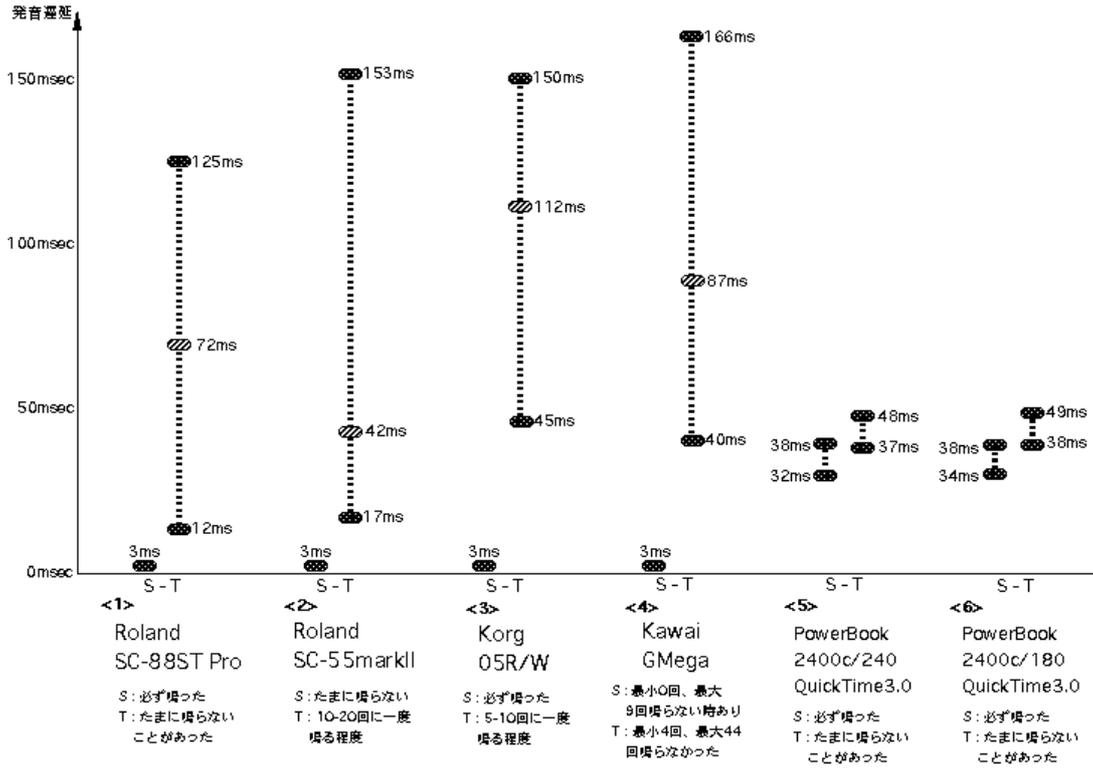


Fig.8

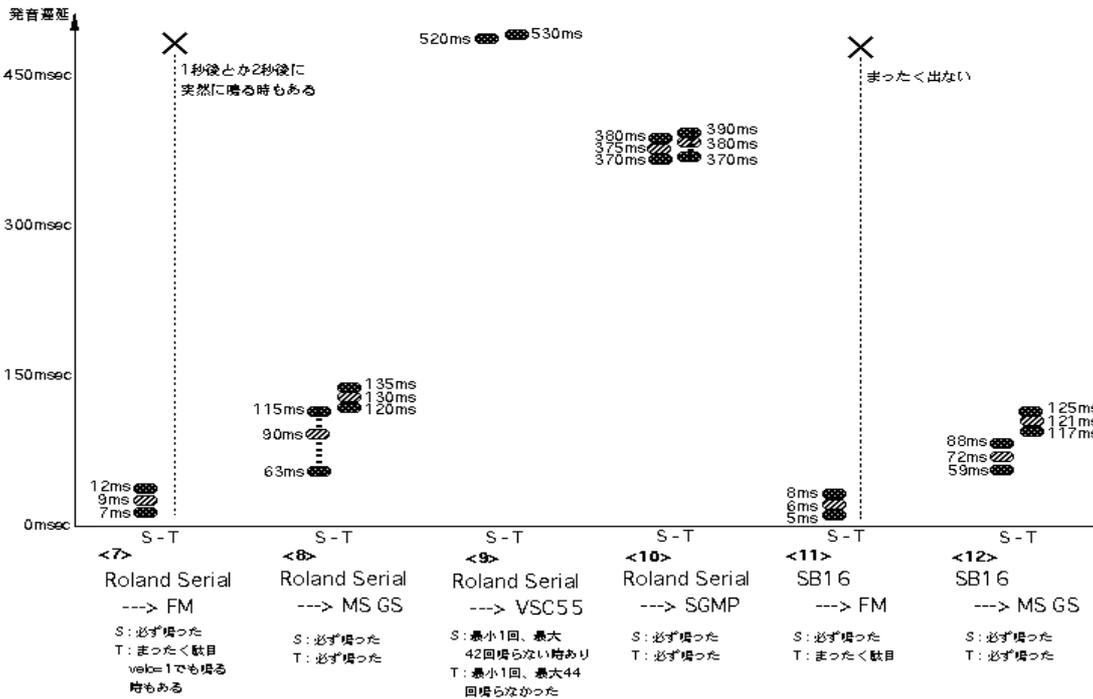


Fig.9

なお、図の中に発音遅延の最大値と最小値を示しているものの中で、「たまに鳴った」「*回鳴らなかった」というような記述のある音源のデータは、実質的にはあまり意味をなさない事に注意すべきである。バックグラウンドに多量の「ベロシティ=1」のイベントがあるにせよ、個々の瞬間での同時ON数は常に1か2かゼロであり、MIDI情報としては必ず発音すべきであるのに、これはたとえば10回続けて鳴らない、というような現象である。その意味するのは、センサデータのス

トリームとマージしているような状況であるにせよ、演奏者が鍵盤に向かって渾身の一撃の演奏を行ったのに、これが空を切って音源が鳴らない「ことがある」、という音楽的には致命的な事実であり、私見としては「音源(楽器)としては失格」という印象を持たざるをえない。また、この測定に使用した音源装置やソフトウェアについては、製造ロットやバージョンによって、ユーザの見えないところで仕様が変更されていることも予想できるので、追試に関しては十分に注意する必要がある点を指摘しておく。

