

ギターにおけるコード列演奏の最適化に関する手法

澤山 康二*, 江村 伯夫*, 三浦 雅展**, 柳田 益造*

*同志社大学 工学部 **龍谷大学 理工学部

コード列演奏はギター演奏において最も基本的な演奏形態の一つである。本研究では、初心者がギターでコード列演奏をする際の負担を軽減するために、与えられたコード列に対して演奏の総負担量が最小となるようなコードフォーム系列を決定するシステムを提案している。システムは、すべての押弦パターンから物理的に押弦が不可能なものを削り、指配置や運指のしやすさなどの評価項目についての負担の程度を実演奏時のミスの量から最小二乗法によって求め、その項目別の負荷推定値に基づいて、コードフォームの候補を絞り込むことによって実現されている。

Obtaining Optimum Performances of Chord Sequences on Guitar

Koji SAWAYAMA*, Norio EMURA*, Masanobu MIURA**, and Masuzo YANAGIDA*

*Faculty of Engineering, Doshisha University, Kyoto

**Faculty of Science and Technology, Ryukoku University, Shiga

Playing chord sequences is a basic style in playing the guitar. Proposed is a system which yields a chord-form sequence of minimum performance load to a given chord sequence for novice players.

Discarding physically impossible chord-form patterns, the system selects candidates of chord-form patterns of minimum performance load predicted as a linear sum of load factors obtained by least-squares estimation counting mistakes in preliminary performance.

1. はじめに

コード列演奏はギター演奏において最も基本的な演奏形態の一つである。ポピュラー音楽系のギター演奏を目指す初心者の大半は、まずコードを押さえることからギターを始めるのが現状であるが、初心者の多くがこのコード演奏を練習する段階で挫折してしまう。その理由として、初心者にとって押弦の困難なコードフォーム(指板上の指の使い方、押さえ方)を教則本が強制していることが考えられる。

ギターは一つの音高に対して複数の奏法、すなわち指板上で複数の押弦位置が存在する楽器であるので、一つのコード(和音)に対して一般にかなり多数のコードフォームが存在する。各コードに対

するコードフォームが列挙されているものとして“コードブック”[1-4]とよばれる書物が存在する。コードブックにはそれぞれのコードに対してコードフォームが複数通り記載されており、ギターの初心者や中級者は自分が演奏したい曲のコード進行に合わせて、そのコードフォームを、(自ら考案することなく)コードブックの中から拾ってくることになる。ただ、コードブックには、音色等の音楽的要素を重視したコードフォームも多数記載されており、これらは必ずしも押弦が容易でなく、初心者が最初にコードを練習する題材としては適切でない場合がある。また、初心者がコードブックから拾ってきたコードフォームは、連結して演奏するのには適していない、つまり隣接して弾くには負担の大きいものである場合が

ある。

このような状況の中で、近年初心者を対象としたコード演奏習得のための練習を支援する文献やソフト[5][6]が普及しているが、これらの練習支援ツールの中に、「押弦やコードチェンジのしやすさ」を追求したものは見当たらない。本研究では、ギター初心者を対象として、コード列演奏をより簡単にするため、与えられたコード列に対して「押弦やコードチェンジのしやすさ」を最優先したコードフォーム系列を出力するシステム YG(You are never afraid of a Guitar)を提案する。

YG は、各コードに対して、考えられる可能なコードフォームを全て探索し、その中からコードチェンジ時に生じる手の位置および指の動きに対する負荷が最も軽くなるコードフォーム系列を実演奏時のミス量から最小二乗法によって求めた負荷値に基づいて出力するシステムである。この YG による出力を利用することで、連続するコードをユーザが最も少ない負担で演奏することが可能となり、初心者のギター練習の負担を軽減することが期待できる。前回の報告[7]では孤立コードの押弦しか扱っていなかったが、本報告はそれをコード列演奏に拡張するものである。

2. 最適コードフォーム系列の決定方法

すべてのコードについて押弦可能な全てのコードフォームを列挙しておき、与えられたコード列に対しては、その押弦パターンとしてのコードフォームの中から押弦やコードチェンジ時に生じる総負荷が最小となるコードフォーム系列を選択する。押弦やコードチェンジ時に生じる総負荷は、個々の指にかかる負担の線形和になるとモデル化し、その押弦とコードチェンジに関する個々の困難要素にかかる負荷の程度を少数のコードフォーム系列に対する演奏データから推定しておき、それを他のコードフォーム系列を演奏する際の総負荷の予測に用いる。

