

修 士 論 文 の 和 文 要 旨

研究科・専攻	大学院 情報理工学研究科 情報・通信工学専攻 博士前期課程		
氏 名	篠原 透	学籍番号	1331052
論 文 題 目	シーケンシャルパターンマイニング拡張による 特徴的なコード進行の抽出手法		
<p>要 旨</p> <p>近年、楽器演奏技術を持たないユーザーでも作曲を楽しむことができる作曲環境が注目されている。しかし、初心者が音楽理論を習得し、適切なコード進行を構成することは容易ではない。また、経験者にとっても、終止形などの特定の進行や楽曲の展開を重視したり、一定の作曲者やジャンル等を意識して楽曲を構成したりすることが考えられる。</p> <p>したがって、実際の楽曲に使われている特徴的なコード進行を直感的かつ適切に構成できる作曲支援環境が必要であると考え。特徴的なコード進行とは、単によく使われているというだけでなく、音楽理論上重要視されるコード進行や、楽曲情報（作曲者やジャンルなど）・楽曲構造（小節線や楽曲の開始・終了など）と関連して用いられるコード進行である。そのようなコード進行を積極的に作曲に用いることで、和声を構成する負担が軽減され、より容易な楽曲制作が可能になると筆者は考える。</p> <p>そこで本研究では、実際の楽曲のコード進行から頻出パターンマイニングの一種であるシーケンシャルパターンマイニングを拡張することで、特徴的なコード進行を定量的に発見・顕在化する手法を提案した。具体的には、アイテムへの親子関係を導入することで転回形を含むコード進行を抽出しやすくし、作曲者や小節線などの疑似アイテムを付加してマイニング制御を行うことで楽曲情報や楽曲構造を考慮したパターンを抽出できるよう拡張した。</p> <p>本提案手法を楽曲 150 曲のコード進行データに適用した実験を行ったところ、親子関係の導入によって、support 値が少なく抽出しにくいパターンや既存手法では発見できないパターンが高い最小 support 値で提示できるようになった。また、疑似アイテムとして楽曲のメタ情報や小節線、楽曲の開始・終了を表すアイテムを付加することにより、作曲者・ジャンルごとのコードの使用傾向を明らかにし、出現位置やコードの音価の特徴を抽出パターンとして反映することができた。</p>			

平成26年度修士論文

シーケンシャルパターンマイニング 拡張による 特徴的なコード進行の抽出手法

電気通信大学大学院 情報理工学研究科
情報・通信工学専攻 コンピュータ学講座

学籍番号 : 1331052

氏名 : 篠原 透

指導教員 : 沼尾 雅之 教授

副指導教員 : 小林 聡 教授

提出日 : 平成27年 1月 30日

概要

近年、楽器演奏技術を持たないユーザーでも作曲を楽しむことができる作曲環境が注目されている。しかし、初心者が音楽理論を習得し、適切なコード進行を構成することは容易ではない。また、経験者にとっても、終止形などの特定の進行や楽曲の展開を重視したり、一定の作曲者やジャンル等を意識して楽曲を構成したりすることが考えられる。

したがって、実際の楽曲に使われている特徴的なコード進行を直感的かつ適切に構成できる作曲支援環境が必要であると考え。特徴的なコード進行とは、単によく使われているというだけでなく、音楽理論上重要視されるコード進行や、楽曲情報（作曲者やジャンルなど）・楽曲構造（小節線や楽曲の開始・終了など）と関連して用いられるコード進行である。そのようなコード進行を積極的に作曲に用いることで、和声を構成する負担が軽減され、より容易な楽曲制作が可能になると筆者は考える。

そこで本研究では、実際の楽曲のコード進行から頻出パターンマイニングの一種であるシーケンシャルパターンマイニングを拡張することで、特徴的なコード進行を定量的に発見・顕在化する手法を提案した。具体的には、アイテムへの親子関係を導入することで転回形を含むコード進行を抽出しやすくし、作曲者や小節線などの疑似アイテムを付加してマイニング制御を行うことで楽曲情報や楽曲構造を考慮したパターンを抽出できるよう拡張した。

本提案手法を楽曲 150 曲のコード進行データに適用した実験を行ったところ、親子関係の導入によって、support 値が少なく抽出しにくいパターンや既存手法では発見できないパターンが高い最小 support 値で提示できるようになった。また、疑似アイテムとして楽曲のメタ情報や小節線、楽曲の開始・終了を表すアイテムを付加することにより、作曲者・ジャンルごとのコードの使用傾向を明らかにし、出現位置やコードの音価の特徴を抽出パターンとして反映することができた。

目次

第1章	はじめに	1
1.1	背景	1
1.2	目的	2
1.3	論文の構成	2
第2章	関連研究	4
2.1	作曲支援・自動作曲手法に関する研究	4
2.2	音楽的情報に対するマイニングに関する研究	5
第3章	音楽理論	7
3.1	音楽の三要素	7
3.2	調	7
3.3	和音	8
3.3.1	三和音	8
3.3.2	四和音	9
3.3.3	五和音	9
3.4	転回形	9
3.5	コード進行	10
3.6	終止形	12
第4章	シーケンシャルパターンマイニング	14
4.1	シーケンシャルパターンマイニングの定義	14
4.2	アルゴリズム	15
第5章	提案手法	17
5.1	楽曲のコードシーケンス表現	18
5.1.1	基本コードアイテム	18
5.1.2	疑似アイテム	19
5.2	ナイーブなアルゴリズムの拡張方法	20
5.2.1	終止形パターン発見のための逆順走査手法	21

5.2.2	各種アイテムとギャップの取り扱い	22
5.2.3	基本コードアイテムへの親子関係の導入	24
5.2.4	コード進行の繰り返しパターンの抑制	26
第 6 章	作曲支援システム	27
6.1	提案システムの設計	27
6.2	提案システムの実装	29
6.2.1	入力	29
6.2.2	コード認識部	31
6.2.3	マイニング部	32
6.2.4	MIDI データ出力部	32
6.2.5	出力	32
第 7 章	評価実験	33
7.1	実験の方針	33
7.1.1	実験の目的	33
7.1.2	実験手法	33
7.2	アイテムへの親子関係の導入と出力パターンの比較	36
7.3	メタ情報アイテムを含むパターンの抽出実験	39
7.4	小節線アイテムを含むパターンの抽出実験	45
7.5	構造アイテムを含むパターンの抽出実験	50
第 8 章	おわりに	52
8.1	まとめ	52
8.2	今後の課題	52

表 目 次

5.1	本提案手法で導入するアイテムの種類	18
5.2	アイテムの種類による数え上げの有無と後続のギャップの走査範囲 .	22
5.3	親子関係導入後の数え上げ方の例	26
7.1	⟨I⟩ に対する後続のコード進行の有意差と confidence 値 (作曲家アイテムを含む場合)	40
7.2	⟨I⟩ に対する後続のコード進行の有意差と confidence 値 (ジャンルアイテムを含む場合)	41
7.3	⟨I⟩ に終止するコード進行の有意差と confidence 値 (作曲家アイテムを含む場合)	42
7.4	⟨I⟩ に終止するコード進行の有意差と confidence 値 (ジャンルアイテムを含む場合)	43
7.5	confidence 値が 1.00 となる作曲家アイテムを含むコード進行とその有意差	44
7.6	最小 support 値 10 を満たす全終止 ⟨V', I'⟩ のパターンと support 値 .	47
7.7	最小 support 値 10 を満たす全終止 ⟨V', I _m '⟩ のパターンと support 値	48
7.8	最小 support 値 10 を満たす偽終止 ⟨V', VI _m '⟩ のパターンと support 値	48
7.9	最小 support 値 10 を満たす偽終止 ⟨V', III _m '⟩, ⟨V', II _m '⟩ のパターンと support 値	48
7.10	最小 support 値 10 を満たす変終止のパターンと support 値	49
7.11	楽曲の開始を表す構造アイテムを含むパターンと support 値	50
7.12	楽曲の終了を表す構造アイテムを含むパターンと support 値	50

目次

3.1	長調 (長音階)	8
3.2	短調 (自然的短音階)	8
3.3	三和音の種類	9
3.4	四和音の種類	9
3.5	五和音の種類	10
3.6	分数コード表記	10
3.7	C major におけるダイアトニックコード	11
3.8	カデンツにおけるダイアトニックコードの進行	11
3.9	全終止 (不完全終止と完全終止)	12
3.10	半終止	12
3.11	偽終止	13
3.12	変終止	13
3.13	女性終止	13
5.1	アイテムの親子関係の例	25
6.1	提案システムのフローチャート	29
6.2	提案システムの GUI 図	30
7.1	最小 support 値ごとの出力パターン数とその分布 (長さ 2 以上 100 以下)	37
7.2	最小 support 値ごとの出力基底パターン数とその分布 (長さ 2 以上 100 以下)	37
7.3	最小 support 値ごとの出力パターン数とその分布 (長さ 2 以上 4 以下)	38
7.4	最小 support 値ごとの出力基底パターン数とその分布 (長さ 2 以上 4 以下)	38

第 1 章 はじめに

1.1 背景

楽曲制作の環境として DeskTop Music (DTM) と呼ばれる作曲環境が注目されている。これは Digital Audio Workstation (DAW) と呼ばれる市販の楽曲制作ソフトウェアを使用することで、コンピュータ上で作曲にまつわる作業を完結させることができる方法であり、楽器演奏技術を持たないユーザーでも作曲を楽しむことができる。昨今ではコンピュータ自体の性能向上や、ソフトウェア・音源などの成熟により、誰でも気軽に始めることができるようになった。また、日本では VOCALOID[1] の登場を機に、自身で制作した楽曲を動画サイトや SNS 等で発表する文化が一般に広まった。

このように DTM を取り巻く環境は飛躍的に向上しているが、実際にどのような楽曲を制作するかは作曲者自身の経験と技術次第であり、一定の音楽的な知識を必要とする。現在最も大衆的に聴かれているポピュラー音楽は、おおむね音楽の三要素と呼ばれるメロディー・ハーモニー・リズムの 3 つによって構成されている。中でも、ハーモニーとはコード（和音）のことを意味し、一つ一つのコード進行を音楽理論に基づき構成することが求められる。一般に音楽理論に違反する進行は、収束感がなく落ち着かない進行となったり、違和感のある進行となってしまうものが多い。作曲を始めたばかりの初心者がコード理論を使いこなすまでの習得につまづくことは多く、楽曲制作を敷居の高いものにしてしている。

また、作曲経験者の場合でも、終止形などの特定の進行や楽曲の展開を重要視して作曲を行ったり、一定の作曲者やジャンル等を意識して楽曲を構成したりすることが考えられる。したがって、ポピュラー音楽に使われる特徴的なコード進行を直感的かつ適切に構成できる作曲支援環境が必要であると考えられる。

2012 年 12 月にドイツの Steinberg 社によって製品化された DAW である Cubase 7[2] では、コードトラック及びコードアシスタントと称して、楽曲のコード進行の一元管理やコード進行例の提示を可能にしたインターフェースが提供された。このインターフェースはユーザーへの負担を軽減する一つの解決策であり、コード進行の構成を補助する機能への需要は高まっているといえる。

1.2 目的

本研究の主たる目的は、実際の楽曲のコード進行から計算機科学的手法を用いて、特徴的なコード進行を発見・顕在化し、音楽的な特徴を定量的に導出することである。ここでの特徴的なコード進行とは、単純にポピュラー音楽によく使われているコード進行というだけではなく、音楽理論上重要視されるコード進行や、楽曲情報（作曲者やジャンルなど）・楽曲構造（小節線や楽曲の開始・終了など）と関連して用いられるコード進行である。そのようなコード進行を積極的に作曲に用いることで、和声を構成する負担が軽減され、より容易な楽曲制作が可能になると筆者は考える。

筆者はかつて、実際の楽曲のコード進行から抽出される特徴的なコード進行を作曲支援に有効に活用する手段として、シーケンシャルパターンマイニングを用いてコード進行を提示する作曲支援システムを提案した [3]。その結果、システムを利用することで、楽曲の制作時間の短縮に一定の効果が示された。しかし、そこで使用されていたマイニング手法は既存手法を適用したナイーブな手法であり、音楽的に有用と考えられるパターンが抽出されにくいなどの問題があった。

そこで、本論文ではナイーブなシーケンシャルパターンマイニングを拡張し、コードの進行の仕方に特化した探索や、楽曲情報・楽曲構造を条件としたコード進行の抽出を可能にすることにより、実際の楽曲に含まれる特徴的なコード進行を発見・顕在化し、音楽的な特徴を定量的に導出する手法を提案する。

1.3 論文の構成

本論文の構成は次の通りである。

第 1 章

本研究の背景について述べる。

第 2 章

本研究の関連研究について述べる。

第 3 章

本実験で使用した音楽理論について説明する。

第 4 章

本実験で使用したシーケンシャルパターンマイニングについて説明する。

第 5 章

提案手法として、シーケンシャルパターンマイニングの拡張手法について述べる。

第 6 章

提案手法を利用した作曲支援システムについて述べる。

第 7 章

提案手法に対する評価実験について述べる。

第 8 章

全体のまとめと今後の課題について述べる。

第 2 章 関連研究

本研究は作曲支援システムに関する研究分野と、音楽的情報に対するマイニングに関する研究分野にまたがっている。そこで本章では、関連した既存研究について分野別に述べる。

2.1 作曲支援・自動作曲手法に関する研究

計算機の介入した作曲には、自動で旋律や楽曲を生成するものや、最終的な決定をユーザーに委ねた作曲支援などがあり、様式は多岐にわたる。その手法は、決定論的手法と確率論的手法の 2 種類に大別される [4]。

決定論的手法は、古典的な音楽理論を用いた形式的な作曲手法であり、計算機が作曲支援に使用される前から音楽の旋律の生成に利用されてきた。非常に厳密な定義を持つため、計算機への実装は容易であり、音楽理論からの逸脱のない「整った」音楽の生成を可能とする。しかし、現代のポピュラー音楽で許容される禁則や作曲者の裁量による表現は再現されないといった問題がある。

一方、確率論的手法では、乱数などの不確定要素をパラメータとして用いることにより、一定の偶発性をもった楽曲を生成できる。決定論的手法に対して、確率論的手法を作曲に用いる利点は、作曲者自身も予想できなかった結果を得ることにある。しかし、従来の確率論的手法では、生成される音楽の芸術的な評価は難しい。そのため、計算機による出力を作曲者が評価選択し、得られた結果をさらに修正する必要があった。深山らは、歌詞の韻律に着目した自動作曲システム Orpheus[5] を Web 上で公開している。これは日本語の韻律を制約として、動的計画法を用いて旋律を組み立て、歌詞のある楽曲を生成するシステムである。これにより、歌詞が聞き取りやすく、自然な旋律をもつ楽曲の生成ができる。ただし、Orpheus では旋律の生成に最適化手法を用いているため、音楽理論の意図的な逸脱や、経過的な非和声音の使用が少なく、音楽性における不自然さが指摘されている。また、あらかじめ用意されたパターンを用いて確率遷移によってコード進行を生成しているため、生成される楽曲が似てしまうなどの問題がある。

