

2階マルコフ過程を用いた HMMによるコード付与手法の提案

森 篤史¹ 新井 イスマイル²

概要：近年、様々なソフトの流通に伴い一般の人が作曲することが増えている。しかし、初心者が一から作曲を行うのは困難である。そこで、初心者への作曲支援を目的とし、作曲した旋律に対して適切なコードを自動で付与する方法を検討する。既存の技術として、隠れマルコフモデル (HMM) を使用したものがある。コードの遷移を隠れ状態で表現し、旋律を出力状態としている。しかしこの方法では、1段階前のコードしか考慮していないため、特定の2種類のコードの繰り返しが多く見られた。そこで、我々は2階マルコフ過程を用い、2段階前のコードを考慮したHMMを作成する手法を提案する。その結果、特定の2種類のコードの繰り返しがなくなり、比較的バランスの良いコード進行になったことを確認した。また、音楽経験者21人による10段階の主観比較評価を行なった結果、本手法の平均は5.45、既存手法の平均は5.19と、本手法が既存研究の評価を上回った。

キーワード：音楽情報処理システム、計算機の介在した作曲・編曲、自動コード付与

1. はじめに

近年、PCを使って作曲を行うDTMソフトウェアや、歌詞と旋律を入力するだけで歌声を合成してくれるボーカロイドの流通によって一般の人が作曲することが増えている。しかし、楽曲の制作には音楽的知識・経験が不可欠であり、実際作曲を始めても挫折してしまうことが多々ある。初心者の作曲を支援するシステムが求められている。

作曲を行う場合、旋律を先に作成しそれに合うコードを後から付けていく方法と、コード進行を先に決定しそれに合う旋律を乗せる方法がある。音楽的知識・経験が少ない初心者はコードそのものに親しみがなく、比較的親しみのある旋律から先に作成する方が一般的であると考えられる。しかし、旋律を作成できた場合であってもそれに合うコードが見つけれなければ、伴奏付きの曲を完成させることができない。旋律に合うコード進行を見つけるためには知識・経験に加え音楽的センスも必要となり、初心者にとって容易なことではない。

本稿では、与えられた旋律に対して適切なコードを自動で付与することを目的とする。既存の技術として、隠れマルコフモデル (HMM) を使用したもの [1] がある。コードの遷移を隠れ状態で表現し、旋律を出力状態としている。

しかしこの方法では、特定の2種類のコードの繰り返し等、一般的に使われるコード進行と比べて単純な進行が多く見られた。これは、次のコードは一つ前のコードによってのみ定まるという単純マルコフ過程を隠れ状態に用いているためである。

そこで、我々は2階マルコフ過程を用い、2段階前のコードを考慮したHMMを作成する手法を提案する。これにより、より複雑で音楽的なコードが付与されることを期待する。

提案手法によるコード付与結果を確認すると、特定の2種類のコードの繰り返しがなくなり、比較的バランスの良いコード進行になった。また、音楽経験者21人に本手法、既存手法、原曲のコードを比較評価してもらった。全ての曲、コードに10段階で評価値をつけてもらった結果、それぞれの平均は本研究が5.45、既存研究が5.19、原曲が6.20であった。本手法は原曲に勝ることはできなかったが、既存研究の評価を上回った。

以降本稿では、2章にて既存研究を3つ挙げ、現状の問題点を述べる。その後、3章で提案手法の方針を示し、4章に実装方法を記す。5章では本手法、既存手法、原曲のコードを比較するための実験方法、6章でその実験結果を示し、それらを元に考察を行う。最後に7章で本研究について振り返り、今後の方針を表す。