2.1 押弦可能なコードフォームの列挙

2.1.1 押弦可能コードフォームの絞り込み

押弦可能なコードフォームを全て調べ上げるために、まず各コードに対し人間が押弦可能かどうかに関わらず全てのコードフォームを列挙する。すなわち、ギター指板上の押弦位置の全組み合わせを列挙する。その中から、ギター演奏上で音楽的な要素を満たしているか(例えば「四和音の場合、5 度音以外の 3 音を全て含むか」など)どうかをチェックするための条件を定めたルール”rule C(Chord)”，及び人の指の物理構造上、押弦が可能であるかどうかを検定するための条件(例えば「1 本の指でフレットを越えて複数弦の押弦を要求しない」など)を定めたルール”rule F(Fingering)”を適用することにより、押弦可能なコードフォームの絞り込みを行う。

2.1.2 絞り込み結果

12 種の階名音に対してそれぞれ”M(メジャー)”, ”m(マイナー)”, ”7(セブンス)”, ”M7(メジャーセブンス)”, ”m7(マイナーセブンス)”の 5 種類で、計 60(=5×12)通りのコードを対象とし、それらのコードについて存在し得る全てのコードフォームから前述の rule C 及び rule F を適用し、最終的に押弦可能なコードフォームに絞り込んだ結果を表 1 の提案法の列に示す。

これとは別に、市販のコードブック 4 冊[1~4]について、対象とする 60 種類のコードを調査した結果、全コードに対するフォームの合計値は 432 であった。つまり、一つのコードに対して平均 7.2 通り(最大 11, 最小 5)のコードフォームが記載されていることになる。コードブックを調査した結果については、さらに rule C に合格したコードフォーム数の平均値も調査した。その結果、コードブックには一つのコードに対して平均 7.2 通りのコードフォームが記載されていたが、「最低音がルート音ではないコードフォーム」などの rule C を満たさないコードフォームが多数存在するため、rule C に合格したコード

フォームは一つのコードに対して平均 2.2 通りとなった。これを表1の「コードブック」の列に併せて示す。表1より、提案システムで全検索によって列挙されるコードフォームの数がコードブックに記載されているものより多く、負荷の少ない演奏のための押弦位置決定には、検索漏れを防ぐために全検索を行う必要があることが明らかになったといえる。

表1 一つのコードに対する平均コードフォーム数

	提案法	コードブック
“rule C”	8.5 ($\sigma=6.4$)	2.2 ($\sigma=1.2$)
“rule F”	4.3 ($\sigma=3.7$)	※

※“rule F”はフィンガリングに関する条件であるが、コードブックにはフィンガリングの情報がないため、コードブックでの“rule F”による絞り込みは調査不能である。

2.2 コードフォームのローポジション化

コード列演奏におけるコードチェンジ時には、左手首の移動や指の配置などの負荷が演奏者にかかることになる。2.1 節で押弦可能なコードフォームを列挙したが、それらの中にはハイポジション(高いフレットでの押弦位置)のコードフォームが含まれている。そこでコードチェンジ時に生じる手首の移動量を最小限に押さえるための基準として、「コードフォームのローポジション化」を設定する。これは、複数存在するコードフォーム候補群のうち、ローポジションのコードフォームを選択する手法であり、YGでは、コードフォームに含まれる全押弦位置の中で、最高フレットとなる押弦位置が第7フレット以下であればローポジションであると考え。これにより開放弦の使用頻度が高くなるため押弦負荷が少なくなり、またコードチェンジ時の手首の移動量が減少し、演奏の負担を軽減することが期待できる。

2.3 最適コードフォーム系列の決定

各コードに対して可能なコードフォームを列挙し、コードフォームのローポジション化を行った後、それらの中から、コード列演奏において左手の各指

や手首にかかる様々な負荷を考慮し、それらの線形和によってコードフォーム系列の総負荷が決まるというコード列演奏に関する「線形負荷モデル」に基づいて、最適なコードフォーム系列を決定する。コード列演奏における各負荷要素の負荷の重さは、線形負荷モデルに基づいて演奏実験から非負条件(負荷要素は弾き易さにはつながらない)の下で最小二乗推定するものとする。

2.3.1 コード列演奏時に生じる負荷

コード列演奏時に生じる負荷と考えられる項目は、次の7つである。

- (i) 単一弦の押弦
- (ii) コードチェンジ時の各指のフレット位置移動量
- (iii) コードチェンジ時の各指の弦間移動量
- (iv) コードチェンジ時の各指の追加
- (v) セーハ*の使用頻度