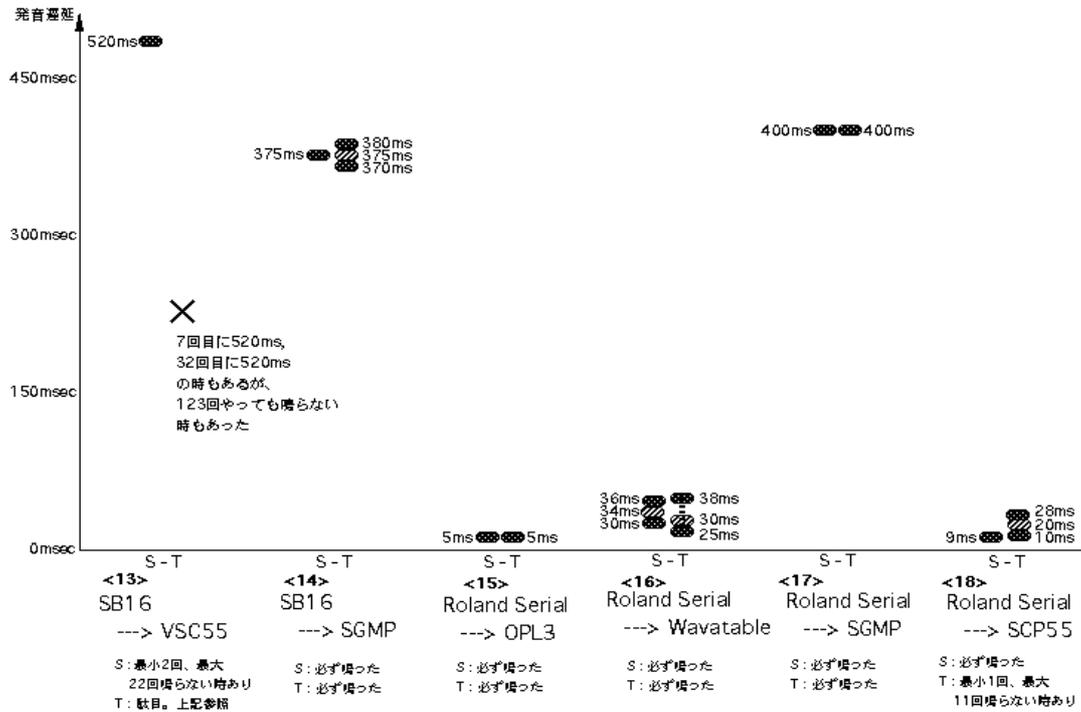


Fig.10

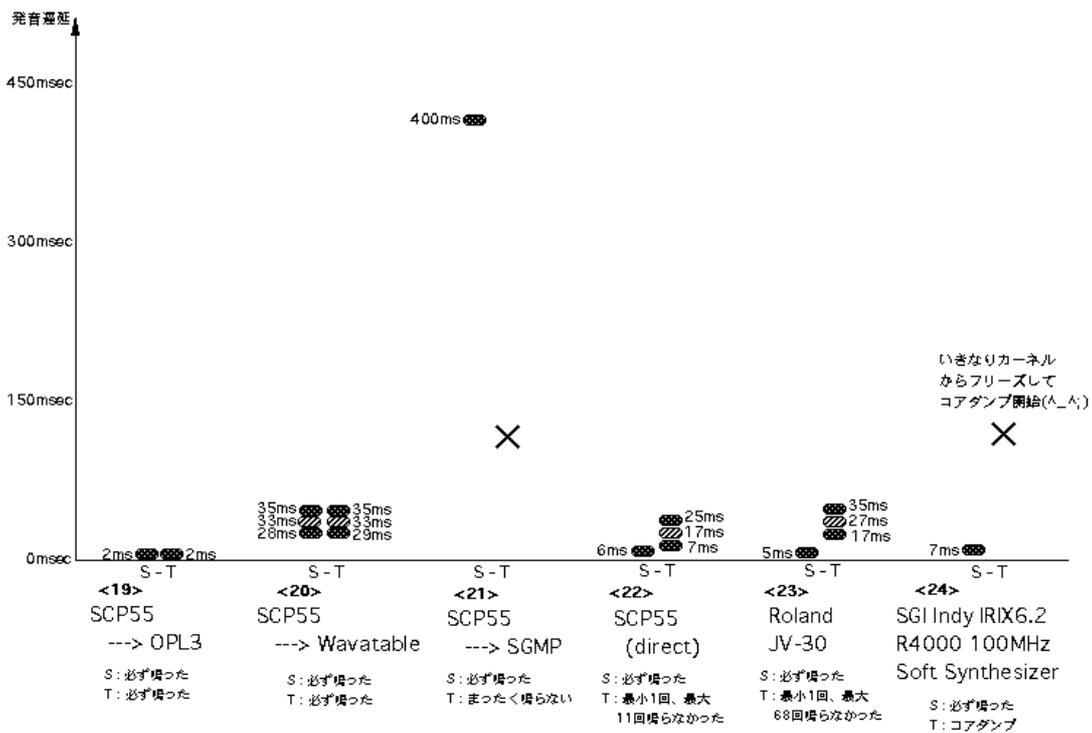


Fig.11

3.2 計測データの検討と考察

24種類の実験結果について、以下、いくつかの特徴を共有するグループに分けて検討した。

3.2.1 旧世代のGM音源装置

ここでは、<1>,<2>,<3>,<4>が対象である。これらの音源に共通する点としては、(1) シングルトリガモードでは非常に高速に発音する、(2) トラフィックモードでは、発音時間の絶対値が大きくなりそのばらつきがとても大きい、という特徴がある。また、<2>と<4>ではシングルモードで鳴らない事があったが、これは与えたイベントのON-OFF時間が非常に短いために、有効な演奏情報ではない、という判定処理を行った可能性はあるが、音源としては問題と思われる事実である。これら「旧世代GM音源」に共通するのは、内蔵された制御CPUの処理能力の制限で、MIDI受信割り込みから発音割当処理・音源DSP処理のルーチンのループ時間が長いために、MIDIイベントの発生したタイミングによって実際の発音までの時間が大きくばらつく事が大きな理由と考えられる。なお、本稿ではこの音源内部の情報処理について詳解する誌面がないので、以下のURLの解説を参照されたい。