¹ 明石工業高等専門学校 電気情報工学科

² 奈良先端科学技術大学院大学 総合情報基盤センター

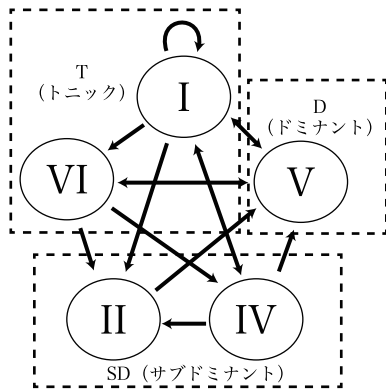


図 1 和声法のコード許容遷移図

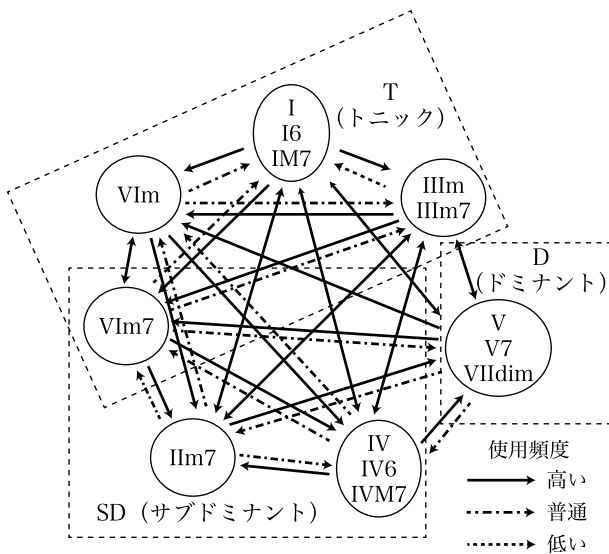


図 2 AMOR のコード許容遷移図

2. 既存研究

ここでは既存研究を3つ取り上げ、それぞれの貢献と問題点について述べる。

2.1 和声法に従う方法

三浦らは音楽専門教育による和声法 [2] をポピュラー音楽に拡張させた和声付与システム「AMOR」 [3] を実現した。和声法の入門段階ではコードの遷移が図 1 のように許可されているが、それをポピュラー音楽に合わせて図 2 のように拡張した遷移図を作成した。さらに経過音（異なる高さのコード構成音の間の音）、刺繍音（同じ高さのコード構成音の間の音）を除いた旋律とコード構成音を比較することで、適切なコードの付与を目指している。実際に三浦らの手法に従いコード付与を行った結果と理想のコード進行を、図 3 に示す。最後 2 つのコードを見ると、「ソ・シ・レ」の和音から 1 音追加されただけの「ミ・ソ・シ・レ」の和音に遷移しており、理想とするコード進行に比べて平坦な印象を受ける。その他の旋律に対しても、繰り返しのよってコードが変わったり、フレーズの終わりで終止しな

理想とするコード進行	IVM7	V7	IIIIm7	VIIm7
AMOR によるコード進行	IVM7	IIIIm7	V	IIIm7

図 3 AMOR によるコード進行

	C (I)		F (IV)		G (V)		C (I)	
	G (V)		C (I)		G (V)		C (I)	
	G (V)		C (I)		G (V)		C (I)	
	C (I)		F (IV)		G (V)		C (I)	

図 4 MySong によるコード進行

いなど、結果があまり音楽的でないことが多々あった。この手法では、音楽理論のみに従っているため、理論的には間違っていないけれども一般的には使われないコード進行が付与される可能性を多分に含んでいる。

2.2 データベースを使用する方式

東山は予め用意したコード進行パターンからコードを付与するシステム [4] を構築した。2 小節など決められた長さのコード進行をデータベースに登録しておき、旋律にコード構成音、アボイドノート（コード構成音に対して不協和音となる音）、アベイラブルノート（アボイドノート以外の音）がどれくらい含まれるかなどのマッチングスコアを計算することによってコード進行の絞り込みを行っている。データベースに音楽的なコード進行を登録することで、付与されるコードを音楽的にすることができる。その反面、旋律に対して適切なコードがデータベースに登録されていない場合、正しくコードが付与されない問題がある。入力される可能性がある旋律は膨大に存在するため、それらに適切なコード進行を全て用意するのは現実的ではない。また入力できる曲の長さが、用意したコード進行の長さ限定される問題もある。

2.3 隠れマルコフモデル (HMM) を利用する方式

Simon らは、既存楽曲を隠れマルコフモデル (HMM) で学習することによって、コード付与を行った。 [1] 「MySong」というシステムで、マイクに向かって旋律を歌うだけで自動でコードが付与され、簡単な伴奏付きの楽曲を作成してくれる。また、JAZZ ファクタと Happy ファクタの 2 つのパラメータによってコードを変更することができた。

コード付与を行うところで HMM を使用しており、コードの遷移を隠れ状態で表現し、旋律を出力状態としている。一般的に使われているコード全てを対象とするのは数も

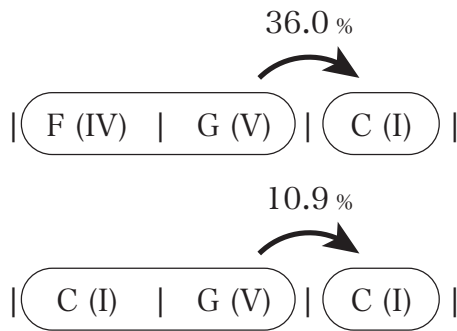


図 5 状況の異なる G (V) → C (I) の遷移確率

多く困難なため、考慮するコードを絞った。コード構成音が3音のコードは、ルート音がCの場合C, Cm, Cdim, Caug, Csus4の5つである。これをルート音12種類全てに適用し、合計60種類のコードが存在する。ほぼ全てのコードはこれら60種類の内のどれか一つに単純化することができる。これらのコード60種類に、曲の開始と終了の状態を加え、62個の隠れ状態を作成した。出力状態は、旋律を小節ごとに区切り、旋律に含まれる音の長さを12個のベクトルで表現した。

HMMを使用することによって簡潔に旋律とコードの関係性を表すことができ、前述の研究と比べ細かいパラメータ設定の必要がなくなった。さらに、既存楽曲から学習を行うことによって、常識から逸脱したようなコード進行が付与される可能性を一段と下げること成功した。