徳丸らは、カオスニューラルネットワークを用いて、メロディーと伴奏による楽曲を、雰囲気徐徐に変化させながら奏で続けるシステムを提案した [6]。カオス

ニューラルネットワークでは、全くのランダムではないが、事実上予測不可能な値を用いている。これにより、楽曲に一種の偶奇性をもたらすことができる。またこの研究では、記憶するパターンの入れ替えによって、数小節の旋律が繰り返される音楽に比べて飽きない音楽が生成された。しかし、この研究では音楽的理論はごく簡単なものにとどまっており、生成される楽曲が芸術的な評価を得られるものとは言い難い。

そこで、決定論的手法と確率論的手法の双方の利点を用いることのできる手法として、対話型進化論的計算 (Interactive Evolutionary Computation, IEC) を作曲支援に用いるシステムが提案された。IEC では、初期の段階では既存の確率論的手法に基づきランダムに生成されるが、ユーザーとシステムの対話を通して徐々にユーザーが求める一点へ収束していき、ユーザーは出力結果として完全に満足できるものを得られるというものである。安藤らは、IEC を用いてメロディーを提示し選択させる作曲支援システムを提案した [7]。しかし、IEC を用いた作曲手法では、出力結果として得られる数十秒から数分あるメロディーを全て聞いて評価する必要があり、ユーザーに対する負担が大きいことが問題として知られている。

2.2 音楽的情報に対するマイニングに関する研究

音楽的情報の繰り返し部分や特徴的な部分の抽出に対するマイニング方法は、2種類のアプローチが考えられる [8]。1つは音楽音響信号を直接解析対象としてマイニングする手法である。もう1つは、音楽的情報をテキストなどの記号に変換したのに対し、テキストマイニングの手法を適用するものである。本研究では後者のマイニング方法を扱う。

藤川らは音楽音響信号に対して短時間のクロマベクトルを算出し、得られた音高を記号に変換したのに対するマイニングを行い、楽曲の特徴的な部分 (リフレイン構造) を抽出する手法を提案した [9]。また、Jiménez らは MusicXML からメロディーラインを記号列化したものをマイニングし、楽曲の主題 (motifs) を抽出した [10]。その他、Koh らはメロディーラインなどの音楽的情報をビット列に変換し、繰り返されるパターンの抽出を既存手法よりも高速に抽出できることを示した [11]。

コード進行に特化してマイニングを試みた研究例として、Ogihara [12] らによる作曲家ごとのコード進行の特徴抽出がある。この手法は、楽曲のコード進行について N-gram 頻度を計り、その作曲家を示す類似度を定義しクラスタリングを行うことで、音楽情報検索に利用しようとした。また、Pérez らも楽曲のコード進行についての N-gram 頻度から楽曲のジャンルを推定する手法を提案した [13]。N-gram

頻度はテキストマイニングのみならず，コード進行の分析に有効な手法の一つである [14]。しかし，これらが対象とするデータは専らコードのみであり，楽曲情報や楽曲構造のような付加的な要素を考慮してコード進行を抽出することはできない。

第3章 音楽理論

古典的な音楽理論は、現在のコード理論の基礎となっている。本章では、本研究で使用する最低限の音楽理論について紹介する。音楽理論の解釈は個人によって若干異なるが、本章で紹介しているものは北川 [15] による解釈を参考にしている。

3.1 音楽の三要素

一般的な楽曲は旋律（メロディー, melody）・和声（ハーモニー, harmony）・拍子（リズム, rhythm）の3つで構成されている。旋律はその楽曲を印象付ける特徴的な音の並びを構成する。和声は和音（コード, chord）を用いて特有の響きを楽曲に加えることにより、旋律を下支えする。拍子はその楽曲に一定の区切りを提供し、時系列的に連続する楽曲全体の構成単位となる。この3つをまとめて音楽の三要素という。実際の作曲過程では、どの要素からでも制作されるが、本研究では和声から制作する場合を扱う。

3.2 調

ある楽曲のメロディーや和音が一定の音を中心に構成されているとき、その音の構成組織を音階 (scale) と呼ぶ。音階は時代や地域によって異なるものが多数存在するが、特に現代のポピュラー音楽で重宝されているものが長音階 (major scale) と短音階 (minor scale) である。この2つを元に作られた楽曲には調性 (tonality) があるという。また、調性を作り出す構成音の組織を調 (key) と呼ぶ。調には大きく分けて長調 (major key) と短調 (minor key) の2種類がある。長調は長音階で構成されており、一般的には明るい雰囲気を持ち、陽気・活発さを励起するといわれている。短調は短音階で構成されており、一般的には暗い雰囲気を持ち、陰気・悲しさを励起するといわれている。なお、短音階には自然的短音階 (natural minor scale) ・和声的短音階 (harmonic minor scale) ・旋律的短音階 (melodic minor scale) の3種類の派生音階が存在する。調にはその基準となる音があり、これを主音 (ト

ニック, tonic) と呼ぶ。調は主音と長調・短調の記述を併記し, C major, B^bminor などと書き表す。

具体例として, 図 3.1 に C major, 図 3.2 に A minor の構成音を示す。ルートの

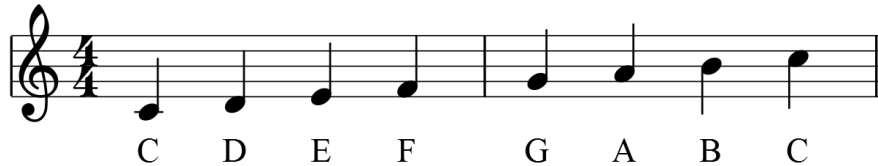


図 3.1: 長調 (長音階)

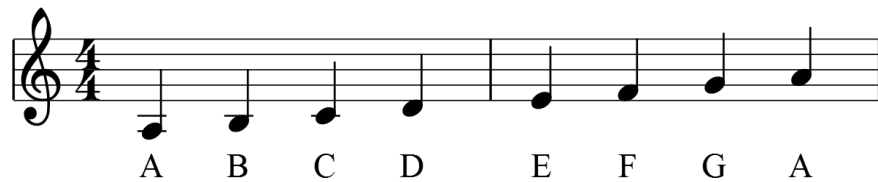


図 3.2: 短調 (自然的短音階)

音名は楽曲の調の主音からの距離に基づいて, I, II, III, IV, V, VI, VII とローマ数字によって表記することもある。このローマ数字を音度記号と呼ぶ。

3.3 和音

複数の音の積み重ね, またその響きを和音およびコード (chord) と呼ぶ。調と同じくコードにもその基準となる音があり, これを根音 (ルート, root) と呼ぶ。積み重ねた音に対して, 根音から音高の低い順に, 第 1 音・第 3 音・第 5 音とも呼ぶ。コードを記号化したものがコードネームであり, ルートの音名とコードの種類を並べて示す。なお, コードのルートの音名は音度記号を用いて表すこともあり, 本稿ではこれをコードの階名表記と呼称する。

3.3.1 三和音

3つの音で構成される和音が三和音 (triad) である。重ねる音の間隔を変えることで得られる, 図 3.3 に示す和音を本研究での対象とする。

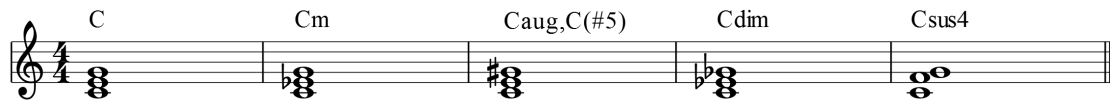


図 3.3: 三和音の種類

3.3.2 四和音

4つの音で構成される和音が四和音である。基本的に、三和音を基盤にして一音を付加する形で作られている。四和音では図3.4に示すものを扱う。

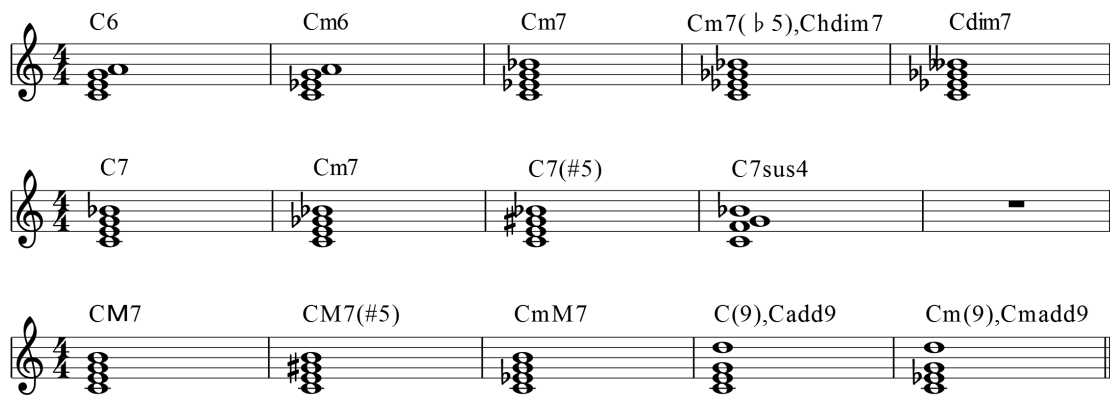


図 3.4: 四和音の種類

3.3.3 五和音

5つの音で構成される和音が五和音である。五和音となるものは、三和音・四和音となるものにあえて異質な響きをもたらす一音を加えることによって作られている。五和音では図3.5に示すものを扱う。

3.4 転回形

和音には、根音を最低音とせず構成音の配置を入れ替えたものが存在する。これを和音の転回形という。転回形には、根音を最低音とした基本形、第3音を最低音とした第一転回形、第5音を最低音とした第二転回形、第7音を最低音とした第三転回形がある（第三転回形は四和音のみに示される）。五和音は三和音・四和音をベースに1音を付加した和音であり、付加音は基本的に転回して使用しない。

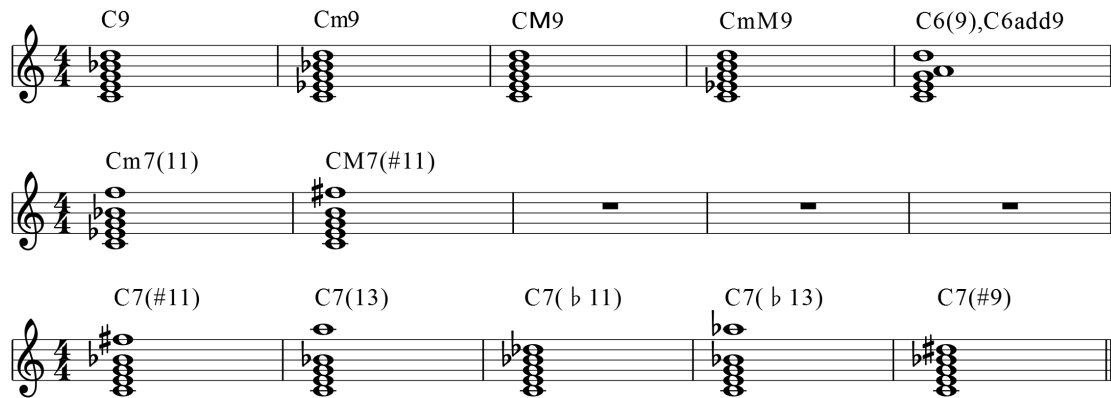


図 3.5: 五和音の種類

転回形のコードネームの表記法として分数コード表示がある (図 3.6)。これはコードネームを分子，最低音を分母とした分数方式による表示で，G/B，I/V などと表記する。ただし，I/II など，分母に和音の構成音が含まれないものも存在する。これは転回形ではなく，コードにベース音を付加することを意味するものである。実質的に四和音や五和音およびその転回形と同じ構成音を持つ場合もあり，I/II と Iadd9 はその例である。このような表記をあえて用いる理由として，コードを直感的に読みやすくしたり，コードの前後におけるベース音の滑らかな進行を強調したりするなどの用法が考えられ，一概にその機能が同じであるとはいえない。

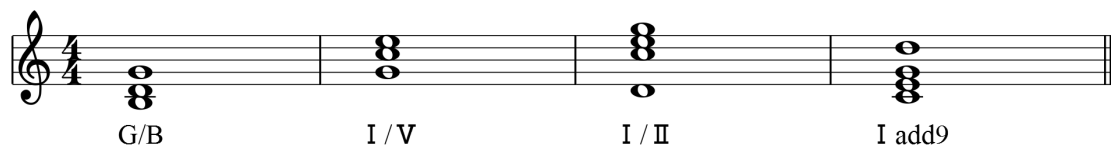


図 3.6: 分数コード表記

3.5 コード進行

ある調を構成する音のみで構成される和音のことをダイアトニックコードと呼ぶ。図 3.7 に C major におけるダイアトニックコードの例を示す。ダイアトニックコードには，楽曲における調性を確立し，スムーズなコード進行を印象付ける一定の並べ方が存在する。これを体系づけるものがコード理論である。

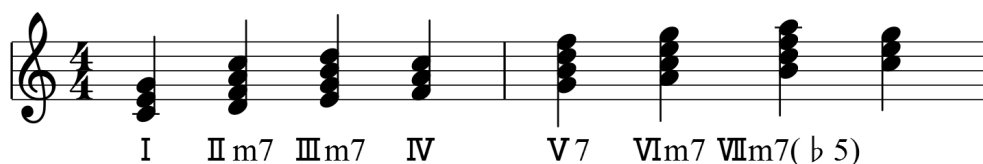


図 3.7: C major におけるダイアトニックコード

図 3.7 において，I 系統をトニック・コード (Tonic chord), IV 系統をサブドミナント・コード (SubDominant chord), V 系統をドミナント・コード (Dominant chord) と呼ぶ。I 系統を T, IV 系統を SD, V 系統を D と表記するとき，古典的な楽典においては $T \rightarrow SD \rightarrow D \rightarrow T$, $T \rightarrow D \rightarrow T$, $T \rightarrow SD \rightarrow T$ の 3 種類の進行が認められている (図 3.8)。D \rightarrow T の進行を特にドミナント進行といい，ドミナント・コードに含まれる不協和音がトニック・コードの協和音に滑らかに進行することで楽曲の収束感を生み出すことから，楽典における最も基本的で重要な進行とされる。サブドミナント・コードはドミナント・コードに滑らかに進行するという意味で，ドミナント進行を準備する役割をもつ。この一連の進行をカデンツ (cadence) という。