<http://nagasm.org/hightech/03-03/>

<http://nagasm.org/hightech/03-04/>

3.2.2 新世代のGM音源装置

ここでは、<18>,<22>,<23>が対象である。音源に共通する点としては、(1) シングルトリガモードでは高速に発音する、(2) トラフィックモードでは、鳴らない時があるが鳴る時にはかなり高速に対応する、という特徴がある。これは比較的新しいハード音源であり制御CPUの性能が向上していることと、音源処理(アサイナ)[解説は上記URL参照]のアルゴリズムを改良したためと思われる。

3.2.3 QuickTimeソフト音源

ここでは、<5>,<6>が対象である。この音源に共通する点としては、(1) シングルトリガモードではやや遅く、30msec台の遅延がある、(2) トラフィックモードでは、たまに鳴らないことはあるが鳴る場合にはほとんどシングルトリガモードと同じ遅延時間で発音している、という事である。これはアサイナを開発した事のある者であれば驚嘆すべき性能であるが、逆に言えば「馬鹿正直にはアサイナ動作を行わない」という指針のアルゴリズムであると思われる。つまり、「ベロシティ=1」というような、実質的には発音しても聞こえない情報については、音源処理として正直に扱わずに無視して捨ててしまう、という方針を取ることににより、MIDI割り込みの直後の判定処理で捨てることで音源としてはシングルトリガモードとほぼ同等の性能を実現しているものと思われる。これをうまい手であると見るか、ずるい手であると見るかは、開発者の哲学と良心の問題である。

3.2.4 高速に対応する内蔵音源チップ

ここでは、<15>,<19>が対象である。Windowsノートパソコンに内蔵している標準音源チップOPL3による発音は、シングルトリガモードにおいてもトラフィックモードにおいても、驚異的に高速対応し、発音漏れもゼロと好成績となった。<15>と<19>の発音時間の差は、MIDIインターフェースとしてシリアルポートを経由するか、PCMCIAバスによりパラレル転送するか、という差であると思われるが、いずれにしてもリアルタイム性としては最高の性能である。

3.2.5 成績優秀なWavatableシンセ群

ここでは、<8>,<12>,<16>,<20>が対象である。この音源に共通する点としては、(1) シングルトリガモードでは古典的な電子楽器レベルの発音遅延(数10ms-100ms台)がある、(2) トラフィックモードでも発音抜けすることは無く、遅延もシングルトリガモードに近い、という事である。これは「たまに発音しない」というハード音源よりもある意味では信頼できるものであり、遅延のばらつき幅も小さいために、この遅延量を把握して組み込んだ実験においてはかなりの正確性を確保できるものである。この遅延のばらつき小ささは、共通するWavatable音源方式に理由があると思われるが、トラフィックモードでの振る舞いから、「ベロシティ=1」などの情報を捨てる、という現実的なアルゴリズムも活用していると思われる。

3.2.6 SGMPソフトシンセ

ここでは、<10>,<14>,<17>が対象である。この音源の特徴としては、シングルトリガモードであってもトラフィックモードであっても、約400msと発音開始が異常に遅い、ということであるが、逆に発音遅延のばらつきは非常に小さい、という長所も持っている。また、<21>の実験結果だけは矛盾しているので追試が必要であるが、トラフィックモードであっても決して発音漏れを起こさない、という点にも注目したい。この音源はつまり、リアルタイムにMIDI情報を受け取って鳴らす、という用途を主目的にしているのではなくて、SMFのようなシーケンスデータを「あらかじめ先読み」する、という自動演奏ソフトの音源ドライバとしての機能を優先している、と解釈すると納得できる。ちょうど「MIDI演奏アコースティックピアノ」(ソレノイドでアクションを駆動するタイプ)と同様に、MIDI入力からの遅延が一律に400msec、というように固定している場合には、演奏の同時性を実現するためには、このピアノのパートは400msecだけ先にMIDI送信すればいいわけである。