この手法をJ-POPに適用した場合のコード進行を図4に示す。図は楽譜と同様に左上から右下に向かってコードが進行し、縦線は小節の区切りを表している。MySongではこのような、G (V) と C (I) のコードの繰り返しが多く見られるなど、単調なコード進行が頻出した。また、メジャーコードのみで構成され、マイナーコード等が出現していないことも特徴としてあげられる。J-POP120曲を解析したことによって判明した、状況の異なるG (V) → C (I) の遷移確率を図5に示す。これはF (IV) → G (V) と遷移してきた場合と、C (I) → G (V) と遷移してきた場合において、次にC (I) が来る確率は大きく異なっていることを示している。にもかかわらずこのシステムでは、一つ前のコードからの遷移確率のみから判断し、それ以前のコード進行は加味していない。そのため、既存楽曲には無いような単調な繰り返しが発生すると考えられる。

3. 2階マルコフ過程を用いたコード付与手法の提案

既存研究3つについて長所と短所を表1にまとめる。総合的に考えて、現状2.3のMySongが最も好ましいと考えられる。従って本研究では、MySongをベースに改良を加えていく。

MySongでは、隠れ状態に単純マルコフ過程を使用して

表 1 既存研究の長所と短所

条件	AMOR	東山ら	MySong
付与されたコードの適切度	×	△	△
入力旋律の長さの自由度	○	×	○

表 2 N階マルコフ過程と隠れ状態数

階数	隠れ状態数
1	62
2	3,844
3	238,328
4	14,776,336

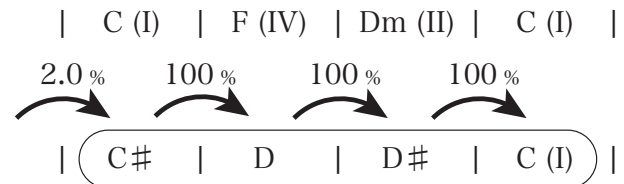


図 6 2階マルコフ過程による問題 特殊なコード進行の例

おり、次のコードは一つ前のコードによってのみ定まるという仮定に基づいていた。しかし、2.3でも述べたように、実際はそれ以前のコードもコードの選択に深く関わってくる。隠れ状態にN階マルコフ過程を用いることで、連続するN個のコードから次のコードが定まるというモデルを構築することができる。

3.1 N階マルコフ過程の検討

何階のマルコフ過程を用いるのが適切なかの検討を行う。そのためには、次のコードが定まるのに、いくつ前のコードが影響を及ぼしているのか調査する必要がある。旋律はフレーズと言われる4小節から8小節ほどの区間で区切られる。コード進行もこのフレーズに従い終止に向かうものがほとんどで、この8小節を超えて互いのコードが影響していることは考えにくい。1小節にはだいたい1つか2つのコードを含むため、最大16個前のコードが関係している可能性がある。

しかし16階マルコフ過程の実装は、計算量及び必要学習曲数が膨大となるため現実的ではない。c種類のコードを対象とした場合、曲の開始と終了の2つの状態を含めてc+2の状態が存在することになる。ここから、c種類のコードを対象とした場合のN階マルコフ過程に必要な状態数xは、

$$x = (c + 2)^N \quad (1)$$

で表される。これに基づき、MySongで使用されていた60コードを対象とした場合、各階数による隠れ状態数がどれくらいの値になるのか表2に示す。また、学習曲内である程度は各状態が出現する必要があるため、必要な学習曲数も隠れ状態数に従って増えていく。今回はN階マルコフ過程を用いた際に、単純マルコフ過程よりも良くなるかど

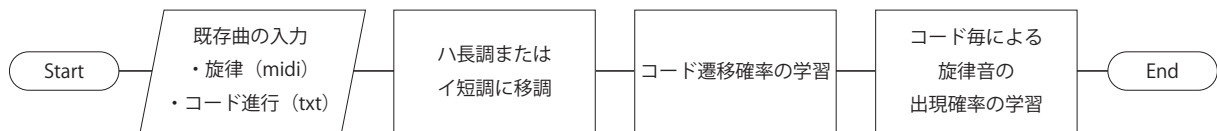


図 7 既存曲の学習の流れ

うかを検証することを主目的とするため、要求される計算量、学習曲数が最も少ない 2 階マルコフ過程を用いることにした。

3.2 2 階マルコフ過程を用いた時の問題点とその改善策

隠れ状態に 2 階マルコフ過程を用いる以外は基本的に MySong の手法に従っていく。しかし、そのままだと問題が発生した部分があるので、その原因と改善策について述べる。

MySong の手法を 2 階マルコフ過程に置き換え、新たな旋律に対しコード付与を行うと、図 6 のような学習曲内で稀にしか出てこないような特殊なコード進行が頻出した。このようなコード進行は、その進行の先頭に遷移する確率は極めて低いが、その進行途中の遷移確率は非常に高いという特徴がある。この特徴が原因で、総合的な評価値が高くなってしまい、このコード進行が選ばれたと推測できる。

そこで、学習曲内で一定回数しか出てこないような隠れ状態については無視することにし、出現回数を 0 に直した。具体的には、試行錯誤の結果 10 回未満のものを排除すると良いことが判明した。これにより、学習曲で稀にしか出てこないようなコード進行が頻出することはなくなった。