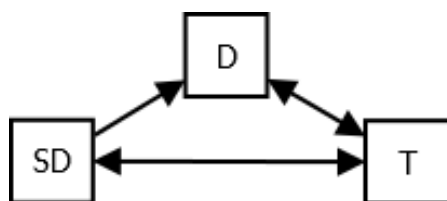


図 3.8: カデンツにおけるダイアトニックコードの進行

この進行は調の中で使用される最も基本的な進行であるが，調の中で使用される和音は，ダイアトニックコードだけに限らず，積極的に調性外音を用いたコードも使用される。これにより，調性をもたらす響きとは異なる雰囲気楽曲に付加することができる。しかし，そのようなコードは本来，調の中で違和感を覚える音を含んでおり，使い方や並べ方には一定の注意を払わなければならない。作曲を始めたばかりの初心者にとって，転回形も含めた多様な並べ方を選択し，積極的に楽曲に使用することは容易ではない。

3.6 終止形

一般に、楽曲は調の中で最も安定しているトニック・コードで終了し、一つの段落感をもたらす。この音楽の段落の終わりのことを終止と呼び、その種類によって以下のような呼び名がある。

全終止

V の和音から I の和音に移行する終止を全終止と呼ぶ。カデンツにもある通り、強い終止感をもたらす。特に、和音の転回がなく主旋律（ここでは最高音）が主音で終わるものを完全終止（図 3.9 右）そうでないものを不完全終止と呼ぶ（図 3.9 左）。

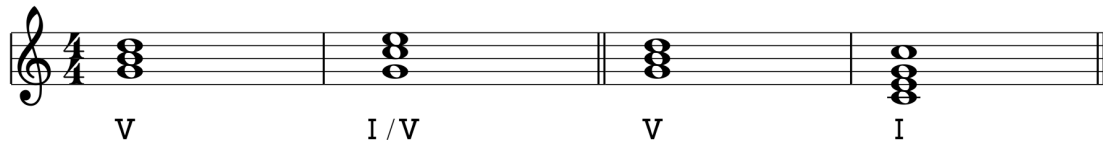


図 3.9: 全終止（不完全終止と完全終止）

半終止

V の和音で終止するものを半終止と呼ぶ。終止感はあまりない。

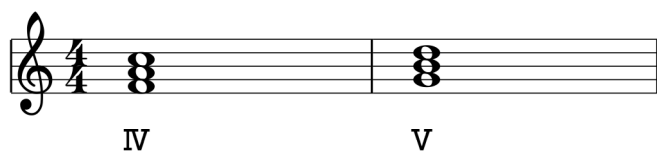


図 3.10: 半終止

偽終止

V の和音から I の和音以外の和音に移行する終止を偽終止と呼ぶ。聞く人に意外感をもたらす進行で、次の段落への接続感をもたらす。

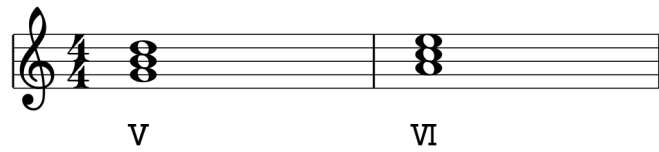


図 3.11: 偽終止

変終止

IV の和音から I の和音以外に移行する終止を変終止（アーメン終止）と呼ぶ。弱い終止感をもたらす。完全終止の後、さらにこの終止を付け加えるように用いられることもある。

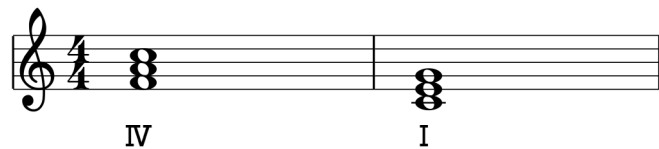


図 3.12: 変終止

女性終止と男性終止

拍子の始まりの拍を強拍、それ以外の拍を弱拍と呼ぶ。一般に、強拍のところに安定したコードを、弱拍のところに不安定なコードを置くことにより、不安定から安定に至る解決感をより安定させることができる。このとき、弱拍で終わる終止を女性終止と呼び、強拍で終わる男性終止とは区別されて呼ばれることがある。

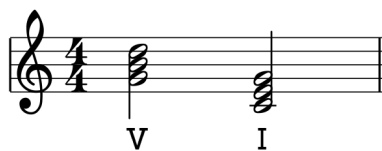


図 3.13: 女性終止

第4章 シーケンシャルパターンマイニング

本研究では、シーケンシャルパターンマイニングと呼ばれるデータマイニングの手法を用いてコード進行のパターンを予測し、提示する。本章ではシーケンシャルパターンマイニングについて説明する。

4.1 シーケンシャルパターンマイニングの定義

シーケンシャルパターンマイニング [16] とは、1995年に Agrawal らによって提唱された問題で、時系列に配置されたデータ列から頻出パターン系列を抽出するためのデータマイニング手法のことである。

出現する**アイテム**の集合を $\Sigma = \{i_1, i_2, \dots, i_m\}$ と表すとき、時系列のデータ列 S を**シーケンス**と呼び、 $S = \{a_1, a_2, \dots, a_r \mid a_i \in \Sigma\}$ と表す。なお、**アイテムトランザクション** T は、シーケンスのデータベースであり、シーケンス S_i と固有の識別子 sid_i のタプル $\langle sid_i, S_i \rangle$ の集合として表される。

$$T = \{\langle sid_1, S_1 \rangle, \langle sid_2, S_2 \rangle, \dots, \langle sid_n, S_n \rangle\} \quad (4.1)$$

シーケンス α 中の全てのアイテムが、別のシーケンス β 中に存在し、さらにその順序が保持されている場合、シーケンス α をシーケンス β の部分シーケンスであると定義し、 $\alpha \subseteq \beta$ と表す。ここで、トランザクション T を構成する全てのシーケンスのうち、部分シーケンス α を含むタプルの数を部分シーケンス α の support 値と定義する。

$$support_T(\alpha) = |\{\langle sid_i, S_i \rangle \in T \mid \alpha \subseteq S_i\}| \quad (4.2)$$

ただし、この定義では1つのシーケンス S に対して同じパターン α が t 回出現しても、support 値は1回として数えられる。これを t 回として数える場合の定義は次の通りである。

$$support(\alpha) = \sum_{i=1}^n |\{\alpha \subseteq S_i \mid S_i \in T\}| \quad (4.3)$$

また、2つのシーケンス $\alpha_{r-1} = \langle a_1, \dots, a_{r-1} \rangle$, $\alpha_r = \langle a_1, \dots, a_r \rangle$ (すなわち $\alpha_{r-1} \subseteq \alpha_r$) となるとき、 α_{r-1} の出現回数に対する α_r の出現回数の比率を、 α_{r-1} から α_r への confidence 値と定義する.

$$\text{confidence}(\alpha_{r-1} \rightarrow \alpha_r) = \frac{\text{support}(\alpha_r)}{\text{support}(\alpha_{r-1})} \quad (4.4)$$

この総積を定義することにより、任意の2つの部分シーケンスに対する confidence 値を定義することができる. すなわち、 $0 < i < r$, $\alpha_i \subseteq \alpha_r$ のとき、

$$\text{confidence}(\alpha_i \rightarrow \alpha_r) = \prod_{k=i}^{r-1} \text{confidence}(\alpha_k \rightarrow \alpha_{k+1}) = \frac{\text{support}(\alpha_r)}{\text{support}(\alpha_i)} \quad (4.5)$$

シーケンシャルパターンマイニングとは、任意の閾値 (**最小 support 値, minimum support**) ζ に対し、 $\text{support}(\alpha) \geq \zeta$ となるような部分シーケンス α を全て列挙する問題のことをいう. このような α を **頻出パターン** と呼ぶ.

抽出された頻出パターンは時系列的な順序を保っているものの、必ずしも隣接したアイテムの関係を抽出するわけではない. 頻出パターンにおける不連続アイテム間のことを **ギャップ** と呼ぶ.

4.2 アルゴリズム

シーケンシャルパターンマイニングを行うアルゴリズムとして、2001年に Pei らは PrefixSpan [17] を提案した. この手法では、prefix projection という射影方法とその射影によって生成される射影データベースを用いて頻出パターンを深さ優先的に探索する. まず、射影の概念について説明する.

射影元のシーケンス $s = \langle a_1, a_2, \dots, a_l \rangle$ と射影対象のアイテム a に対し、 $a_1 \neq a, a_2 \neq a, \dots, a_{j-1} \neq a, a_j = a$ となるような整数 $j (1 < j < l)$ が存在するとき、部分シーケンス $\langle a_1, a_2, \dots, a_j \rangle$ を s の a に対する prefix ($\text{prefix}(s, a)$) と定義する. また、部分シーケンス $\langle a_{j+1}, \dots, a_l \rangle$ を s の a に対する postfix ($\text{postfix}(s, a)$) と定義する. もし、 j が存在しない場合、prefix, postfix は未定義 (ϵ) となる. あるトランザクション T の各シーケンスに対してアイテム a を射影し、それらを改めてパターンのデータベースとしたものを **射影データベース** (projected database) $T|a$ と定義する.

$$T|a = \{ \langle sid, s' \rangle \mid (\langle sid, s \rangle \in T) \wedge (s' = \text{postfix}(s, a)) \wedge (s' \neq \epsilon) \} \quad (4.6)$$

次に、PrefixSpan の疑似コードを示す.

PrefixSpan の疑似コード

Require: トランザクション T , 最小 support 値 ζ

Ensure: 全ての頻出パターンの集合

```

1: Call PrefixSpan( $\langle \rangle, T$ )

2: Procedure PrefixSpan( $\alpha, T|\alpha$ )
3: Begin
4:    $B \leftarrow \{b | (s \in T|\alpha, b \subseteq s) \wedge (\text{support}_{T|\alpha}(\langle b \rangle) \geq \zeta)\}$       ▷ 5 章で拡張
5:   for all  $b \in B$  do
6:      $(T|\alpha)|b \leftarrow \{\langle sid, s' \rangle | (\langle sid, s \rangle \in T|\alpha) \wedge (s' = \text{postfix}(s, b)) \wedge (s' \neq \epsilon)\}$ 
7:     Call PrefixSpan( $\langle \alpha, b \rangle, (T|\alpha)|b$ )
8:   end for
9: End

```

このとき、最小 support 値を満たさなかったパターンを棄却し、再帰的探索の対象から除くことを**枝刈り**する、と呼ぶ。

頻出パターンの探索は深さ優先探索により再帰的に処理される。これにより無駄な候補シーケンスを生成せずに処理できるため、頻出パターン系列を高速に抽出することができる。

以上は原著の PrefixSpan アルゴリズムであるが、本稿ではさらにパターンの長さやギャップの長さ、ギャップの出現回数などを細かく指定し、枝刈りの条件として拡張したもの [18][19] をナイーブなシーケンシャルパターンマイニングとして扱う。

また、Agrawal らによるシーケンシャルパターンマイニングの定義では、1つのシーケンスに対して同じパターンが複数回出現しても、support 値は1回として数えられる。PrefixSpan の改良として、浅原は PrefixSpan-rel[20] をオープンソースとして公開している。PrefixSpan-rel では、1つのシーケンスに対して同じパターンが $n (n > 1)$ 回出現した場合、式(4.2)のように1回として数えることも、式(4.3)のように n 回として数えることもできる。本提案手法では、PrefixSpan-rel での式(4.3) 方式をベースにしてマイニング手法を拡張する。

第5章 提案手法

本章では、第3章で説明した音楽理論をもとに、第4章で紹介したナイーブなシーケンシャルパターンマイニングを拡張し、特徴的なコード進行の抽出に特化したマイニング手法を提案する。

シーケンシャルパターンマイニングを本問題で用いる場合、アイテムを実際の楽曲のコードに、シーケンスを1楽曲のコードシーケンスに、トランザクションを楽曲のコード進行データベースに対応させることで、頻出なコード進行の抽出を行うことができる。

本提案手法ではこれに加え、楽曲情報や楽曲構造を含むコード進行を抽出するために、アイテムを**基本コードアイテム**と**疑似アイテム**の2種類に分ける。これらを用いたコードシーケンスの表現方法を5.1節で述べる。なお、ここで述べるコードシーケンスは、一般に普及しているMIDI (XF[21]) 形式のデータから生成することを念頭に置いており、それにならう定義を行う。

続く5.2節では、5.1節で生成したコードシーケンスに対するマイニング手法として、以下に挙げるナイーブなシーケンシャルパターンマイニングの拡張手法を述べる。これにより、特定のコード進行の効率的な抽出や、楽曲情報・楽曲構造を考慮したパターンの抽出を可能にする。

- 終止形パターン発見のための逆順走査手法
- 各種アイテムとギャップの取り扱い
- 基本コードアイテムへの親子関係の導入
- コード進行の繰り返しパターンの抑制

なお、疑似アイテムの種類を属性として捉えたとき、属性を条件に含めたシーケンシャルパターンマイニングをSatoらが提案している[22]。しかし、本問題においては出現するアイテムの種類によってマイニングの手続きを変える必要があるため、この手法を単純に適用することができない。

5.1 楽曲のコードシーケンス表現

本提案手法では、アイテムを大きく**基本コードアイテム**と**疑似アイテム**の2種類に分ける。疑似アイテムには、さらに3つの小分類を設け、表5.1のようにアイテムを導入する。

表 5.1: 本提案手法で導入するアイテムの種類

種類	小分類	定義
基本コードアイテム		Σ_C
疑似アイテム	メタ情報アイテム	Σ_M
	小節線アイテム	$\Sigma_B = \{\text{bar}\}$
	構造アイテム	$\Sigma_S = \{\text{START}, \text{END}\}$
	セパレーターアイテム	$\Sigma_{sep} = \{-\}$

これらのアイテムを用いて、コードシーケンス S_i を定義する。

$$S_i = \langle m_{1,-}, m_{2,-}, \dots, m_{n,-}, \text{START}, -, c_1, \dots, c_m, -, \text{END} \rangle \quad (5.1)$$

後述する逆順走査で用いるコードシーケンス S'_i は次のように定義する。

$$S'_i = \langle \text{START}, -, c_1, \dots, c_m, -, \text{END}, -, m_{n,-}, m_{n-1,-}, \dots, -, m_{1,-} \rangle \quad (5.2)$$

ただし、

$$m_i \in \Sigma_M \quad (5.3)$$

$$c_i \in \Sigma_C \cup \Sigma_B \quad (5.4)$$

である。

以下、各アイテムについて詳細を述べる。

5.1.1 基本コードアイテム

楽譜等に出現するコードネームを、根音・コードの種類・最低音の組み合わせとして表したアイテムであり、その集合を Σ_C で表す。ここで、コードアイテム $c_i \in \Sigma_C$ には C, Am などの音名表記ではなく、I から VII のローマ数字による音度記号を用いた階名表記を用いる。入力を音名表記で受け取った場合は、その楽曲の調を基準として変換を行う。なお、曲の途中で転調がある場合は、その時点から転調先の調に基づいて変換を行うものとする。以下、単にコードアイテムと呼称した場合も、基本コードアイテムを指すものとする。

