

音響信号のループ再生条件に関する研究*

○安部綾太, 木村敏幸（東北学院大学）

1 はじめに

従来の自動伴奏システムでは人間がコンピュータの演奏に合わせてきた。そこで、我々はコンピュータが人間のフットペダルの操作に合わせて演奏の開始や停止を行う自動伴奏システムの実現を目指し、これまでにフットスイッチとMIDIのホールドオン機能を用いることで人が操作をするまで音が鳴り続ける自動伴奏システムを開発し[1,2]、さらに音響信号を用いたシステムの開発にも取り組んでいる[2,3]。

音響信号はMIDIのようにファイルの最後や最初にホールドオン、オフのMIDIコントロールチェンジメッセージを入れることができないため、オーディオファイルの最後の音を鳴り続けさせることができない。従って、MIDIにおけるホールドオンの状態を再現するためには、オーディオファイルの再生後にオーディオファイルの最後のごく短い時間をループ再生させることが必要となる。

しかしながら、ループ再生をする際に問題となる点は、ループの終わりと始まりが切り替わる際に急激に時間波形が変化することによってクリックノイズが発生することである。

そこで、本研究ではクリックノイズが発生せずループ再生される条件を検討し、どの条件が自動伴奏システムに適用可能かを確認するために評価実験を実施する。

2 自動伴奏システム

2.1 原理[2]

Fig. 1 に提案する自動伴奏システムの構成図を示す。あらかじめ演奏者の足元にフットペダルを配置しておき、演奏者は演奏しながらフットペダルを操作する。その結果、コンピュータは演奏者のフットペダル操作に合わせて伴奏するので、これまでの自動伴奏システムでは実現できなかったライブ演奏（例えば、曲の終わりで人間の合図に合わせてコン

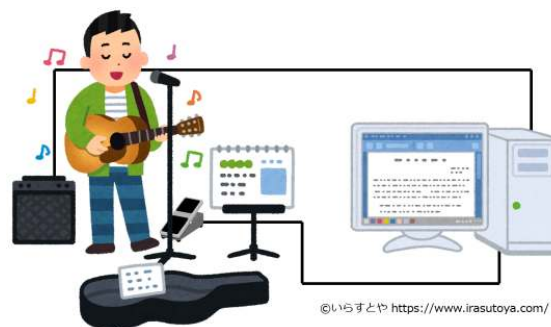


Fig. 1 自動伴奏システムの構成図 [2]

ピュータが同時に演奏を停止する）が可能になると期待される。

2.2 開発環境

Fig. 2 に音響信号を用いたシステムの開発環境を示す。デスクトップPCに2台のディスプレイと2台のUSBフットスイッチ（Route-R RI-FP3MG & Route-R RI-FP3BK）が接続されており、PC内蔵のサウンドカードを通して演奏音が再生される。その際、サウンドカードをASIOインターフェースとして認識させるためにASIO4ALLを使用し、プログラミングソフトウェアにはMax [4] Ver. 8.1.3を用いた。



Fig. 2 音響信号を用いた自動伴奏システムの開発環境

*Study of Loop-playing Conditions of Audio Signals, by ABE, Ryota and KIMURA, Toshiyuki (Tohoku Gakuin University).

3 ループ再生条件

3.1 従来法

まず、先行研究[2,3]でも用いたように、Max に装備されている groove オブジェクト[5]を用いたループ再生条件を設定した。groove オブジェクトはバッファに格納されたオーディオファイルをループ再生するためのオブジェクトであり、開始時間と終了時間を指定するとその時間の間だけループ再生される。本研究では、ループの開始時間と終了時間の時間波形が急激に変化しない個所を手動で指定した。

3.2 Max を用いたクロスフェード法

しかしながら、従来法によってループの開始時間と終了時間を手動で指定する場合、時間の指定に非常に手間がかかる。そこで、次にクリックノイズが発生しないループ再生方法として、クロスフェードを用いたループ再生条件を設定した。

ループ個所の終わりの数 ms をフェードアウトすると同時にループ個所の始めの数 ms をフェードインすることにより、クリックノイズが発生しないループ再生が期待される。本研究では直線に変化する三角窓、曲線に変化する cos 窓の 2 種類の窓関数を用いて窓掛けを実施し、二つのオーディオファイル再生を順次切り替えることによりループ再生を実現した。

Max を用いた窓掛けでは、Max に装備されている line オブジェクト[5]を用いて窓掛けを実装した。line オブジェクトは音量の目標値と到達時間を入力すると、現在の音量から目標値に向けて到達時間をかけて線形に変化するオブジェクトである。従って、入力する音量の値を常に 0 もしくは 1 にすることで三角窓、入力値を cos 関数で変換すれば cos 窓を