*複数の弦を一本の指で押弦する押弦法のこと

これらの負荷と考えられる各項目にそれぞれ負荷値(負荷の程度)を未知数として実際のコード列演奏に基づいて推定する。負荷値はユーザによって異なるが、その平均的な値の推定は多数の奏者による演奏実験に基づいたデータから最小二乗推定できる。特に必要な場合のみ、特定の奏者に関する負荷値を求めるということになる。

3. コード列演奏時の負荷の検討

コード列演奏時、奏者の左手には様々な負荷がかかっている。コード列演奏の総負荷 \mathbf{b} はそのコードフォームに含まれる負荷要素(詳細は後述)の出現頻度ベクトル \mathbf{d} と負荷値ベクトル \mathbf{w} の内積として次のように定式化できる。ただし、2.3 節より \mathbf{w} の各要素は正とする。

$$\mathbf{b} = \mathbf{d}^T \mathbf{w} \quad (1)$$

最適なコードフォーム系列を決定するためには、押弦可能なコードフォームからこの総負荷 \mathbf{b} が最

も小さいものを検索しなければならない。そこで、コード列演奏時に生じる負荷要素(2.3.1 節の(i)~(v))に関する各指にかかる負荷)に関する重み w を最小二乗法によって求めることを考える。

3.1 コード列演奏における負荷要素

コード列演奏における負荷項目と各指に対する要素負荷値(未知数 $w_1 \sim w_{27}$)を表 2 に示す。

表 2 各負荷項目に対する各指への要素負荷値

負荷の項目	人差指	中指	薬指	小指
単一弦の押弦	w_1	w_2	w_3	w_4
コードチェンジ時のフレット位置移動	w_5	w_6	w_7	w_8
コードチェンジ時の弦間移動	w_9	w_{10}	w_{11}	w_{12}
コードチェンジ時の押弦指追加	w_{13}	w_{14}	w_{15}	w_{16}
2弦セーハ(1-2弦隣接に限る)	w_{17}	w_{18}	w_{19}	w_{20}
3弦セーハ	w_{21}	w_{22}	w_{23}	w_{24}
4弦セーハ	w_{25}	—	—	—
5弦セーハ	w_{26}	—	—	—
6弦セーハ	w_{27}	—	—	—

3.2 要素負荷値の推定方法

表 2 で示した要素負荷値($w_1 \sim w_{27}$)を推定するために、式(1)を重ねて行列で表現しておく。

$$D w = b \quad (2)$$

$$\left(\begin{array}{l} D = [d_1, d_2, \dots, d_i, \dots]^T \\ w = [w_1 \dots w_{27}]^T \\ b = [b_1, b_2, \dots, b_i, \dots] \end{array} \right)$$

行列 D は、演奏に要求されるコードフォーム i に対して各負荷要素がいくつ含まれるかを示す d_i を重ねて行列にしたものである。また b は、コードフォーム列に対して演奏した時の失敗した弦の本数を示す b を並べて、ベクトルにしたものである。例えば、与えられたコードフォーム列で演奏する際の鳴るべき弦が 6 本でそのうち 4 弦しか鳴らなかったら“ $b=2$ ”, 鳴るべき弦が 5 本(1 本はミュート)でそのうち 1 本も鳴らなかったら“ $b=5$ ”となる。鳴るべき弦がすべて撥弦されたときは成功で“ $b=0$ ”となる。また、ミュートするべき弦が鳴ってしまった場合、その弦は失敗となる。 w の最小二乗推定値 \hat{w} は、

$$\hat{w} = (D^T D)^{-1} D^T b = D^+ b$$

$$w_i \geq 0 \quad (i = 1 \sim 27) \quad (3)$$

として求められる。ただし、 D^+ は D の最小二乗型一般逆行列である。 w の最小二乗解 $\hat{w} = (\hat{w}_1, \hat{w}_2, \dots, \hat{w}_{27})^T$ の要素の値は、表 2 の対応する各項目の負荷量の相対値であり、値が大きいほど演奏者にとって演奏困難であることを意味する。また、要素負荷値が負の値となつてはいけないので、 \hat{w} の各要素は正とする。