4. 提案手法の実装

以下に、既存曲の学習と新しい旋律に対するコード付与の 2 つの行程にわけ、提案手法の実装を説明する。

4.1 既存曲の学習

コードと旋律の関係をモデル化した HMM を作成するため、既存曲から学習を行う。学習の全体の流れは図 7 に示す。以下にその学習の手順の詳細を示す。

4.1.1 学習曲データの用意

既存曲を複数曲用意し、データ化を行う。旋律の情報を MIDI データに保存する。記録するのは主旋律の情報のみとし、同一時間には単音しか鳴らない。また、小節とコードの情報、及び調性の情報は MIDI には記録できないため、テキストデータを使い保存する。小節の情報が記載されているため、MIDI データとテキストデータはリンクできる。

既存曲は様々な調性で書かれているが、楽曲内全ての音高を変更し移調を行っても、雰囲気は大幅に損ねることはない。また、コード進行は調性を統一することで同じように扱うことができる。よってすべての楽曲は調性の情報に従い、調号が一つもない最も単純なハ長調もしくはイ短調

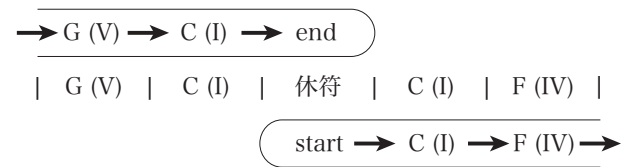


図 8 start コードと end コードの割り当て方

に移調する。

4.1.2 コード遷移確率の学習

次に、曲のコードの遷移確率を学習する。まず最初に、使用するコードについて定義を行う。

既存研究に従い、「曲の始まり」と「曲の終わり」を仮想のコードとし、start コード、end コードとして扱った。また、途中でコードが割り当てられていない休符は図 8 のように、前から見るときは end コードとして、後ろから見るときは start コードとする。

現在、J-POP で使われているコードは一般的に使われるものだけで 240 種類、分数コード等も含めると 1000 種類を超える。これらを全て独立した状態として学習を行うと、出現確率が非常に低いコードが多数出現し、計算資源及び学習用のデータが無駄になってしまう。計算量を抑えるため、また学習用のデータを有効的に使用するため、コードの分類を行う必要がある。MySong を踏襲して、コード構成音が 3 つのみの 60 種類のコードを対象とする。これに先程の start コードと end コードを加え、62 個の状態を作成する。

全ての学習曲に対して、コードを先程述べた 62 種類に単純化する。2 階マルコフ過程を用いるため、現在のコードとその一つ前コードをひとまとめとして一つの状態とし、表 3 のようなコード遷移表を埋めていく。この際、3.2 で述べた通り、全ての曲の中で 10 回未満しか出てこない状態については無視するため、遷移確率を 0 に直す。最終的に、コード遷移確率は $62^2 \times 62^2$ すなわち、 $3,844 \times 3,844$ の行列で表現できるようになる。

4.1.3 コードごとによる旋律音の出現確率の学習

各コードにおいて、どのような音が旋律に使われるのかを学習する。すべての学習曲を通して、各コードが鳴っている間に使用されている旋律の各音符の長さを合計した。ここで、オクターブが違う音に関しては同じ音として扱う。

その後、各コードに対して旋律の各音符の合計が長さ 1 になるよう正規化を行う。60 コードにそれぞれ 12 音のベクトルがあるので、コードごとによる旋律音の出現確率は 60×12 の行列で表現できる。例えば、コード C に対する

表 3 2 階マルコフ過程を用いたコード遷移表

		次の状態 (現在のコード, 次のコード)							
		(start, start)	(start, C)	(start, Cm)	...	(start, end)	(C, start)	(C, C)	...
		(start, start)	(start, C)	(start, Cm)	...	(start, end)	(C, start)	(C, C)	...
現在の状態 (一つ前のコード, 現在のコード)	(start, start)	0	0.4464	0	...	0	0	0	...
	(start, C)	0	0	0	...	0	0	0.3400	...
	(start, Cm)	0	0	0	...	0	0	0	...
	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
	(start, end)	0	0	0	...	0	0	0	...
	(C, start)	0	0	0	...	0	0	0	...
	(C, C)	0	0	0	...	0	0	0.2874	...
	(C, Cm)	0	0	0	...	0	0	0	...
	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
	(C, end)	0	0	0	...	0	0	0	...
	(Cm, start)	0	0	0	...	0	0	0	...
	(Cm, C)	0	0	0	...	0	0	0	...
	(Cm, Cm)	0	0	0	...	0	0	0	...
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	
(end, end)	0	0	0	...	0	0	0	...	