5.1.2 疑似アイテム

コードそのものではなく、楽曲情報や楽曲構造など、付加的な情報を表したアイテムである。以下に示す3つの小分類を合わせたもので、 $\Sigma_M \cup \Sigma_B \cup \Sigma_S$ で表す。

メタ情報アイテム

楽曲のメタ情報として含まれる情報を表したアイテムであり、その集合を Σ_M で表す。作曲家・ジャンル・制作年・制作地などが該当する。これらは基本コードアイテムの前に走査するものとし、コード進行データベース上では基本コードアイテムの前に置かれる。

メタ情報アイテムを導入することにより、コード進行と楽曲情報を関連付けて抽出することができる。例えば、ある作曲者はこのようなコード進行を多用する、という情報をパターンとして抽出できる。

小節線アイテム

楽曲の拍子情報をもとに、小節線にあたる部分に付加するアイテムであり、本稿では“bar”と表記する ($\Sigma_B = \{\text{bar}\}$)。コードが小節線をまたぐ（小節線を挟んで同じコードが存在する）場合は、その数だけ小節線アイテムを付加する。ただし、小節線アイテムの後に同じコードアイテムを追加することはしない。

小節線アイテムを導入することにより、そのコードが強拍であるか、弱拍であるかの情報を得ることができる。3.6節で述べた通り、弱拍で終わる女性終止と、強拍で終わる男性終止は区別されることがある。すなわち、同じコード進行であっても強拍か弱拍かで役割が異なると考えることができ、例えば、 $\langle \text{bar}, V, I \rangle$ と $\langle V, \text{bar}, I \rangle$ は区別して抽出されることが望ましい。

また、既存手法ではコードアイテムには音価が一切考慮されていなかった。しかし、小節線を付加することにより、同じコード進行であっても小節線を隔てているか否かで出力パターンを区別することができる。これにより、完全とはいええないものの、音価の違いをパターンに反映することができる。

構造アイテム

楽曲の開始・終了などの楽曲構造を表したアイテムであり、それぞれ“START”、“END”と表記する ($\Sigma_S = \{\text{START}, \text{END}\}$)。

楽曲の開始と終了を表すアイテムを導入することにより、楽曲の開始と終了に使用されるコード進行に特化したパターンを抽出することができる。また、後述

する逆順走査手法においても、曲の開始であるか終了であるかを明確に区別することができる。

セパレーターアイテム

セパレーターアイテムは主に以下の2つの目的でコード進行データベースに付加されるアイテムで、これを“_”と表記する ($\Sigma_{sep} = \{_ \}$)。このアイテムは提案手法では一切数え上げを行わず、単にコード進行データベースを走査するときの目印となるものである。したがって、出力パターンには出現しない。

- メタ情報アイテム同士の区別

メタ情報アイテムはシーケンシャルなアイテムではなく、その出現順序によってコード進行の出現頻度に影響を及ぼすことがあってはならない。また、作曲者とジャンルは1曲に対して複数個定義される可能性があり得る。複数のメタアイテムをコード進行データベースに付加した場合、それぞれが独立して出現するパターンと共起するパターンのいずれも等価に抽出されるべきである。そこで、任意のメタ情報アイテムに対し、その直後にセパレーターアイテムを1つ付加する。

- 構造アイテムの前後の表記揺らぎの防止

構造アイテムの目的から、その直前直後のコードアイテムの存在は大変重要である。しかし、MIDI データからコード進行データベースを生成する場合、楽曲によっては楽曲の開始直後や終了直前にコードアイテムが存在するとは限らず、小節線アイテムを挟んでいることがある。また、小節線アイテムの数も、意図的に小節が隔てたものなのか、データ上の仕様なのかを区別することはできない。そのため、コードアイテムが隣接しているパターンとそうでないパターンを区別して support 値を数え上げることは望ましくない。そこで、“START”の直後と“END”にセパレーターアイテムを1つ付加する。

5.2 ナイーブなアルゴリズムの拡張方法

コード進行データベース D において、以下に述べる手法をもとに抽出されたコードシーケンス S の出現回数が m 回のとき、 $m = support(S)$ である。ここでは、 S が閾値として設定した ζ 回以上出現した場合に頻出なコード進行として抽出する問題として定義する。

このとき、パターンの抽出を下記に挙げる順序で行う。これは4.2節で説明した PrefixSpan の疑似コードの4行目にあたる処理を拡張したものである。

1. 対象となるデータベースを正順走査および逆順走査によって数え上げ、候補パターンとする (5.2.1 節にて説明).
 - 数え上げの際は、原則としてギャップを含まない隣接したパターンを抽出する。ただし、prefix に応じて表 5.2 の条件で後続のアイテムに対するギャップを許容して数え上げる (5.2.2 節にて説明)。
 - 親子関係があるアイテムを走査した場合は、親アイテムについても出現したものとみなして数え上げる (5.2.3 節にて説明)。
2. 候補パターンから support 値が ζ 以上であるパターンのみを抽出し、他を枝刈りする。ただし、走査したアイテム i に最小アイテム support 値 ζ_i がある場合は、これを満たす (5.2.2 節にて説明)。
3. 同じパターンの繰り返しとなる冗長なパターンは枝刈りする (5.2.4 節にて説明)。

頻出パターンとして抽出されたコード進行のうち、support 値が高いものは一般的に使われているもの、support 値が低く confidence 値が高いものは、珍しいが一方向性があるものを意味する。確率的な最適化手法と異なり、コード進行が一般的に使用されているという妥当性を保ちながら、音楽的に珍しいコード進行や意図的な音楽的表現を損なわずに抽出することができる。

5.2.1 終止形パターン発見のための逆順走査手法

通常、シーケンシャルパターンマイニングはコードシーケンス $S = \langle c_1, \dots, c_m \rangle$ に対し、 c_1 から c_m に向かって走査を開始する。これを c_m から c_1 に向かって逆順に走査することを考える。

この場合、confidence 値は頻出パターン $\alpha = \langle a_l, \dots, a_1 \rangle$, $\beta = \langle a, a_l, \dots, a_1 \rangle$ に対して、 α の出現回数に対する β の出現回数の比率であり、正順走査したときの値とは異なるため、以下正順走査時の confidence 値を正順 confidence 値、逆順走査時の confidence 値を逆順 confidence 値と呼称する。正順と逆順を記述しない場合は、正順 confidence 値を表すものとする。

5.1 節でのコードシーケンスの定義より、アイテム $c_j = I$ に対して頻出パターン系列を探索することで、終止形のコード進行に関する頻出なパターンを抽出することができる。このパターンには全終止 $\langle V, I \rangle$ や偽終止 $\{\langle V, c \rangle | c \neq I\}$ や変終止 $\langle IV, I \rangle$ などが含まれている。

5.2.2 各種アイテムとギャップの取り扱い

本提案手法では，アイテムの種類に応じてギャップを一定の条件で認めることにより，計算量を抑えながら多様なコード進行を抽出できるようにする．ここで，長さ 1 以上の任意長のギャップを“//”と表記する．抽出したシーケンスが $\langle a, //, b \rangle$ であるとき，コード進行データベース上の a と b は 1 個以上のアイテムを隔てていることを意味する．ただし，その長さは区別せず，単に 2 つのアイテム間が隣接していないという情報のみを表す．このとき，任意のアイテム i_0 とその射影データベース $D|i_0$ に出現する後続アイテムに対する数え上げの有無と，ギャップの走査範囲についてまとめると，表 5.2 のようになる．以下ではこの制御方法についてアイテムごとに詳しく述べる．

表 5.2: アイテムの種類による数え上げの有無と後続のギャップの走査範囲

			後続アイテム					ギャップ走査範囲
			Σ_C	Σ_M	Σ_B	Σ_S	Σ_{sep}	
対象 i_0	Σ_C	隣接	○	—	○	○	×	直近のコード アイテムまで
		ギャップ“//”	○	—	×	×	×	
	Σ_M	隣接	×	×	×	×	×	後続のアイテム 全て
		ギャップ“//”	○	○	○	○	×	
	Σ_B	隣接	○	—	○	○	×	なし
		ギャップ“//”	×	—	×	×	×	
	Σ_S	隣接	×	—	×	△	×	直近のコード アイテムまで
		ギャップ“//”	○	—	○	△	×	
	Σ_{sep}	隣接	—	—	—	—	—	なし
		ギャップ“//”	—	—	—	—	—	

○ 数え上げる

× 数え上げない

— コードシーケンス表現の定義上あり得ない

△ 未定義動作だが事実上そのようなケースが存在しない

基本コードアイテムとギャップ

コード進行は前後のコードとの関係が非常に重要である。したがって、基本コードアイテムを走査した場合は、原則として隣接するコードアイテムのみを走査するものとする。ただし、隣接するアイテムが小節線アイテムであった場合に限り、後続の直近のコードアイテムまで走査しギャップ“//”を認める。ここで、やみくもにギャップを増やすことを防止するため、直近のコードアイテムに至るまでに走査した非コードアイテムは数え上げない。

メタ情報アイテムとギャップ

5.1 節でのコードシーケンスの定義により、アイテムは走査時に始めに探索されることになる。提案手法では、メタ情報アイテム m_k の射影データベース $D|m_k$ を走査し、後続のアイテム全てに対してギャップ“//”を許容する。5.1 節での定義から、メタ情報アイテムに隣接するアイテムは全てセパレーターアイテムであり、メタ情報アイテムと後続のアイテムが隣接しているかどうかは考慮されない。

また一般に、1つのコードシーケンス内におけるメタ情報アイテム m_k の出現個数は、コードアイテム c_j の出現個数より遥かに少ない。したがって、共通の最小 support 値を設定して抽出を行った場合、相対的に低い support 値をもつ m_k を含むパターンは抽出されにくくなる。そこで、最小 support 値 ζ とは別に、各作曲家ごとの最小アイテム support 値 ζ_{m_k} を定義する。ここでは一般的に ζ_{m_k} は ζ よりも低く設定し、走査の段階でメタ情報アイテム m_k を発見した場合は、 ζ ではなく ζ_{m_k} を優先して適用する。これにより、 m_k を含むパターンが枝刈りされることを防ぐことができる。

小節線アイテムとギャップ

小節線アイテム“bar”の射影データベース $D|bar$ について、隣接アイテムの走査のみを行い、ギャップは挿入しない。

これにより、既存手法における頻出パターンの中に、小節線アイテムを含むものが現れる。例えば、既存手法における頻出パターン $\langle V, I \rangle$ は $\langle V, I \rangle$, $\langle V, bar, I \rangle$, $\langle V, bar, bar, I \rangle$, $\langle V, bar, bar, \dots, I \rangle$ などに分けて数え上げられる。このうち、1個以上の小節線アイテムを含むパターン $\langle V, bar, \dots, I \rangle$ は $\langle V, //, I \rangle$ というパターンとしてまとめて認識される。このパターンの support 値は個々のパターンの support 値の和と一致する。

構造アイテムとギャップ

提案手法における構造アイテム s の射影データベース $D|_s$ に対する走査は、直近のコードアイテムまでを対象とする。ただし、5.1.2 節で述べた理由により、構造アイテム “START” の直後と “END” の直前には必ずセパレーターアイテムが含まれる。このため、走査したコードアイテムとの間には必ずギャップ “//” が含まれ、隣接しているかどうかは考慮されない。

5.2.3 基本コードアイテムへの親子関係の導入

本問題においては、複雑なコードほど support 値が小さく、枝刈りされやすい傾向にある。しかし、作曲支援の観点では、少なく数え上げられる複雑なコードについても、有用なものがあると考えられる。これをアイテムへの親子関係の導入によって緩和することを考える。

まず、コードの同値関係として 3 つの同値関係を定義する。この 3 つのうちでは、機能的同値、音高同値、転回同値の順で同値関係が強くなるものとする。

- 機能的同値

I, I_6, I_{M7}, I_{M9} などは、任意の調におけるトニックとして扱われるため、音楽理論の機能的に等しいコードアイテムであると考えられる。この他、サブドミナント・ドミナントなどに対しても同様に定義することができる。

- 音高同値

異名同和音 (IV_6, II_{m7} など) は、同じ構成音をもつため、コード進行の中での役割も等しいコードアイテムであると考えられる。

- 転回同値

$I, I/III, I/V$ など、転回によって基本形に等しくなるものを役割が等しいコードアイテムであると考えられる。ここでは、 I/II のように、オンベースノートの付加を分数コードで表したコードアイテムについては、これを同値としない。

本稿では、転回同値のみを取り扱う。機能的同値と音高同値はともに、対象となるアイテムの support 値が比較的大きく、既存手法でも十分抽出できると考えられるからである。一方、転回形となっているコードアイテムは、基本形のコードアイテムに比べて support 値が小さいと考えられ、等価に数え上げることが妥当であると考えられる。

相関分析におけるアイテムの階層化については、Srikant らによって提案されている [23]。もっともナイーブな手法は、データベース中の階層構造をもつ各アイテ

ムに対して、全ての上位概念アイテムをデータベースに付加してマイニングを行うというものである。このとき、付加した上位概念アイテムは下位概念アイテムと同時生起するものとみなされる。このとき、自明な相関ルールが多数数え上げられ、計算量の増大につながるため、意味のあるパターンだけを抽出し、出力数を抑える抽出手法が提案されている。

本提案手法では、データベース中に存在する転回同値にあたる基本コードアイテム全てに対して、1階層上位の親コードアイテムを設定する。本問題において、基本コードアイテム同士が同時生起することはないが、転回同値にあたるコードアイテムを走査した場合に限り、親コードアイテムも同時生起したのものとして数え上げを行う。

例えば、基本コードアイテム I, V について、データベースに含まれる転回形コードを子に持つ親コードアイテム I', V' を設定する (図 5.1)。

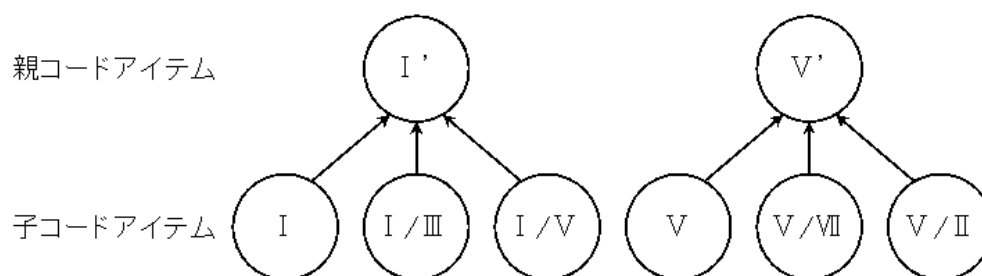


図 5.1: アイテムの親子関係の例

ある頻出パターン $\alpha = \langle c_1, \dots, c_n, a \rangle$ があり、 $a \in \{V, V/VII, V/II, V'\}$, $c \in \{I, I/III, I/V\}$ であるとき、後続するアイテム c が表 5.3 のように数え上げられたとする。このとき、最小 support 値 ζ を小さくしていくごとに、パターンは

$$\langle c_1, \dots, c_n, V', I' \rangle, \langle c_1, \dots, c_n, V, I' \rangle, \langle c_1, \dots, c_n, V', I \rangle, \langle c_1, \dots, c_n, V, I \rangle, \dots$$