4. コード列演奏による負荷値の推定

コードフォーム列を被験者に演奏してもらい、連続した n 個のコードフォームのうちその後 $n-1$ 個の各コードに対する各弦の押弦判定の結果を求める実験を行う。これにより、用いた各コードフォームについての負荷要素の出現頻度行列 D と b から式(3)を用いて負荷ベクトルの最小二乗推定量 \hat{w} を非負条件の下で求める。前回の報告[7]では、要素数が 16 であったが非負条件を考慮せずに解いて、1 つの要素だけわずかに負の値をとつただけであった。従つて強制的に非負条件を入れなくても実験データに矛盾がなければ各要素負荷値は正の値として求まると考えられる。

4.1 要素負荷値を推定するための実験

4.1.1 被験者

本実験では、初心者と中級者の定義をするため、まず被験者に 4.1.2 節の 21 パターン(後述)以外の 2 パターンのコードフォーム列を用意し、4.1.3 節の方法(後述)で演奏してもらった。その際の押弦失敗弦数が予め設けた閾値よりも多ければ初心者、少なければ中級者とした。その結果、本実験の被験者は初心者 2 名、中級者 1 名となった。

4.1.2 演奏課題

上記で $n = 4$ つまり、4 つの連続したコードフォームを 1 パターンとし、計 21 パターンのコードフォー

ム列を作成し、演奏課題とした。例として、図 1 に 1 パターンのコードフォーム列を示す。

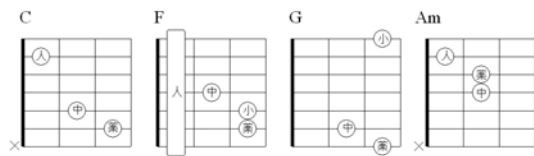


図 1 1 パターンのコードフォーム列

×は開放弦を、縦長の長方形はその中に書いた指によるセーハ(一本の指による同時複数押弦)を表す。

4.1.3 奏法の指定

実験では、被験者に 4.1.2 節で説明したような演奏課題を 21 パターン MIDI ギターを用いてコード列演奏を行ってもらった。

まず被験者に 4 つのコードフォームを覚えてもらうために練習時間として 2 分を与え、1 つのコードあたり一度だけ、ピックを用いて 6 弦から 1 弦方向に弾いてもらった。演奏の際、被験者にはまず初めのコードフォームで弦振動を 2 秒間持続してもらい、その後 3 秒以内でコードチェンジを行い、続いて 2 つ目のコードフォームで弦振動を 2 秒間持続してもらうという形で、同様の演奏を 4 つの目のコードフォームでの 2 秒間の弦振動が終了するまで行ってもらった。

4.1.4 演奏の評価条件

4.1.3 節で採取した演奏データ(MIDI)から後ろ 3 つの各コードの各弦の振動の持続時間が規定以上あるかを調べて発弦判定を行い、演奏ミス数を b に入れる。

4.1.5 奏法の指定と評価条件の根拠

弦振動を 2 秒間持続してもらったのは、正しく押弦されていない場合でも 1 秒程度は弦の振動が持続することがあり、押弦判定を正確に行えないためである。3 秒程度を条件にすると、弦の振動が自然

減衰してしまうため、正しく弾けた場合でも弾く力が弱いと 3 秒間弦振動が続かず、正確に各弦の押弦判定を行えないからである。また、コードチェンジに要する時間を 3 秒にしたのは、これ以上短くすると初心者は次のコードフォームを押さえることができず、ほとんど失敗してしまい、意味のあるデータが採れないからであり、逆に 3 秒より長くすると、全指を一度指板から離してから次のコードフォームを押さえようとするため、運指の負荷値が求まらないからである。

4.2 各弦の押弦判定基準

4.1.4 節の発弦判定を行うために、「このように発音していれば押弦は成功」という、その弦が正しく押弦されたのかされなかったのかの基準を決めておく。

演奏の際、被験者には弦振動を 2 秒間持続してもらったが、実際の演奏では、被験者は弦振動を 2 秒弱で止めてしまう場合があるため、押弦判定基準を 2 秒よりやや短い 1.5 秒に設定した。つまり、弦の持続音が 1.5 秒未満であればその弦は十分押弦されていないとし、1.5 秒以上音が持続されれば十分押弦されたと判定した。また、弾かなければならない弦がまったく発音されなかった場合や誤った位置を押弦してしまった場合は、その押弦は失敗とした。