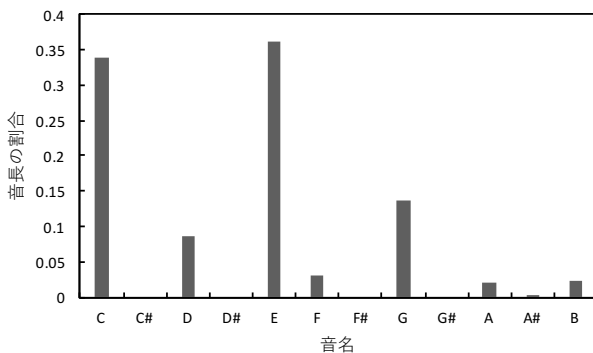


図 9 コード C に対する旋律音長の割合

12 音のベクトルは図 9 のようになる。

4.2 新しい旋律に対してのコード付与

学習が完了すると、HMM を利用し、新しい旋律に対してコード付与を行う。その全体の流れを図 10 に示す。以下にその手順の詳細を示す。

4.2.1 データの入力

入力は MIDI データで行う。また、入力する旋律はハ長調もしくはイ短調のみとする。HMM では、一定区間ごとに状態が遷移する。そのため、1 小節に 1 つずつコードが付与されることになる。入力データもそれを考慮する必要がある。

4.2.2 旋律とコードのマッチング度の計算

入力された旋律の各小節について、4.1.3 と同様に 12 音それぞれの長さを合計する。それを 4.1.3 で作成したコードごとによる旋律音の出現確率の行列と内積を取る。これにより、入力された旋律の各小節と 60 種類のコードのマッチング度合いを数値化することができる。

4.2.3 HMM での最尤状態の選択

今まで作成した情報を元に、2 階マルコフ過程を用いた

HMM を作成する。4 小節の旋律を入力した時の HMM の例を図 11 に示す。

現在のコードと、一つ前のコードの組み合わせを隠れ状態とし、4.1.2 で作成したコード遷移確率を隠れ状態の遷移確率とする。また、入力された旋律に仮想の start 小節と end 小節を最初と最後に追加し、各小節を出力とする。4.2.2 で作成した、コードのマッチング度合いを出力確率とする。start 小節は start コードからの出力確率が 1 であり、その他のコードは 0 である。同様に、end 小節は end コードからのみ出力がある。以上で HMM の必要なパラメータが設定された。

ここで、入力された旋律の順に出力が現れた場合の、隠れ状態について推測を行う。ビタビアルゴリズムによって最尤状態の選択を行い、その結果を出力する。その結果が付与されたコード進行であり、テキスト情報で確認することができる。

5. 評価実験

今回提案した手法の有用性や課題を確認するために評価実験を行う。J-POP 曲に対し、本研究の手法で付与されたコード、既存研究の手法で付与されたコード、元々ついていた原曲のコードについて比較・評価を行った。

5.1 既存研究の実装

本研究との比較を行うため、MySong の実装を行ったが、条件を合わせるために一部修正を行った。

まず、既存研究では入力を音響信号としていたが、本研究と同じく MIDI データで行う。また、JAZZ ファクタと Happy ファクタの 2 つのパラメータが存在していたが、全く人の手を加えない状態で比較検証したいため、この機能は省いた。また、付与されたコードをもとに簡易的な伴奏を作成する機能もあったが、本手法と同様にコード進行を

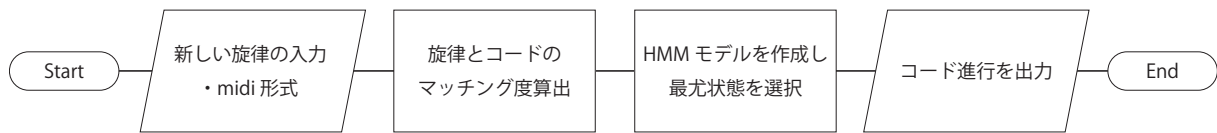


図 10 新しい旋律に対するコード付与の流れ

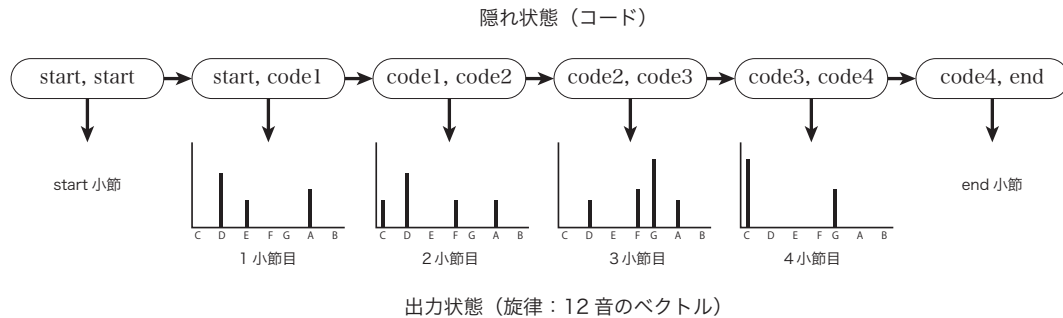


図 11 提案手法の HMM

テキスト出力するように変更した。

5.2 楽曲の準備

J-POP ヒット曲をまとめた音楽書 [5] に掲載されている曲の中で、様々なアーティスト、年代の J-POP10 曲を選ぶ。楽曲の選択には被験者が知っている曲、もしくは知らない曲ばかりにならないよう予め被験者に確認しておく。今回は表 4 に示す 10 曲を選択した。