Fig. 3 Max を用いたクロスフェード法
(左：三角窓，右：cos 窓)

実装することができる。Fig. 3 に Max を用いたクロスフェード法のプログラムを示す。クロスフェードに要する到達時間は手動で 1 ms ずつ調整していき、自分で聴いて最もクリックノイズを感じなかった 52 ms とした。

3.3 Octave を用いたクロスフェード法

Max を用いたクロスフェード法では、二つのオーディオファイル再生をタイミング良く切り替えることができればクリックノイズが発生しないループ再生を実現できるが、実際はフェードアウトと同時にフェードインが始まらない。そこで、Octave [6]を用いてオーディオファイルのクロスフェードを波形処理したうえで、groove オブジェクトで再生するループ再生条件をさらに設定した。

Fig. 4 に Octave を用いたクロスフェード法のソースコードを示す。Octave では 1 ループの長さにクロスフェードする時間を足したオーディオファイルの両端に窓掛けを行い、フェードアウトした部分を切り出してフェードインした部分に足し合わせることで本来の 1 ループの長さになるよう調整している。クロスフェードに要する到達時間は 3.2 節と同様に 52 ms とした。

4 評価実験

4.1 実験手順

実験は Max で実験用のパッチを作成して行った。音源には 500 Hz の sin 波と通常の曲の 2 種類を用いた。通常の曲としては、星野源のアイデア[7]の曲の最後尾を Domino [8]で MIDI として打ち込み、再生・録音してオーデ

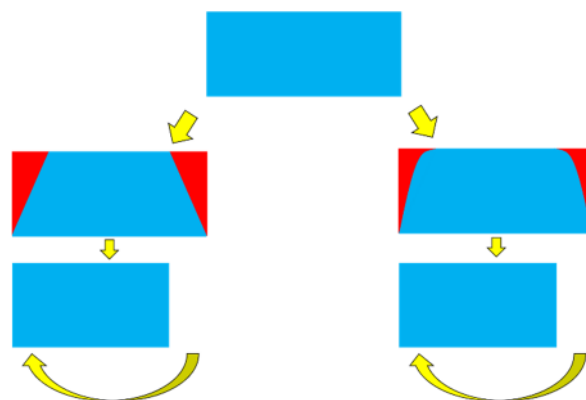


Fig. 4 Octave を用いた
クロスフェード法のイメージ
(左：三角窓，右：cos 窓)



Fig. 5 実験手順

ィオファイルとしたものを使用した。また、再生方法は以下の6種類を用いた。

- 従来法
- Max 使用クロスフェード法 (三角窓)
- Max 使用クロスフェード法 (cos窓)
- Octave 使用クロスフェード法 (三角窓)
- Octave 使用クロスフェード法 (cos窓)
- ループなし

ループなしの場合では10秒間のファイルをそのまま再生し、その他の場合では1秒間のファイルをループ再生することにより10秒間再生した。

Fig. 5に実験手順を示す。実験参加者は4名である。実験参加者にはまず実験手順を説明し、ループなしの音源を2回聴いてもらった。ここで実験手順と正確な音を理解できたら実験に入る。試行では、6種類の再生方法のうちのいずれかを用いた音が10秒間再生される。実験参加者は音源が再生された後の5秒の間にTable 1に示す5段階で評価する。1セットは6試行で、セット内には6種類の再生方法が必ず1回ずつ含まれる。再生方法の提示順番はセットごとにランダム化した。本実験では音源ごとに5セットずつ実施した。音源の提示順番は実験参加者ごとにランダム化した。

Table 1 評価実験における5段階評価

| | |
|---|----------|
| 5 | 全く気にならない |
| 4 | 気にならない |
| 3 | 少し気になる |
| 2 | 気になる |
| 1 | とても気になる |

4.2 sin波の実験結果

500 Hzのsin波を音源として用いた時の平均評定値をFig. 6に示す。Anova4 on the Web

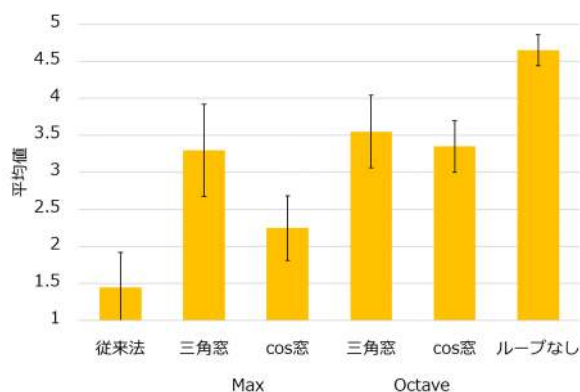


Fig. 6 実験結果 (sin波)