4.3 w の推定

当該コードフォーム列に現れる 2.3.1 節の各項目、つまり表 2 の第 i 負荷項目の出現頻度を d_i 、各コードにおける失敗と判定された弦の数を b とし、 d_i ($i = 1 \sim 27$) と b を実験から得られた各演奏データから入力することによって、式(3)から非負となる w の最小二乗解を求めた。またその際、被験者固有の w の値と被験者全員をまとめた w の値を求めた。

5. 動作評価

4.3 節で求めた負荷値の推定値 \hat{w} を使っていくつかのコード列について本システムが出力するコードフォーム列と、市販されているコードブックから得られたコードフォーム列を 4.1.1 節の被験者 1 名の実演奏データに基づいて比較する。

5.1 コード列の選出

まずポピュラー音楽の曲中によく現れる 4 つの連続したコード列を 1 パターンとし、その各コードに対してコードブック推奨コードフォームを割り当てる。また、そのコード列を本システムに入力することで被験者固有の負荷値を用いて出力したコードフォーム列と、被験者全員をまとめて求めた負荷値を用いて出力したコードフォーム列を得る。これら 3 つのコードフォーム列が全て異なればこのコード列を採用する。このようにして計 11 パターンのコード列(1 パターンあたり 3 通りのコードフォーム列)を選出した。ここで言うコードブック推奨コードフォームとは、コードブック[1-4]に掲載されているコードフォームの中で、各コードごとの最も多く掲載されているコードフォームのことである。

5.2 比較実験

5.1 節で選出された 11 パターンのコード列について、それぞれ 3 通りずつのコードフォーム列を被験者 1 名に 4.1.3 節と同じ奏法でそれぞれ演奏してもらい、4.2 節と同じ方法で押弦判定を行った。ただし、この比較実験では本システムが出力するコードフォームの押弦のしやすさを調べるため、連続する 4 つのコードフォームのうち一番目のコードフォームにも押弦判定を行った。

その結果を図 2 に示す。

6. まとめと今後の課題

本論文では、押弦およびコードチェンジのしやすさを最優先したコードフォーム列をコード列演奏デ

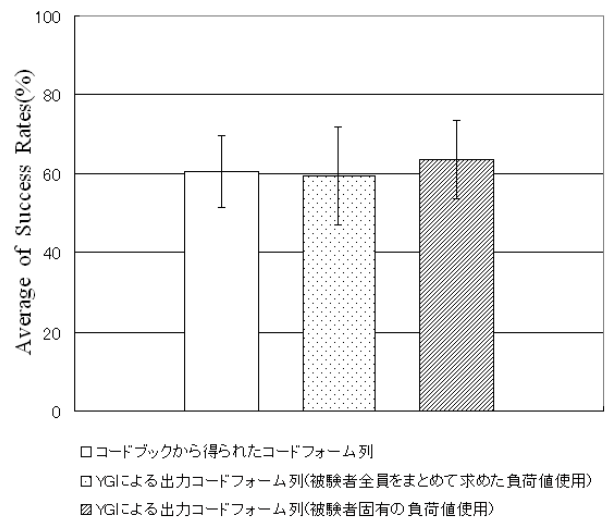


図 2 コードブック及び本システム'YG'のコードフォーム列における押弦成功率の比較

ータから最小二乗推定した負荷値を用いて求めるシステム'YG'を提案し、その有用性を検討した。その結果、評価実験では、あまり明確な差は出なかったが、今後負荷値を求めるための実験で演奏してもらったコードフォーム列のパターン数を増やすことで、より初心者に見合った負荷値が得られ、押弦しやすいコードフォーム列が出力されるであろうと期待される。

謝辞 本研究は本学フロンティア事業「知能情報科学とその応用」、文科省科研費(課題番号 16700154)、龍谷大学 HRC 第 2 プロジェクトの援助を受けた。

参考文献

- [1] ザ・ギターコードバイブル, ヤマハミュージックメディア(1988)
- [2] swingo 高橋他: これでOK! ギターコードの押さえ方, 音楽之友社(2001)
- [3] 塚本慶一郎他: ギター・コード・ブック, リットーミュージック(2000)
- [4] 比羽歩他: 絶対使うエレキ・コード・ナビ, 中央アート出版(1996)
- [5] こ〜どふ, フリーソフト, Bow
- [6] D'Accord Guitar Chord Dictionary, ギターコード習得ソフト, D'accord Music Software
- [7] 澤山・丸山・江村・三浦・柳田: 「ギターにおけるコード演奏の最適化に関する検討」, 音楽音響研究会資料 MA2004-11 (2004.06)