それぞれの曲に対し、サビを中心に 16~32 小節程度のフレーズで切り取り、MIDI データとする。それを提案手法、先程修正した既存手法に適用し、コードを付与し、それぞれ提案手法パターン、既存手法パターンとする。[5] から 120 曲 (評価用楽曲は含まない) を選択し、学習曲とする。また、学習曲は最初から最後まで 1 曲全てを使用する。コードを付与する際は、手動でハ長調もしくはイ短調に直し、付与した後、元の調に手動で戻す。また、もともとついていたコードを、提案手法で付与される可能性のあるコード 60 種類に単純化し、原曲パターンとする。

聴き比べを行うため、旋律と伴奏の簡単な MIDI データを作成する。旋律はそのまま、伴奏はコードのみに集中してもらいたいため、コード構成音 3 つとオクターブ下のルート音を全音符で伸ばした。音色はピアノとし、音響ファイルに書き出す。

5.3 被験者実験

付与されたコードが適切かどうか判断できなければならぬため、音楽的知識がある人を被験者とする。今回は吹奏楽部に所属しており、楽器の経験がある学生 20 名と、音楽の教員 1 名に依頼した。

被験者には本研究パターン、既存研究パターン、原曲パターンを聴き比べてもらった。各パターンはランダムに再生しそれぞれのパターンなのかわからないようにする。

被験者はそれぞれのパターンを何回でも聞くことができるとする。

被験者は楽譜を見ながら楽曲を聴くことができるようにし、気になる点があれば楽譜上にコメント等を書き込んでもらう。さらに、主観的な品質、妥当性を元にそれぞれのパターンについて 1~10 の 10 段階で (数が大きいほど高評価) 評価してもらう。これらのポイントは各パターンで重なっても良いとした。また、その曲を聞いたことがあるかどうかアンケートを行う。

以上を 1 人 10 曲分繰り返す。

6. 実験結果・考察

まず、本手法によってどのようなコードが付与されるようになったか考察を行う。その後、音楽経験者 21 人に本手法、既存手法、原曲のコードを比較評価してもらった結果を記す。3 パターンの比較の他、曲ごとの考察や得られたコメントについての考察も行う。

6.1 本手法により付与されたコード進行とその考察

「トリセツ」の本研究パターンを図 12 に示す。同曲の既存研究パターンは図 4 に示した。既存手法の 3~12 小節目で現れている単調な繰り返しが、本手法ではなくなっていることを確認できる。また、マイナーコード等コードの種類も増え、全体的にバランスの良いコード進行になっていることが推測できる。

6.2 3 パターンの比較

本研究パターン、既存研究パターン、原曲パターンのそれぞれについて、10 曲 21 人分の合計 210 データが得られた。それぞれの評価の平均 (×印) と箱ひげ図を図 13 に示す。平均値は本研究が 5.45、既存研究が 5.19、原曲が 6.20 であった。また、それぞれのパターンの組み合わせにおけ

表 4 評価用楽曲一覧

曲名	アーティスト名	作詞者	作曲者
トリセツ	西野カナ	Kana Nishino	DJ Mass(VIVID Neon*) / Shoko Mochiyama / etsuco
終わらない歌	ゆず	北川悠仁 / 前山田健一	同左
華麗なる逆襲	SMAP	椎名林檎	同左
崖の上のポニョ	藤岡藤巻と大橋のぞみ	近藤勝也 / 宮崎駿	久石譲
SANDY	SADS	清春	同左
あの紙ヒコーキ くもり空わって	19	326	19
残酷な天使のテーゼ	高橋洋子	及川眠子	砂糖英敏
何も言えなくて…夏	J-WALK	知久光康	中村耕一
もう一つの土曜日	浜田省吾	同左	同左
今日までそして明日から	吉田拓郎	同左	同左

Am (VI)	F (IV)	G (V)	Em(III)
Am (VI)	F (IV)	G (V)	Em (III)
Am (VI)	Dm (II)	G (V)	C (I)
Am (VI)	F (IV)	G (V)	C (I)

図 12 「トリセツ」本研究パターン

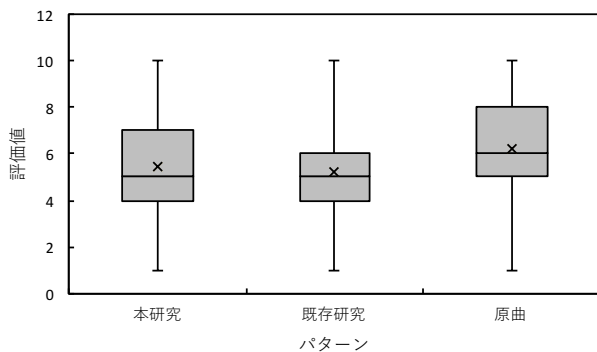


図 13 各パターンの評価平均と箱ひげ図

表 5 各パターンの組み合わせにおける勝敗数

A	B	Aの勝ち	Bの勝ち	引き分け
本研究	既存研究	103	78	29
本研究	原曲	61	121	28
既存研究	原曲	62	121	27

る勝敗数を表 5 に示す。以上 2 つから、本研究は原曲パターンには勝らないものの、既存研究パターンには勝っていることがわかる。

さらに詳しく本研究パターンと既存研究パターンの比較を行う。まず、帰無仮説「2 つの評価結果の分散には差が存在しない」のもと有意水準 5% で F-検定を行ったところ、 $p = 0.0069$ で有意差がみられた。よって、2 つの標本の分散が等しくないと仮定して t-検定を行う。帰無仮説「2 つの評価結果の平均には差が存在しない」のもと有意水準