の順で抽出される。さらに長いパターンを探索する場合は、抽出されたパターンの全てについて後続アイテムの数え上げを行う。

上記手法によってアイテムの親子関係を導入する場合、support 値が小さくなりがちな複雑なコードを枝刈りされにくくする他にも利点がある。それは、親コードアイテムと子コードアイテムを差別化することで、好んで使用される子コードアイテムを顕在化することができるということである。

一般に、同値関係を適用することによって、親コードアイテムを含むパターンが高い最小 support 値で抽出されるようになり、それに次いで基本コードアイテムを直接含むパターンが抽出される。このとき抽出された基本コードアイテムは重

表 5.3: 親子関係導入後の数え上げ方の例

c	I	I/III	I/V	I'
$support(\langle c_1, \dots, c_n, V, c \rangle)$	454	20	12	486
$support(\langle c_1, \dots, c_n, V/VII, c \rangle)$	25	2	0	27
$support(\langle c_1, \dots, c_n, V/II, c \rangle)$	0	6	0	6
$support(\langle c_1, \dots, c_n, V', c \rangle)$	479	28	12	519

要なものであり、一部の進行ではその転回形が好んで用いられるということの意味するようになる。

5.2.4 コード進行の繰り返しパターンの抑制

例えば、出力パターンとして以下のようなコードが与えられたとする。

$$\langle V, I, IV, I, IV, I, IV, I, IV \rangle \quad (5.5)$$

楽曲によっては、このように一定のコード進行が繰り返されるケースがある。これらは音楽的知識の直感的な発見という観点からみると、変化に乏しく他の頻出パターン系列によって補完することができるパターンであるといえる。また、出力パターンの数が増大することで、計算量の増大にも繋がる。

そこで、support 値による枝刈りの後、任意のコードシーケンス S に対して、全ての疑似アイテムを取り除き、基本コードアイテムのみを残したシーケンス S' をつくる。このとき、次のコードで示される S' を枝刈りの対象とし、出力を抑制する。ここで、 s, t は任意のコードシーケンスであり、基本コードアイテムのみを含む。また、 $?$ は直前のコードシーケンスが 0 回または 1 回、 $+$ は 1 回以上繰り返し出現することを表す。

$$S' = \langle s?tt+ \rangle \quad (5.6)$$

このようなパターンを、本稿では**繰り返しパターン**と呼称する。

式(5.5)の場合、 $s = \langle V \rangle, t = \langle I, IV \rangle$ となる。このとき、枝刈りによって $\langle V, I, IV, I \rangle$ までが頻出なコード進行として出力される。

また、この定義にならうことにより、任意のアイテム a が連続する、 $\langle a, a \rangle$ のようなパターンも枝刈りの対象となる。

第6章 作曲支援システム

本章では、第3章で紹介した理論と第5章で提案した手法を、作曲支援にどのように生かすかという観点から、作曲支援システムの設計と実装について説明する。

6.1 提案システムの設計

第1章で述べた通り、作曲を始めたばかりの初心者は、コード理論の習得を困難としている。また、コード理論を考慮した上でコード進行を MIDI シーケンサーや市販の楽曲制作ソフト上で実現し、適切に構築しなければならない。そこで提案システムでは、任意のコード進行に対し、その続きを予測しユーザーに提示する。これによって、コード理論の習得という作曲初心者がもつ障壁を緩和し、適切なコード進行を自然に実現する。また、提示されたコード進行をユーザーに選択させ、自身の楽曲制作に積極的に使用することにより、自然なコード進行をもつ楽曲の制作を補助する。これをまとめると、提案システムで求められる要件は次の通りである。

提案システムの要件

- ユーザーが任意のコードを入力すると、それに続くコード進行を予測し提示する。
- システムによって提示されたコード進行をユーザーが選択することによって、楽曲のコード進行を構築する。
- 選択したコード進行を再利用可能な形で提供する。

提案システムでは、シーケンシャルパターンマイニングを拡張して適用することにより、コード進行を予測する。ここでは、古典的な音楽理論にならったり、確率遷移を利用して提示するのではなく、実際の楽曲に使用されているコード進行を利用する。これにより、大衆的に評価を得ている説得力のあるコード進行を提示することができ、時代に沿った音楽的表現を損なわずに済む。また、出力の条件を細かに指定することができるため、ユーザーの求めるコード進行の予測・抽

出に繋がりがやすいと考えられる。

取り扱うコードは第 3 章で説明したコードを対象とする。抽出されるコード進行は、階名表記でのコードのため、制作する楽曲の調を入力として与える必要がある。また、選択したコード進行は SMF(Standard Midi File) 形式のファイルとして出力する。このデータは直接、MIDI シーケンサーや市販の楽曲制作ソフトウェアに読み込ませることができる。したがって、本システムで出力された MIDI データを補完するように作曲することで、自然なコード進行を既に与えられた状態から作曲することができる。

実際のシステムの操作は、次の手順を経る。

—— システムの操作 ——

1. ユーザーは、出力する楽曲の調を指定する。
2. ユーザーは、画面上の直接入力インターフェース、GUI キーボード、外部接続の MIDI キーボードのいずれかのデバイスを使ってコードを入力する
3. システムは、入力されたコードを認識する。
4. システムは、認識されたコード進行に対して、その続きにあたるコード進行を提示する。
5. ユーザーは、提示されたコード進行から作曲に使用するコード進行を選択し、選択してきたコード進行の末尾に追加する。
6. ユーザーは、選択してきたコード進行を再生する。
7. 続きのコードが再生されると、システムは次のコードを認識し、その続きにあたるコード進行を提示する。
8. 5 から 7 を繰り返すことにより、ユーザーが制作する楽曲のコード進行が完成する。
9. ユーザーは、コード進行を保存する。
10. システムによって、対応する MIDI データが出力される。
11. ユーザーは、出力された MIDI データをコードとして利用し、メロディーやリズムの作曲を行う。

したがって、システムは内部にコードの認識 (3)・マイニング (4, 7)・コード進行

の MIDI データ化 (5, 10) を行う各モジュールを持つこととする。

6.2 提案システムの実装

提案システムの大まかなフローチャートを図 6.1 に、実際の提案システムの GUI を図 6.2 に示す。システムは全て Java SE 6 を用いて実装した。以下、フローチャートで示したそれぞれのモジュールの実装について説明する。

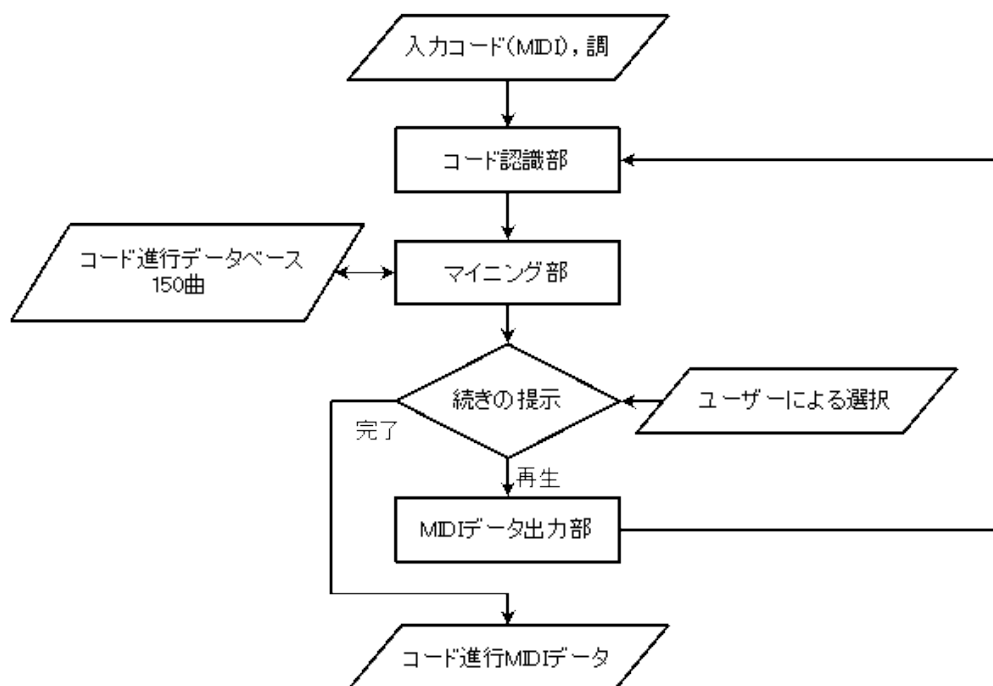


図 6.1: 提案システムのフローチャート

6.2.1 入力

入力として与えるべきデータは、具体的なコードと出力するコードの調の 2 つである。コードの入力はアプリケーション GUI 上の直接入力インタフェースまたは GUI キーボードをタッチするか、接続されている MIDI 入力デバイスを通して入力できる。コードは三和音から五和音で構成されているため、入力の際は該当する音程を同時に入力する。調の指定は GUI インターフェースのみで行う。たとえば C Major, B^b minor などと指定することで、後述のコード認識部において、入力情報を該当する音階上のコードとして解釈する。また後述の MIDI データ出力部でも、指定された調に基づいて MIDI データが生成される。

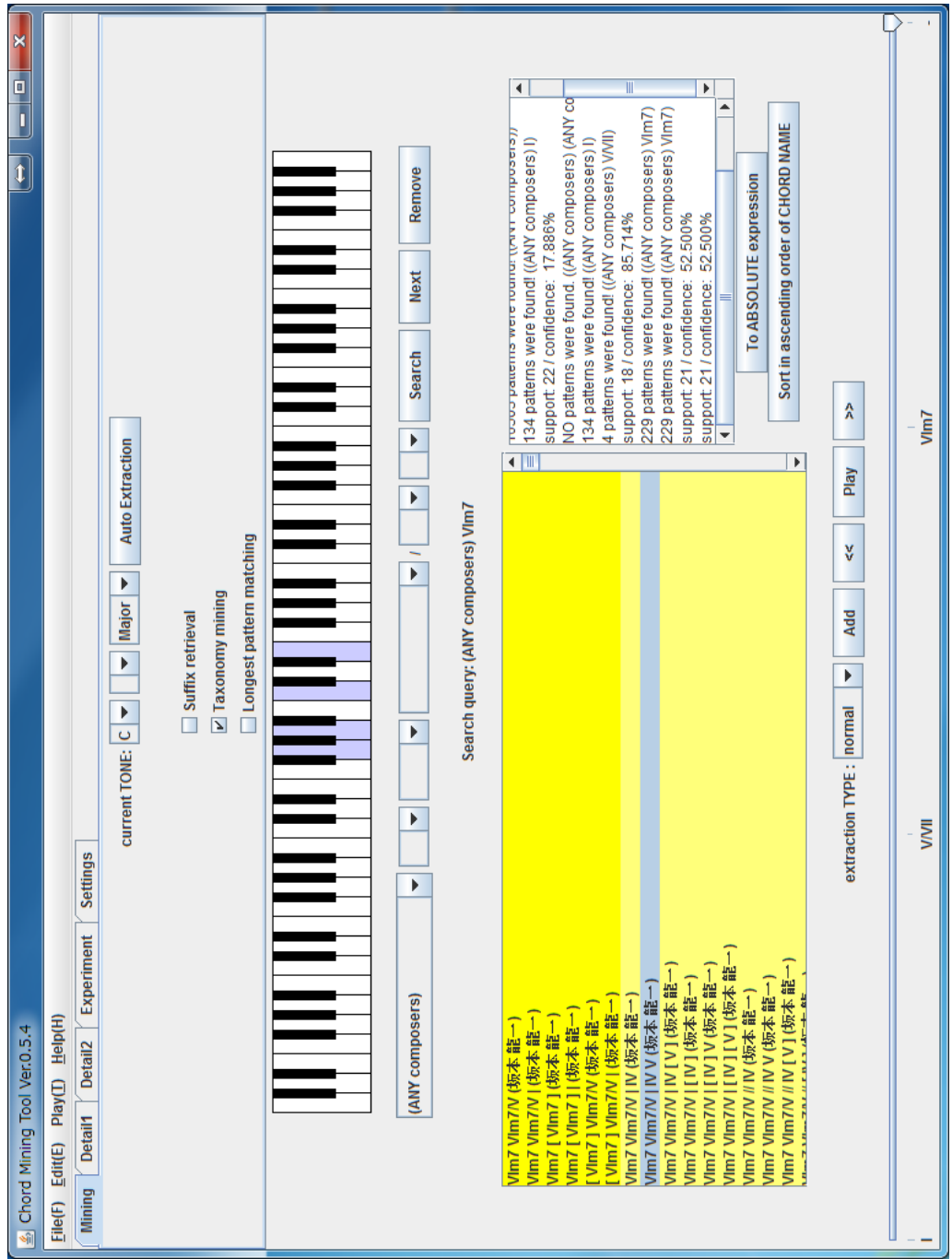


図 6.2: 提案システムの GUI 図

6.2.2 コード認識部

入力で与えられたコードは、システム内部では MIDI 規格に対応した MIDI メッセージとして送られる。MIDI メッセージは MIDI イベント、メタイベント、システムエクスクルーシブイベントの 3 種類のいずれかに時間情報であるデルタタイムを加えた仕様となっている。このうち、本システムで必要となるのは、MIDI イベントのうち、鍵盤を押したことを表すノートオン・メッセージと、鍵盤から離れたことを表すノートオフ・メッセージの 2 つである。この 2 つは音階を示すノート番号と打鍵の強さを表すベロシティの 2 つの情報を持つ。音階名は通常、C4, F7 などと音名とオクターブを併記して区別する。一方 MIDI 規格におけるのノート番号は、C4=60 を基準として半音毎に 1 つずつ増減していき、最低音を 0、最高音を 127 とする範囲での音階名を区別する。そこで、ノート番号からオクターブ情報を除いた絶対的な 12 音名 *note* を識別する場合、与えられたノート番号 *n* を式 (6.1) のように 12 の剰余にして識別する。

$$note = n \bmod 12 \quad (6.1)$$

入力で指定された調性で認識を行うとき、コードのルート音が何度の音であるかを検出する必要がある。そのため、調の根音にあたる 12 音階上のずれ *t* を減算してから式 (6.1) を適用する。すなわち、相対的な 12 音名 *note'* は

$$note' = (12 + n - t) \bmod 12 \quad (6.2)$$

と表される。なお、12 を加算するのは結果が負数になることを防ぐためである。

実際にノート番号から 3.3 節で述べたコードを認識する過程では、次の処理を経る。

1. 同時に鳴っているノート番号を転回し、2 オクターブの範囲内 (24 半音以内) に収める。
2. ノート番号を昇順でソートする。
3. 最低音からの間隔をみて、3.3 節で対象としたコードに合致するかどうかを判断する。このとき、五和音、四和音、三和音の順で合致するかどうかをみる。合致した場合は、対応するアイテムをマイニング部に渡す。
4. 合致しなかった場合は、転回形であるとみなし、鳴っているノートを 1 つずつ転回させ、2. 3. の処理を繰り返す (転回形優先探索)。
5. それでも合致しなかった場合は、オンベースノートが付加されているものと判断し、最低音とそれ以外のコードに分けて再び 2. 3. 4. の処理を繰り返す (オンベースノート優先探索)。