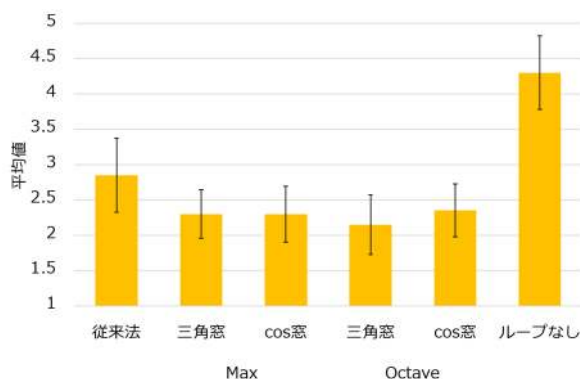


Fig. 7 実験結果 (通常の曲)

[9]で6水準の一要因分散分析を実施したところ、主効果に有意差が見られた。さらに多重比較を実施したところ、Max使用クロスフェード法(三角窓)、Octave使用クロスフェード法(三角窓)、Octave使用クロスフェード法(cos窓)の3種類の再生方法以外の全ての組み合わせにおいて5%水準で有意差が見られた。

分散分析の結果から、sin波をループ再生する場合、Octaveを用いたクロスフェード法を適用するのが良いと考えられる。但し、Maxを用いたクロスフェード法の場合でも三角窓のときは評定値が下がっていないことから、クロスフェードの設定の手間を考慮すると、Maxを用いたクロスフェード法において三角窓を掛ける方法が最も妥当であると考えられる。

4.3 曲の実験結果

通常の曲を音源として用いた時の平均評定値をFig. 7に示す。Anova4 on the Webで6水準の一要因分散分析を実施したところ、主効果に有意差が見られた。さらに、多重比較を実施したところ、ループなしとそれ以外のいずれかの組み合わせにおいてのみ5%水準で

有意差が見られ、ループ再生方法同士には有意差が見られなかった。

曲の最後尾を用いた音源では、10秒間集中して聴くことでいずれのループ時間でもループ部分に違和感を覚え、評価としては1~3としたためにいずれの方法でも有意差が見られなかったと考えられる。しかしながら、実際に自動伴奏システムを用いて演奏を行う際には演奏自体に集中するため、ここまでの違和感を覚えない可能性がある。また、grooveオブジェクトにおけるループの開始時間と終了時間は手動で調整するため、用いる音源に応じてより適した時間を指定できれば、MIDIにおけるホールドオンの状態により近づけることができると考えられる。

5 おわりに

本研究ではフットスイッチの操作に合わせて演奏の開始や停止を行う自動伴奏システムを提案し、そのシステムを開発するための前段階として音響信号をループ再生する際の条件を検討した。その結果、sin波をループ再生する時は三角窓を掛けるとクリックノイズを軽減できることが分かった。また、通常の曲をループ再生する時は有意差こそ出なかったものの、従来法であるgrooveオブジェクトを用いてループ再生する方法に検討の余地がありそうな結果となった。

今後は音響信号を用いた自動伴奏システムを完成させ、そのシステムの評価実験を行う予定である。その評価実験でもMIDIを用いたシステムと同様に従来法より提案法が有意に優れている結果を出せれば、音響信号を用いたシステムも有効であると考えられる。

6 参考文献

- [1] 安部綾太, 木村敏幸, "ペダル操作による自動伴奏システムの評価," 平成31年東北地区若手研究者研究発表会講演資料, No. YS-17-P30, pp. 141-142 (2019).
- [2] 安部綾太, 木村敏幸, "フットスイッチを用いた自動伴奏システムの検討," 第2回東北地区音響学研究会発表資料, No. 2-7, pp. 1-4 (2019).
- [3] 安部綾太, 木村敏幸, "音響信号に対応したフットスイッチ式自動伴奏システムの評価," 令和2年東北地区若手研究者研

究発表会講演資料, No. R2-A-12, pp. 23-24, (2020).

- [4] Max, <https://www.mi7.co.jp/products/cycling74/>.
- [5] ノイマンピアノ(赤松正行+佐近田展康),"2061:Max オデッセイ",古森優(編),株式会社リットーミュージック,(2012).
- [6] Octave, <https://www.gnu.org/software/octave/>.
- [7] 青木紀, バンドスコア「アイデア/星野源」, 株式会社フェアリー, 東京 (2018).
- [8] Domino, <http://takabosoft.com/domino>.
- [9] Anova4 on the Web, <https://www.hju.ac.jp/~kiriki/anova4/>