表 6 パターン別の曲ごと評価平均

曲名	本研究	既存研	原曲
トリセツ	4.14	4.57	6.76
終わらない歌	6.10	5.90	5.43
華麗なる逆襲	6.29	6.05	4.67
崖の上のポニョ	3.48	6.57	6.95
SANDY	5.33	5.33	5.43
あの紙ヒコーキ くもり空わって	5.19	4.24	6.90
残酷な天使のテーゼ	4.81	4.43	7.52
何も言えなくて…夏	6.05	5.71	5.29
もう一つの土曜日	6.48	4.33	7.10
今日までそして明日から	6.62	4.71	6.00

5% で t-検定を行ったところ、 $p = 0.0747$ で有意傾向がみられた。ここからも、本研究は既存研究に若干勝っていることがわかる。

6.3 曲ごとの考察

10 曲について 3 パターンごとの評価値の平均を表 6 に示す。本研究は既存研究と比べて、10 曲中 7 曲で勝っており、2 曲で負けていることがわかる。全体的に見れば本研究は既存研究より勝っているが、全ての曲で高い評価が得られているわけではない。その原因を考察する。

まず、本研究で一番評価が低かった「崖の上のポニョ」のコードを図 14 に示す。前半は不自然なコードは付いていないが、最後 3 小節間 C のコードが続いており、これが評価を下げた原因となっている。解析曲が 120 曲と少ないため、仮想の end コードに向かう可能性のあるコードは数が少ない。一方、2 階マルコフを用いているため、最後の小節の後に end コードへ遷移するためには、2 つ前のコードから影響している。そのため、最後の 3 小節に違和感を持つコード進行が付与されたと考えられる。2 階マルコフを使ったことで隠れ状態が増えたため、学習曲が足りなくなったことが一番の原因である。

また、本研究パターンは原曲パターンに 10 曲中 3 曲で勝った。プロの作曲・編曲者に劣らないコードが付与できている曲もあることがわかる。一番差が開いた「華麗なる

C (I)	G (V)	Am (VI)	Em(III)
F (IV)	Em (III)	Dm (II)	G (V)
C (I)	G (V)	Am (VI)	F (VI)
G (V)	C (I)	C (I)	C (I)

図 14 崖の上のポニョ 本研究パターン

Am (VI)	F (IV)	G (V)	Em(III)
Am (VI)	Dm (II)	G (V)	C (I)
Am (VI)	F (IV)	G (V)	Em (III)
Am (IV)	Dm (II)	G (V)	C (I)

図 15 「残酷な天使のテーゼ」 本研究パターン

逆襲」は椎名林檎が作詞作曲を行っており、原曲コードにはかなり特殊なコードが付与されていた。一方、本手法は J-POP の一般的なコード進行を付与したため、好みが別れたと考える。

逆に、原曲パターンに大差で負けた本研究パターンも多く、未だプロの作曲・編曲者が付与したコードと比べて不自然なものが目立つことがわかる。例えば、図 15 に示す「残酷な天使のテーゼ」では、I 短調にも関わらず本研究パターンは I のコードで終わっている。また、全体的にも比較的明るい印象を受ける。これは、入力された旋律が長調なのか短調なのかを区別していないからである。さらなる改善が求められる。

6.4 得られたコメントとそれに対する考察

音楽の教員より得たコメントの一部を以下に示す。

- (1) 既存研究パターンは間違っていないがつまらないものが多かった。
- (2) 知ってる曲では原曲パターンを探してしまい、その評価を高くしてしまう。
- (3) 原曲パターンにはその曲の印象を深めるような特徴的な進行が効果的に使われている場合がある。しかし、機械が付与したものにはそれが無い。
- (4) 分数コードが全くなく、セブンスや add9 等のコードも存在しないので、不自然な感じが拭えない。特に、ベースラインは全体を大きく支配するので、分数コードの考慮は必須だろう。
- (5) センチメンタルならマイナーコードやマイナーセブンス等、歌の歌詞も和音に影響を及ぼしているはずなの

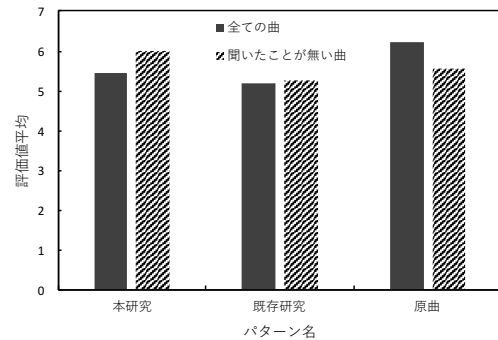


図 16 全ての曲と聞いたことが無い曲の評価平均

で、旋律だけだと情報が足りない気がする。

既存手法では単調な繰り返し等が多くなってしまったため、(1) のような感想が得られたと考えられる。本手法に対してそのようなコメントはないので、既存手法の問題が解決したと言える。