6.2.3 マイニング部

コード認識部から認識されたコードが渡されると、それを検索クエリとして第 5 章で示した提案手法により後続のコード進行を導出する。

提示されるパターンは、 $\langle III_m, IV, V_7, VI_{m7} \rangle$, $\langle V_{sus4}, V \rangle$ など、入力されたコードに後続すると考えられる具体的なコード進行を表す。条件を満たすパターンは、ユーザーが制作する楽曲のコード進行の候補として support 値の高い順に GUI 図中央部分の候補エリアに出力される。ユーザーは候補から作曲に使用するコードを 1 つ選択し、GUI 図の Add ボタンを押すと、選択したコード進行が追加され、GUI 図下部のシークバーに表示される。Play ボタンを押して再生すると、システムは後続のコード進行を次々とマイニングし、続きのコード進行を候補エリアに提示し続ける。なお、間違って入力をしたコードは、アンドゥすることができる。

マイニング部の実装は PrefixSpan-rel[20] にならい、最小 support 値、パターン長、ギャップ数、ギャップ長など、抽出するパターンの内容をユーザーが細かく設定できるようにした。また、提案手法における逆順走査や親子関係の適用は、ユーザーが指定できるようにした。

6.2.4 MIDI データ出力部

候補の選択と再生を繰り返すことで、妥当性を持った一連のコード進行が完成する。MIDI データ出力部では、コードアイテムを調の情報をもとにノート番号を含む MIDI メッセージへ変換する。すなわち、内部のアルゴリズムはコード認識部の逆の変換を行っている。ここでは、1 つのコードを 1 小節に割り当てるよう MIDI メッセージを配置することとした。ユーザーが選択したコード進行を再生するときは、内部に蓄積した MIDI データを再生する。制作したコード進行を再利用可能な形で出力するときは、内部に蓄積したコード進行をあらゆる MIDI メッセージ群を SMF 形式で出力する。

6.2.5 出力

出力は MIDI データ出力部によって得られた、選択したコード進行に対応する MIDI データである。

第 7 章 評価実験

本章では提案手法を用いた各種実験を行い、結果と考察について述べる。

7.1 実験の方針

7.1.1 実験の目的

実験では、特徴的なコード進行の抽出問題に対してシーケンシャルパターンマッピングを拡張して適用したことによるナイーブな手法との違いや効果を調べるため、各拡張手法に対して以下の項目について検討した。

1. ナイーブな手法と比較して、転回同値となる親子関係を導入したとき、未発見だったパターンが出力されるようになったかどうかを確認した。
2. メタ情報アイテムを含むパターンが、適切にメタ情報アイテムの特徴を表しているかどうかを確認した。
3. 小節線アイテムを含む頻出パターンが、どのような音楽的特徴を表しているかを確認した。
4. 構造アイテムを含む頻出パターンが、どのような音楽的特徴を表しているかを確認した。

7.1.2 実験手法

以下、各実験の手法について述べる。ただし、以下の実験では全て、現在 YAMAHA 社がメタデータを含む MIDI (XF[21]) 形式として販売している、J-POP やロック、洋楽を含む楽曲 150 曲のコード進行をデータベースとした。また、対象とするコードの種類も XF 形式がサポートするものを対象とし、分数コード表記にも対応した。

アイテムへの親子関係の導入と出力パターンの比較

コード進行データベースに対して転回同値となる親子関係を導入したとき、各パターンに出現するアイテムをすべて親コードアイテムに置き換えたパターンを**基底パターン**と呼称する。例えば、図 5.1 と同じ親子関係が適用されるとき、 $\langle I, V/VII, IV_{M7}, \dots \rangle$, $\langle I/V, V/VII, IV_{M7}, \dots \rangle$ などの基底パターンは $\langle I', V', IV_{M7}, \dots \rangle$ である。ここでは、親子関係適用前に抽出された基底パターンを旧基底パターン、親子関係適用後に新規に抽出された基底パターンを新基底パターンと呼称する。

作曲支援を目的としたマイニングにおいて最も有益なパターンは、support 値の低い、新基底パターンに由来する新規出力パターンと考えられる。support 値が高い新規出力パターンは、親子関係を適用しない場合でも最小 support 値を減らせば出力される可能性があるためである。

そこで、ナイーブな手法と提案手法における基底パターンに着目し、各最小 support 値と出力パターン数の変化を比較した。なお、ここでは純粹に親子関係についてをみるため、コード進行データベースに対し、疑似アイテムの付加は行わずに実験を行った。

メタ情報アイテムを含むパターンの抽出実験

ここではメタ情報アイテムとして作曲者情報とジャンル情報を取り上げる。便宜上これらをそれぞれ作曲者アイテム、ジャンルアイテムと呼称する。作曲者ごとやジャンルごとに特徴がある場合、メタ情報アイテムのない通常のパターンに比べ、メタ情報アイテム $m_k \in \Sigma_M$ を含むパターンは confidence 値に有為に差があると考えられる。confidence 値は、ある頻出パターン α に対して、これを部分シーケンスにもつシーケンス $\beta (\alpha \subseteq \beta)$ の分布の比率である。したがって、帰無仮説を「メタ情報アイテムを含むコード進行と含まないコード進行では、分布の比に有意差はない」とし、有意水準を 5% とした χ^2 検定を行うことで、作曲者やジャンルに特徴的なコード進行であるかどうかを示す。有意差があると判定された場合は、作曲者アイテムを含むコード進行の分布はメタ情報アイテムを含まないコード進行の分布とは似ておらず、その進行の仕方は作曲者やジャンルに特有であるということができる。

なお、ここではコード進行データベースに対し親子関係の適用や小節線アイテムの付加は行わず、最小 support 値は $\zeta = 2, \zeta_{m_k} = 1$ として可能な限りのパターンを抽出した。

本稿では、コード進行データベースに含まれる楽曲が 4 曲である 5 名の作曲者と 3 曲である 5 名の作曲者の作曲者アイテム含むパターンと、3 つのジャンルのジャ

ンルアイテムを含むパターンに対し、以下の3つの抽出条件について実験した。

1. 正順走査における頻出パターン $P_1 = \langle I \rangle$, $P_2 = \langle m_k, //, I \rangle$ に対して、それぞれ $P'_1 = \langle I, \alpha \rangle$, $P'_2 = \langle m_k, //, I, \alpha \rangle$ の support 値を一つの分布として捉え、confidence 値に有意差があるかどうかを調べた。ただし、 α は長さ 1 以上 3 以下の任意のパターンであり、 P'_i は長さ 2 以上 4 以下のパターンとなる。I はトニックであり、どのコードに進行しても禁則とはならないため、後続の進行の仕方によって作曲者に特徴が出るものと考えられる。
2. 逆順走査における頻出パターン $P_1 = \langle I \rangle$ と $P_2 = \langle I, //, m_k \rangle$ に対して、それぞれ $P'_1 = \langle \alpha, I \rangle$, $P'_2 = \langle \alpha, I, //, m_k \rangle$ の support 値を一つの分布として捉え、confidence 値に有意差があるかどうかを調べた。ただし、 α は長さ 1 以上 3 以下の任意のパターンであり、 P'_i は長さ 2 以上 4 以下のパターンとなる。自由な進行が許される前節とは異なり、全終止 $\langle V, I \rangle$ など、音楽理論で終止形となるパターンが含まれている。
3. メタ情報アイテムを含むパターンのうち、confidence 値が高いものは有為差が高いかどうかを調べるために、作曲者アイテムを含むパターンのうち、support 値の高い順に confidence 値が 1.00 (定義上の最大値) となるものについて χ^2 検定を行った。なお、この実験は作曲者アイテムのみについて行った。

小節線アイテムを含むパターンの抽出実験

コード進行データベースに小節線アイテムを付加して抽出した頻出パターンにはどのような音楽的特徴が含まれているかについて分析した。ここでは、最小 support 値を 10 として、終止形となる下記の頻出パターンを抽出し、小節線を含まないパターンと含むパターンそれぞれの support 値を比較した。ただし、I' は転回同値を適用したときの I の親コードアイテムであり、その他のコードアイテムについても同様に表記する。

- 全終止 ($\langle V', I' \rangle$, $\langle V', I_m' \rangle$)
- 偽終止 ($\langle V', VI_m' \rangle$, $\langle V', III_m' \rangle$, $\langle V', II_m' \rangle$)
- 変終止 ($\langle IV', I' \rangle$)

なお、全終止は転回の有無によって不完全終止と完全終止に分けられるため、転回のないパターン $\langle V, I \rangle$, $\langle V, I_m \rangle$ の support 値も比較した。また、終止形における小節線の有無についても、作曲者ごとの特徴が得られるかどうかを確かめるため、作曲者アイテムとジャンルアイテムを含むパターンについても同様に検討を行った。

構造アイテムを含むパターンの抽出実験

コード進行データベースに楽曲の開始と終了を表す構造アイテムを付加して抽出した頻出パターンにはどのような音楽的特徴が含まれているかについて分析した。ここでは、正順走査において“START”から始まるパターンと、逆順走査において“END”から始まるパターンについて抽出した。

なお、ここではすべて親子関係を適用せずにパターンを抽出し、抽出するパターンの長さは2以上100以下とした。

7.2 アイテムへの親子関係の導入と出力パターンの比較

出力パターンの長さを2以上100以下として最小 support 値を変化させたときの出力パターン数を、基底パターンに着目して測定した結果を図7.1に示す。また、最小 support 値を変化させたときの出力基底パターン数を、親子関係の適用の有無によって、比較した結果を図7.2に示す。

図7.1において、新規に出力されるパターンは最小 support 値ごとに8割ほどを占めた。また、新規に出力されるパターン数は最小 support 値が低くなるほど多くなっていき、新基底パターンに由来するものが大部分を占めるようになることがわかった。一方、図7.2からは、基底パターンの数は親子関係の導入によって爆発的に増えるわけではないことがわかった。

この乖離の原因は、最小 support 値を低くするにつれて基底パターンの長さが増え、その分だけパターンに含まれるアイテムの親子の組み合わせ数が指数的に増加したためであると考えられる。1つの基底パターンのみについてパターン数が多く提示されてしまうことは、冗長なパターンが多く提示されるという意味で望ましいとはいえない。したがって、最小 support 値を低くする場合には、パターンの長さを制限するなどの工夫が必要となる。

そこで、出力パターンの長さを2以上4以下に制限して、同様に最小 support 値を変化させたときの出力パターン数を図7.3に、出力基底パターン数の比較を図7.4に示す。

図7.3では、パターン数の爆発は抑えられた。この場合も新規に出力されるパターン8割から9割と高い割合であるが、新基底パターンに由来する新規出力パターン数は1割ほどであった。また、図7.4より、基底パターンの数もさほど変化していないことから、全出力パターンのうち大半が既知の基底パターンの親コードアイテムと子コードアイテムの組み合わせによる派生パターンであった。

以上から、長さが長いパターンにおいて、既存手法で未発見のパターンを多数抽出することができたが、長さの短いパターンに対する影響は軽微なものであ

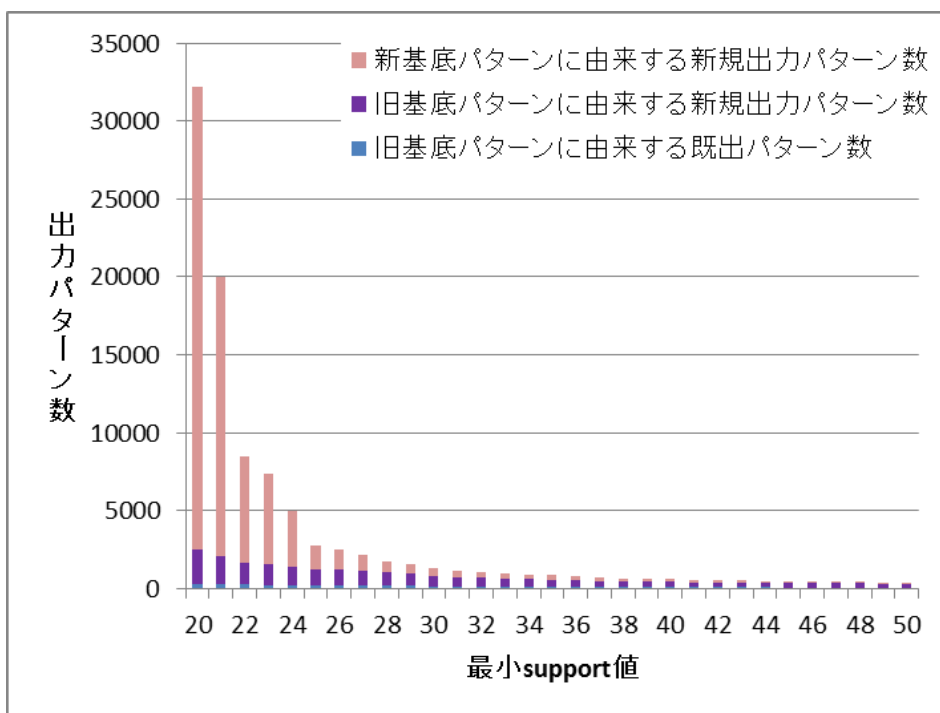


図 7.1: 最小 support 値ごとの出力パターン数とその分布 (長さ 2 以上 100 以下)