3.2.7 その他

<7>と<11>は、Windowsパソコンに内蔵されたFM音源チップであるが、シングルトリガモードでは高速対応が光ったものの、トラフィックモードではまったく発音せず、役に立たないという結果になった。また<9>と<13>のVSC55ソフトシンセについては、SGMPソフトシンセと似たような大幅な遅延があるものの、発音しない事が多く、ここでの成績は非常に悪かった。これは、計測に使った発音イベントのON-OFF間隔の狭さにも起因するかもしれないが、同一条件の他の音源の成績と比較すると見劣りするのも事実である。<24>は、Unixワークステーションでは唯一、MIDI関係で使いものになる、というIRIX(カーネルからリアルタイム性能を強化)での実験結果である。シングルトリガモードにおいては、非常に優秀な成績であったが、対照的にトラフィックモードでは最悪の状態となった。これは、多量のMIDI入力に対してはIRIXカーネルから凍ってコアダンプする、というもので、この現象は筆者はこれまでの研究 / 作品応用の中で把握していたものである。

4. 発音遅延とばらつきの影響

上記の実験結果のような特性を持つMIDI音源を音楽心理学の実験の道具として使用した場合、文献[1]にあるように、人間の聴覚と音楽心理学的な影響から、発音遅延やばらつきによって、実験の意味が消滅する危険性がある。つまり、もともと正確な「同時性」の概念の消滅しているMIDIの世界において、さらに実験データをブラックボックスのシーケンサとMIDI音源に委ねた場合には、音楽心理学的に意味を持つ(感性評価として有意な差がある)オーダの時間的誤差がある、というのは、実験の意味を喪失させることになりかねない。パソコンとシーケンサソフトとMIDI音源(楽器メーカによるハード的な製品であっても、パソコンの内蔵音源チップであっても、ソフトシンセなどソフト音源であっても)、というお手頃な道具の普及は、音楽心理学実験をあまりコンピュータの得意でない人々にまで広げる貢献をしているものの、ブラックボックス故の上記の問題点によって研究の意義を脅かすとなれば大きな問題であろう。

文献[1]によれば、人間が「同時に鳴る」3和音を聴取した際に、5msecとか10msecとかのオーダの発音遅延を特定の音に作用させただけで、和音全体の感覚的な印象が異なる、とされている。もし、ここに発音遅延の絶対値が20msecのオーダでばらつく音源を使用した場合、シーケンサのソフト上では「同時」と思いこんでいる実験データが、実験の意味を失わせるオーダで効いていることは明白であるが、シーケンサと市販音源を盲目的に使用している研究者は、この事実気付かない、という可能性がある。カラオケのBGMを自動再生している場合にはまったく問題ないことであるが、音楽心理学の実験にはそうそう気楽に使ってはいけぬ、という事実を認識すべきである。(文献[1]の実験には、当然のことながらMIDIシーケンサ等は使用せず、正確なタイミングの和音の音響波形を編集して、DATで再生して聴取実験している)

5. 音源アルゴリズムの検討

今回の実験から浮上したもう一つの重要な問題提起は、音源アルゴリズムについてである。基本的には音源DSPの演算スロットによって同時発音演算チャンネル数の制限があるハード音源と、ソフトウェアによって理論的には発音数の制限のないソフトウェア音源とを同等に比較することはできないが、発音遅延、そのばらつき、MIDIトラフィックが多い場合の対応、といった今回の

実験から得られた状況に対応した音源アルゴリズムの改良の余地は十分にあると思われる。以下の4つの図は、4音ポリフォニック音源のシステムで既に4音の発音が継続しているところに「第5の発音イベント」がMIDIから到来した場合の典型的なアサイン(発音割当て)のアルゴリズム例である(<http://nagasm.org/hightech/03-03>を参照のこと)。このような古典的なアルゴリズム(とその特許)に限定されることなく、今回の実験で一部に見られたように、これからのアルゴリズムとしては、たとえば「非常にペロシティの小さな発音イベントはアサイン以前にカットする」「同時性の確保のために、音源全体の発音遅延を大きくする代わりに遅延のばらつきを無くす」などの戦略も検討すべきであろう。同時に、パソコンにおいてはMIDIポートの遅延やバッファリング特性に関する仕様が公開され、安心して実験に使用できるシーケンサの登場に期待したい。