他の被験者からも (2) と同様の感想が得られた。そこで、聞いたことが無い曲のみで本研究パターン、既存研究パターン、原曲パターンそれぞれの平均値を、全ての曲を対象とした場合と比較して図 16 に示す。聞いたことが無い曲になると、原曲パターンの評価値が下がることがわかる。知ってる曲の知ってるコード進行であれば耳馴染みが良く、その分評価が高くなったのだと考えられる。また、本研究パターンの評価値が上昇し、原曲パターンを上回っている。これは「崖の上のポニョ」や「残酷な天使のテーゼ」といった誰もが知っているが、提案手法のコード付与性能が良くなかった楽曲が対象から外れたためである。しかし、全ての曲を対象とした本手法パターンの平均評価値と、聞いたことが無い曲のみを対象とした原曲パターンの平均評価値を比較しても、かなり近くなっていることがわかる。この傾向から本手法は、原曲のコードを再現することはできないが、付与されるコードのクオリティは平均的に見れば原曲に大きく劣らないことがわかる。本来このシステムは、ユーザが作曲した旋律に対してコード付与を行うことを目的としており、この目的を果たすには良い傾向であると言える。

様々な楽曲を一様に学習しているため、ありきたりなコード進行が付与されることになり (3) のような意見が出てきたと考えられる。しかし、インパクトがあるようなコード進行をするためには、人の手でも熟練が必要である。本手法のパターンであっても、曲によっては未だ顕著におかしい小節が残されているため、まだこの問題に取り組むのは早いと考える。どのような旋律に対しても適切なコード付与ができるようになった後、この問題に取り組む。

(4) に関しては、原曲パターンも分数コード、セブンスや add9 等を省いているため同条件である。しかし、ベー

ス音によってコードの響きが急に薄くなることもあり、そこが気になると回答した人も多かった。評価値にはコード構成音の組み合わせ以外にも、ベース音による響きの違い等が関係していると推測できる。特に、原曲パターンの評価が低いものは、分数コード、セブンスや add9 を含まないことによる違和感も大きいと感じた。使用するコードの再検討が必要である。

また、(5) の歌詞について今まで一切考慮していなかったが、これについても検討する余地があると思う。

7. おわりに

本報告では、2階マルコフを用いた旋律に対するコード付与について提案を行った。本手法を用いることで、既存手法で頻出した2つのコードの繰り返し等の単純な遷移が減り、より複雑なコード進行が付与されるようになった。音楽経験者の主観による10段階評価の平均を既存研究・原曲パターンと比較した結果、本手法が5.45、既存手法が5.19となり、本手法が既存手法の評価平均を上回った。また、全ての曲を対象とした場合と、知らない曲のみを対象とした場合を比較することで、本手法は原曲のコードを再現することはできないが、付与されるコードのクオリティは平均的に見れば原曲に大きく劣らないことがわかった。本来このシステムは、ユーザが作曲した旋律に対してコード付与を行うことを目的としており、この目的を果たすには良い傾向であると言える。

一方で、未だ原曲コードに大きく評価を離された曲も多くあった。学習曲が足りてない問題を始め、長調・短調の認識を行っていないことに原因があると考えられる。また、J-POPには分数コードやセブンスなどのコードも多用されており、利用するコードを再検討する必要があると感じた。今回はHMMを使ったコード付与に注目したため、調の推定等、実装を省いた部分がある。実際に初心者がコード付与システムを利用する場合には、自動での調の推定、コードアレンジのサポート等の機能が必要になってくる。今後はこれらのことについても検討を重ねていきたい。

さらに、歌詞を考慮したコード付与や、一般的なコードを並べるだけでなく特徴的なコードを効果的に使う必要性等の意見を得た。また、小節ごとにコードを付与しているが、実際には小節内で変化することも多い。すぐに対応するのは困難な問題だが、将来的にはこれらの問題にも対処していきたい。

参考文献

[1] Simon, I., Morris, D. and Basu, S.: MySong: Automatic Accompaniment Generation for Vocal Melodies, *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '08, New York, NY, USA, ACM, pp. 725–734 (2008).

- [2] 島岡譲, 池内友次郎, 他: 和声: 理論と実習, pp. 138–152, 音楽之友社 (1964).
- [3] 三浦雅展, 青山容子, 谷口光, 青井昭博, 尾花充, 柳田益造: ポップス系の旋律に対する和声付与システム: AMOR, *情報処理学会論文誌*, Vol. 46, No. 5, pp. 1176–1187 (2005).
- [4] 東山恵祐: 作曲支援システムにおけるコード進行及びキーの決定方法, *電子情報通信学会技術研究報告*. SP, 音声, Vol. 114, No. 52, pp. 261–266 (2014).
- [5] オールヒット曲, 自由国民社 (2015).