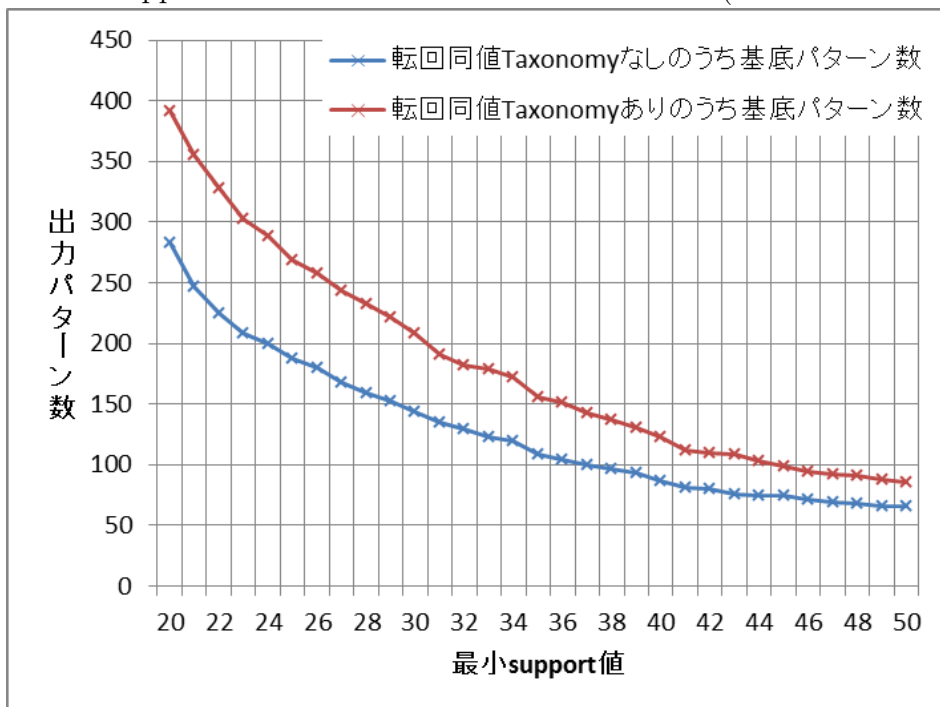


図 7.2: 最小 support 値ごとの出力基底パターン数とその分布 (長さ 2 以上 100 以下)

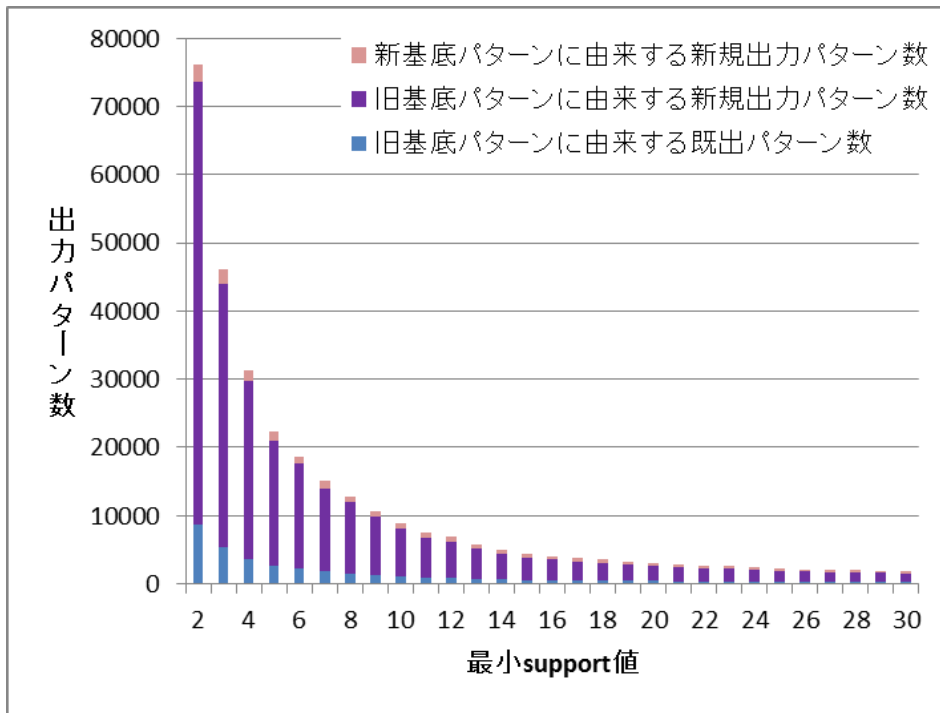


図 7.3: 最小 support 値ごとの出力パターン数とその分布 (長さ 2 以上 4 以下)

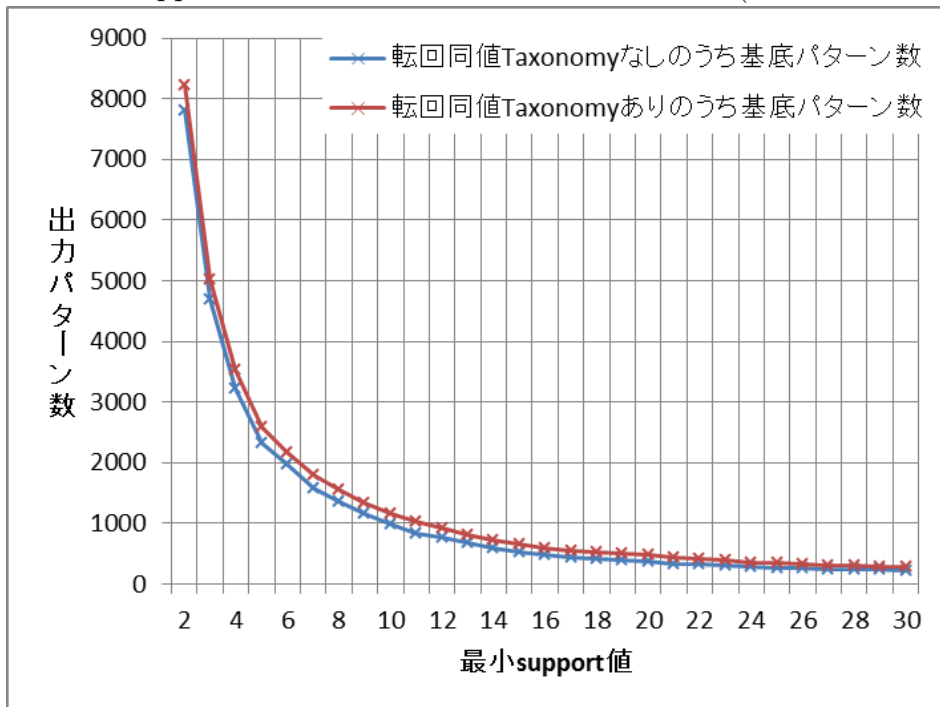


図 7.4: 最小 support 値ごとの出力基底パターン数とその分布 (長さ 2 以上 4 以下)

た。しかし、転回同値となる親子関係の導入によって、support 値が少なく抽出しにくいパターンや既存手法では発見できないパターンが高い最小 support 値で提示できるようになったことは有意義であると考えられる。そこで、コード進行の抽出・提示には基底パターンを特に重視して表示し、あてはまる転回形をユーザーに選ばせるインターフェースとすることで、ユーザーに混乱を与えることなく、既存手法で抽出されないパターンや抽出されにくかったパターンを利用できるようになると考えられる。

7.3 メタ情報アイテムを含むパターンの抽出実験

作曲家アイテム m_k を含むパターン $\langle m_k, //, I \rangle$ に対する後続のコード進行に対して、表 7.1 の左側に検定した結果を示す。有意水準 5% の χ^2 検定で有意差があると判定されたものは、生起確率に下線を引いて示す。また、最も正順 confidence 値の高かったパターンを表 7.1 の右側に示す。同様の形式でジャンルアイテムについて検討したものを表 7.2 に示す。なお、 $s = support(\langle m_k \rangle)$ である。

表 7.1, 表 7.2 の結果とも、すべてのパターンにおいて有意差が認められ、帰無仮説は棄却される。したがって、メタ情報アイテムを付加してマイニングを行うことにより、 $\langle I \rangle$ に対する後続のコード進行は作曲家やジャンルに特徴的なパターンであることが確かめられた。また、メタ情報アイテムのない通常のパターンに比べて、総じて confidence 値が高いことが確認できた。また作曲家アイテムの場合について、長さ 3 のパターンよりも、長さ 4 のパターンの方が生起確率が高いケースが見受けられた。この原因として、複雑なコードを挟んで単純なコードに進行する場合は該当するものと考えられる。

続いて、作曲家アイテム m_k を含むパターン $\langle I, //, m_k \rangle$ に終止するコード進行に対して、表 7.3 の左側に検定した結果の一部を示す。有意水準 5% の χ^2 検定で有意差があると判定されたものは、生起確率に下線を引いて示す。また、最も逆順 confidence 値の高かったパターンを表 7.3 の右側に示す。同様の形式でジャンルアイテムについて検討したものを表 7.4 に示す。なお、 $s = support(\langle m_k \rangle)$ である。

表 7.3 から、長さ 2 および長さ 3 のパターンにおいて有為差が認められないケースがあった。このとき、当該作曲者の分布は一般的なパターンの分布に近いということである。実際に、有意差がないと評価された作曲家アイテムを含むパターンにおいては、confidence 値の最も高いパターンはコード進行 $\langle V, I \rangle$ で一致しており、一般的に終止形が多用されていることが伺える。これは逆に、当該作曲者が一般的に使用されるコード進行を多用しているという特徴を表しているともできる。一方、有為差が認められるにも関わらず confidence 値の最も高い

表 7.1: $\langle I \rangle$ に対する後続のコード進行の有意差と confidence 値 (作曲者アイテムを含む場合)

m_k	s	χ^2 値	生起確率	コード進行	正順値
宇多田 ヒカル	4	408.72	8.25×10^{-42}	$\langle I, VI_m \rangle$	0.52941
		1085.30	4.09×10^{-89}	$\langle I, VI_m, V_6 \rangle$	0.52941
		1189.94	9.52×10^{-92}	$\langle I, VI_m, V_6, IV_{M7} \rangle$	0.52941
菅野 よう子	4	821.05	6.37×10^{-118}	$\langle I, I_{sus4} \rangle$	0.34247
		773.45	2.55×10^{-43}	$\langle I, I_{sus4}, I \rangle$	0.24658
		619.88	3.54×10^{-17}	$\langle I, V_{7sus4}/II, I/III, IV_{add9} \rangle$	0.10959
坂本 龍一	4	573.95	3.67×10^{-71}	$\langle I, III/^bVI \rangle$	0.23214
		1294.09	4.71×10^{-123}	$\langle I, III/^bVI, VI_{m7} \rangle$	0.23214
		1075.63	1.46×10^{-74}	$\langle I, III/^bVI, VI_{m7}, VI_{m7}/V \rangle$	0.23214
椎名 林檎	4	256.03	1.29×10^{-17}	$\langle I, III_m \rangle$	0.40000
		756.70	4.24×10^{-41}	$\langle I, III_m, VI_m \rangle$	0.40000
		639.97	3.86×10^{-19}	$\langle I, III_m, VI_m, V_{sus4} \rangle$	0.25000
平井 堅	4	299.00	5.57×10^{-24}	$\langle I, V/VII \rangle$	0.26471
		759.83	1.64×10^{-41}	$\langle I, V/VII, VI_{m7} \rangle$	0.11765
		601.60	1.85×10^{-15}	$\langle I, I_{M9}/III, IV, IV/V \rangle$	0.11765
アンジェラ ・アキ	3	279.40	5.08×10^{-21}	$\langle I, V \rangle$	0.28571
		653.20	3.99×10^{-28}	$\langle I, V, VI_m \rangle$	0.14286
		623.96	1.44×10^{-17}	$\langle I, V, IV, I \rangle$	0.09524
小室 哲哉	3	261.54	2.09×10^{-18}	$\langle I, VI_m \rangle$	0.48485
		806.53	2.50×10^{-55}	$\langle I, VI_m, V \rangle$	0.22727
		906.74	8.62×10^{-51}	$\langle I, VI_m, V, IV \rangle$	0.22727
つんく	3	490.78	4.11×10^{-56}	$\langle I, IV_{M7} \rangle$	0.33333
		1034.77	3.14×10^{-81}	$\langle I, IV_{M7}, V \rangle$	0.33333
		823.89	4.92×10^{-40}	$\langle I, IV_{M7}, V, III_{m7} \rangle$	0.33333
徳永 英明	3	203.76	1.28×10^{-10}	$\langle I, V/VII \rangle$	0.31818
		757.90	2.95×10^{-41}	$\langle I, I_{add9}, I \rangle$	0.10227
		657.95	5.87×10^{-21}	$\langle I, V/VII, VI_{m7}, VI_{m7}/V \rangle$	0.09091
福山 雅治	3	258.09	6.55×10^{-18}	$\langle I, V/VII \rangle$	0.21154
		609.06	4.67×10^{-23}	$\langle I, V_{sus4}, IV_{M9} \rangle$	0.15385
		735.24	2.27×10^{-29}	$\langle I, V_{sus4}, IV_{M9}, V_{sus4} \rangle$	0.15385
なし	-	-	-	$\langle I, V/VII \rangle$	0.12227
		-	-	$\langle I, V/VII, VI_{m7} \rangle$	0.04748
		-	-	$\langle I, ^bVII, IV, I \rangle$	0.02059

表 7.2: $\langle I \rangle$ に対する後続のコード進行の有意差と confidence 値 (ジャンルアイテムを含む場合)

m_k	s	χ^2 値	生起確率	コード進行	正順値
J.Pop	113	235.45	9.53×10^{-15}	$\langle I, V/VII \rangle$	0.14426
		426.17	2.63×10^{-6}	$\langle I, V/VII, VI_{m7} \rangle$	0.05713
		409.31	1.72×10^{-2}	$\langle I, V/VII, VI_{m7}, I/V \rangle$	0.01673
Rock	9	280.44	3.55×10^{-21}	$\langle I, V \rangle$	0.27309
		563.05	3.94×10^{-18}	$\langle I, V, I \rangle$	0.23214
		591.25	1.62×10^{-14}	$\langle I, V, IV, I/III \rangle$	0.06024
Pops (洋楽)	7	307.72	2.53×10^{-25}	$\langle I, IV \rangle$	0.14516
		629.91	2.06×10^{-25}	$\langle I, IV, I \rangle$	0.12903
		645.38	1.11×10^{-19}	$\langle I, VI_{m7}, IV, V \rangle$	0.06452
なし	-	-	-	$\langle I, V/VII \rangle$	0.12227
		-	-	$\langle I, V/VII, VI_{m7} \rangle$	0.04748
		-	-	$\langle I, {}^b VII, IV, I \rangle$	0.02059

パターンが $\langle V, I \rangle$ と同じであるものは、一般的な終止形を使用するが、他にも多くの終止形を使用しているということがいえる。

一方、表 7.4 からはほとんど全てのパターンで有意差が認められた。ただし、最も confidence 値の高いパターンは $\langle V, I \rangle$ や $\langle IV, V, I \rangle$ などであり、support 値の高いパターンはどのジャンルも似通っていることが予想される。また、J.Pop の長さ 4 のパターンのみ有意差が認められなかった。この原因については、音楽理論による明確な裏付けはないものの、潜在的によく使用されるコード進行であると考えることができる。

さらに、confidence 値が 1.00 となる作曲者アイテムを含むコード進行に対して、表 7.5 の左側に実際のコード進行、右側に検定の結果の一部を示す。有意水準 5% の χ^2 検定で有意差があると判定されたものは、生起確率に下線を引いて示す。なお、 $s = support(\langle m_k \rangle)$ である。