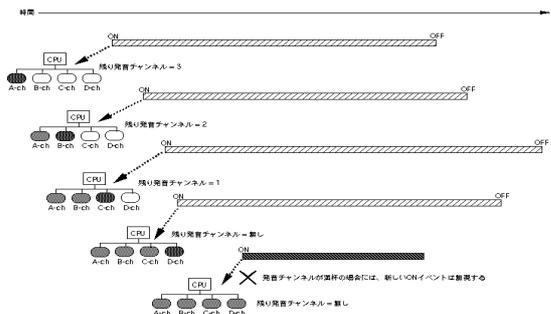


Fig. 12

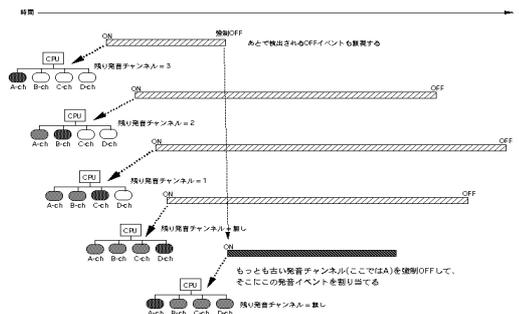


Fig. 13

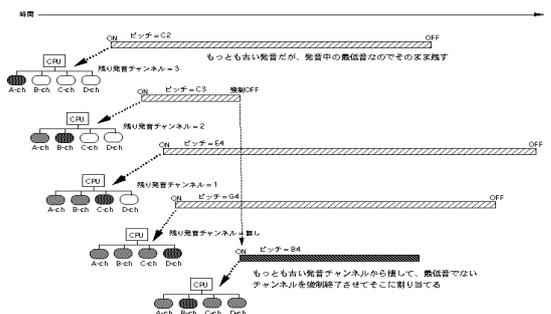


Fig. 14

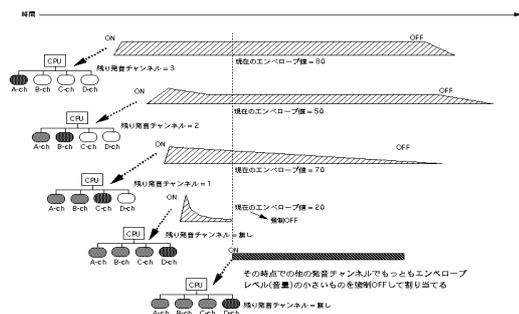


Fig. 15

6. おわりに

MIDIは既に約15年の歴史のあるComputer Musicにおけるスタンダードであるが、SCSI2やZIPIやUSBやRMCPやIEEE1394やインターネットMIDIストリーム等が登場しても、SMFやGMとしての音楽演奏データの規格としては消えることはないだろう。そして同時に、31.25kbpsという低速ながら、ある種の音楽心理学実験や人間の音楽演奏情報伝達としては十分に活用できるために、リアルタイム情報プロトコルとして生きていくと思われる。その特性と課題を十分に理解し活用して、意義ある関連/応用研究を進めていきたいものである。

(参考文献)

- [1]梅田・山田・北村「ピアノの長3和音の各音の発生時間による変化」、日本音響学会講演論文集、平成元年3月
- [2]長嶋「ハード音源/ソフト音源のMIDI発音遅延について」、情報処理学会平成11年度前期全国大会講演論文集、1999.
- [3]長嶋「ハード音源/ソフト音源のMIDI発音遅延と音楽心理学実験環境における問題点の検討」、電子情報通信学会平成11年度前期全国大会講演論文集、1999.
- [4]長嶋「MIDI音源の発音遅延と音源アルゴリズムに関する検討」、情報処理学会研究報告 Vol.99, No.68 99-SIG-31、1999
- [5]長嶋洋一「作るサウンドエレクトロニクス」、<http://nagasm.org/hightech/02-11/>
- [6]長嶋洋一「コンピュータサウンドの世界」、CQ出版
- [7]長嶋・橋本・平賀・平田編「コンピュータと音楽の世界」、共立出版