表 7.5 の結果からは、有為差が認められたケースは 8 つであった。これらは、コード進行に多くの選択肢がある中で、作曲者が好んで使用しているコード進行であると考えることができる。一方、有為差が認められないと判定されるパターンのうち、 $\chi^2 = 0.00$ となるもの (有効桁数上 0.00 となるものもある) は、作曲者を含むパターンと含まないパターンの分布が全く同じであることを意味する。これは、当該コード進行がデータベース D 中にもその作曲者の楽曲にしか存在せず、作曲

表 7.3: $\langle I \rangle$ に終止するコード進行の有意差と confidence 値 (作曲者アイテムを含む場合)

m_k	s	χ^2 値	生起確率	コード進行	逆順値
宇多田 ヒカル	4	353.14	1.10×10^{-38}	$\langle \text{II}_{m7}, I \rangle$	0.52941
		1118.94	2.51×10^{-121}	$\langle \text{VI}_m, \text{II}_{m7}, I \rangle$	0.52941
		1048.73	4.65×10^{-77}	$\langle \text{IV}, \text{VI}_m, \text{II}_{m7}, I \rangle$	0.52941
菅野 よう子	4	554.86	1.17×10^{-75}	$\langle \text{I}_{\text{sus4}}, I \rangle$	0.34247
		754.59	1.50×10^{-60}	$\langle I, \text{I}_{\text{sus4}}, I \rangle$	0.24658
		538.59	8.26×10^{-13}	$\langle \text{I}/\text{III}, \text{V}_{7\text{sus4}}/\text{II}, \text{V}_{7\text{sus4}}, I \rangle$	0.34247
坂本 龍一	4	85.09	1.39×10^{-1}	$\langle \text{V}, I \rangle$	0.73214
		399.15	6.50×10^{-13}	$\langle \text{IV}, \text{V}, I \rangle$	0.37500
		1065.13	1.55×10^{-79}	$\langle \text{VI}_{m7}/\text{V}, \text{V}, I \rangle$	0.37500
椎名 林檎	4	83.77	1.62×10^{-1}	$\langle \text{V}, I \rangle$	0.40000
		272.60	6.19×10^{-3}	$\langle \text{V}_{\text{sus4}}, \text{V}, I \rangle$	0.35000
		496.66	2.58×10^{-9}	$\langle \text{VI}_m, \text{V}_{\text{sus4}}, \text{V}, I \rangle$	0.30000
平井 堅	4	126.32	8.04×10^{-5}	$\langle \text{V}, I \rangle$	0.29412
		412.00	3.07×10^{-14}	$\langle \text{V}_{\text{sus4}}, \text{V}, I \rangle$	0.22059
		536.42	1.34×10^{-12}	$\langle \text{II}_{m7}, \text{V}_{\text{sus4}}, \text{V}, I \rangle$	0.11765
アンジェラ ・アキ	3	250.40	1.47×10^{-21}	$\langle \text{IV}, I \rangle$	0.32143
		575.52	2.95×10^{-34}	$\langle \text{III}_m, \text{IV}, I \rangle$	0.32143
		636.57	1.97×10^{-22}	$\langle \text{VI}_m, \text{III}_m, \text{IV}, I \rangle$	0.09524
小室 哲哉	3	52.11	9.63×10^{-1}	$\langle \text{V}, I \rangle$	0.66667
		220.26	4.26×10^{-1}	$\langle \text{IV}, \text{V}, I \rangle$	0.54545
		652.73	3.42×10^{-24}	$\langle \text{V}, \text{IV}, \text{V}, I \rangle$	0.24242
つんく	3	34.58	1.00	$\langle \text{V}, I \rangle$	1.00000
		412.45	2.76×10^{-14}	$\langle \text{IV}_{M7}, \text{V}, I \rangle$	0.66667
		1017.35	2.27×10^{-72}	$\langle \text{VI}_{m7}, \text{IV}_{M7}, \text{V}, I \rangle$	0.66667
徳永 英明	3	326.08	5.16×10^{-34}	$\langle \text{IV}, I \rangle$	0.29545
		643.11	8.92×10^{-44}	$\langle I, \text{I}_{\text{add9}}, I \rangle$	0.10227
		633.96	3.76×10^{-22}	$\langle \text{VI}_{m7}, \text{VI}_{m7}/\text{V}, \text{IV}, I \rangle$	0.09091
福山 雅治	3	164.88	3.03×10^{-9}	$\langle \text{V}, I \rangle$	0.28846
		407.29	9.51×10^{-14}	$\langle \text{IV}_{\text{add9}}, \text{V}, I \rangle$	0.17308
		657.80	9.40×10^{-25}	$\langle \text{V}_{\text{sus4}}, \text{IV}_{\text{add9}}, \text{V}, I \rangle$	0.28846
なし	-	-	-	$\langle \text{V}, I \rangle$	0.33908
		-	-	$\langle \text{IV}, \text{V}, I \rangle$	0.14286
		-	-	$\langle \text{VI}_m, \text{IV}, \text{V}, I \rangle$	0.02311

表 7.4: $\langle I \rangle$ に終止するコード進行の有意差と confidence 値 (ジャンルアイテムを含む場合)

m_k	s	χ^2 値	生起確率	コード進行	逆順値
J.Pop	113	94.56	3.86×10^{-2}	$\langle V, I \rangle$	0.34218
		295.70	3.02×10^{-4}	$\langle IV, V, I \rangle$	0.15061
		334.99	3.39×10^{-1}	$\langle III_m, IV, V, I \rangle$	0.02770
Rock	9	295.76	6.68×10^{-29}	$\langle V, I \rangle$	0.36948
		476.47	1.65×10^{-21}	$\langle IV, V, I \rangle$	0.12450
		511.91	1.58×10^{-10}	$\langle IV, I/III, II_{m7}, I \rangle$	0.06024
Pops (洋楽)	7	238.31	1.12×10^{-19}	$\langle V, I \rangle$	0.27419
		403.43	2.38×10^{-13}	$\langle IV, V, I \rangle$	0.14516
		559.17	1.19×10^{-14}	$\langle V_6, IV, V, I \rangle$	0.06452
なし	-	-	-	$\langle V, I \rangle$	0.33908
		-	-	$\langle IV, V, I \rangle$	0.14286
		-	-	$\langle VI_m, IV, V, I \rangle$	0.02311

者アイテムを含まないパターンと差別化できないケースである。現状では、このようなケースは作曲者に特徴的であると断定はできないが、データベース D の楽曲数の増加に伴って進行の仕方が多岐にわたった場合、適切に作曲者の特徴を示すパターンになることも考えられる。それでもなお、楽曲数を増加させても進行の仕方があまり変わらないものについては、もともと進行の仕方がごく限られており、作曲者に特徴的とはいえないと考えられる。

以上から、メタ情報アイテムを含むパターンは概ねその特徴を表しており、confidence 値も高い傾向にあるが、その逆は必ずしも成り立たず、進行の仕方が多岐にわたるかどうか重要な要素であると考えられる。今回は $s = 4$ または $s = 3$ の場合の一例のみを示したが、データベース中の同じ作曲者の楽曲が少なすぎている可能性も考えられるため、今後データベースの量を増加させていく場合には慎重に検討する必要があるといえる。

表 7.5: confidence 値が 1.00 となる作曲者アイテムを含むコード進行とその有意差

m_k	s	P	χ^2 値	生起確率
宇多田 ヒカル	4	$\langle {}^b\text{VII}_{\text{aug}}, {}^b\text{VII}_6 \rangle$	0.00	1.00
		$\langle \text{IV}, \text{VI}_m, \text{II}_{m7} \rangle$	691.55	3.67×10^{-37}
		$\langle \text{IV}, \text{VI}_m, \text{II}_{m7}, \text{I} \rangle$	723.50	1.07×10^{-18}
菅野 よう子	4	$\langle {}^b\text{III}, {}^b\text{VII}_{\text{sus4}} \rangle$	163.77	8.78×10^{-23}
		$\langle {}^b\text{III}, {}^b\text{VII}_{\text{sus4}}, {}^b\text{VII} \rangle$	173.38	1.92×10^{-14}
		$\langle {}^b\text{III}, {}^b\text{VII}_{\text{sus4}}, {}^b\text{VII}, \text{V}_{m7} \rangle$	173.38	2.27×10^{-11}
坂本 龍一	4	$\langle \text{VI}_{m7}, \text{VI}_{m7}/\text{V} \rangle$	193.33	1.97×10^{-18}
		$\langle \text{III}_{m7}/{}^b\text{VI}, \text{VI}_{m7}, \text{VI}_{m7}/\text{V} \rangle$	44.43	7.08×10^{-2}
		$\langle {}^b\text{V}_{\text{hdim7}}, \text{IV}_{M7}, \text{I}_{M7}, \text{I}_7 \rangle$	44.20	1.37×10^{-1}
椎名 林檎	4	$\langle \text{IV}_{m7}^{(11)}, \text{V}_{m7} \rangle$	0.98	1.00
		$\langle \text{IV}_{m7}^{(11)}, \text{V}_{m7}, \text{I}_{m9} \rangle$	0.98	1.00
		$\langle \text{IV}_{m6}, \text{III}_{m7}, {}^b\text{III}_{\text{dim7}}, \text{II}_{m7} \rangle$	13.13	9.75×10^{-1}
平井 堅	4	$\langle \text{I}_{M9}/\text{III}, \text{IV} \rangle$	0.00	1.00
		$\langle {}^b\text{VI}_{m7}, \text{V}_{m7}, \text{I}_7 \rangle$	3.27	1.00
		$\langle {}^b\text{VI}_{\text{aug}}, \text{VI}_m/\text{V}, {}^b\text{V}_{\text{hdim7}}, \text{II}_{m9} \rangle$	5.12	9.99×10^{-1}
アンジェラ・アキ	3	$\langle \text{I}_{\text{add9}}/\text{III}, \text{IV}_{M9} \rangle$	10.85	2.86×10^{-1}
		$\langle \text{II}, \text{IV}, \text{I} \rangle$	40.30	9.91×10^{-2}
		$\langle \text{I}_{M7}/\text{VII}, \text{VI}_{m7}, \text{I}/\text{V}, \text{IV}_{\text{add9}} \rangle$	0.00	1.00
小室 哲哉	3	$\langle \text{III}/{}^b\text{VI}, \text{VI}_m \rangle$	13.16	1.07×10^{-1}
		$\langle \text{III}_7, \text{VI}_m, \text{VI}_{m7}/\text{V} \rangle$	63.43	1.33×10^{-1}
		$\langle \text{III}_7, \text{VI}_m, \text{VI}_{m7}/\text{V}, \text{IV}_{M7} \rangle$	84.12	7.70×10^{-2}
つんく	3	$\langle \text{III}_{m7}, \text{VI}_m \rangle$	33.51	3.94×10^{-1}
		$\langle \text{III}_{m7}, \text{VI}_m, \text{IV}_{M7} \rangle$	309.88	4.53×10^{-23}
		$\langle \text{III}_{m7}, \text{VI}_m, \text{IV}_{M7}, \text{V} \rangle$	309.88	5.79×10^{-16}
徳永 英明	3	$\langle \text{V}_6, \text{IV} \rangle$	8.34	7.58×10^{-1}
		$\langle \text{II}/{}^b\text{V}, \text{V}, \text{IV}/\text{V} \rangle$	21.90	9.83×10^{-1}
		$\langle \text{II}/{}^b\text{V}, \text{V}, \text{IV}/\text{V}, \text{V} \rangle$	21.90	7.43×10^{-1}
福山 雅治	3	$\langle \text{I}_7, \text{IV}_{M7} \rangle$	9.17	9.81×10^{-1}
		$\langle \text{I}_7, \text{IV}_{M7}, \text{V} \rangle$	68.94	3.14×10^{-1}
		$\langle \text{III}_7/{}^b\text{VI}, \text{VI}_m, \text{II}_{m7}, \text{II}_{m7}/\text{V} \rangle$	24.18	9.48×10^{-1}

7.4 小節線アイテムを含むパターンの抽出実験

最小 support 値を 10 と設定した場合の出力パターンとその support 値を抜粋して示す。全終止 $\langle V', I' \rangle$ について調べた結果を表 7.6, $\langle V', I_m' \rangle$ について調べた結果を表 7.7, 偽終止 $\langle V', VI_m' \rangle$ について調べた結果を表 7.8, $\langle V', III_m' \rangle$, $\langle V', II_m' \rangle$ について調べた結果を表 7.9, 変終止 $\langle IV', I' \rangle$ について調べた結果を表 7.10 にそれぞれ示す。

全終止については、小節線アイテムを 1 つ含むパターンが最も高い support 値を持ち、次いで小節線アイテムのないパターン、小節線アイテムを 2 つ以上含むパターンの順に高くなっていることがわかった。一般に、強拍で終わる終止を男性終止、強拍以外で終わる終止を女性終止と呼び、男性終止の方が終止感が強い。提案手法によって抽出されたパターンは、ポピュラー音楽では女性終止よりも、男性終止が数多く使用されているという特徴を表しているといえる。また、support 値の高さがこのような順番になった理由として、小節線アイテムを 2 つ以上含むパターンの使われ方が考えられる。これらは、終止感を先送りにするという意味で音楽的な「タメ」を表しており、楽曲の印象を大きく左右する。そのため男性終止の仲間であるが、女性終止よりも出現回数が少なくなったと考えられる。

ただし、作曲者アイテムを含む頻出パターンをみると、一部の作曲者は女性終止の方が頻出パターンとして抽出されている。こうしたケースでは、彼らが女性終止を好んで使用しているという特徴がうかがえる。

ジャンルに関してしてみると、J.Pop では女性終止がよく使われており、終止形を最も幅広く使い分けられていると考えられる。一方、Rock については、その大半が完全終止であったり男性終止であったりすることから、常に分かりやすい進行を重視して楽曲が作られていることがうかがえる。

さらに、全終止のパターンのうち、大半が転回形のない終止形であることがわかった。転回形でない全終止は完全終止と呼ばれ、全終止の中でも最も終止感の強い進行である。この結果から、クラシック音楽だけでなく、ポピュラー音楽において完全終止が重要な位置を占めているということがわかった。ただし、例外として椎名林檎の楽曲では完全終止が全く使用されていないことがわかった。これは椎名林檎の楽曲に特徴的なコード進行であると考えられる。

偽終止についても、小節線アイテムを含むパターンの support 値の方が多く、今回の実験では各作曲者・ジャンルとも小節線を含むパターンが頻出なものとして抽出された。偽終止はトニックの代理コードに進行するものが多いため、全終止と似た結果が得られたと考えられる。

変終止は、全体としては小節線アイテムを含まないものの方が多いという結果になった。これは変終止が十分な終止感を得る目的で使用されるものではないとい

う習慣を裏付ける結果であるといえる。したがって、強拍にトニックコード I を置くことによる安定感に変終止においてあまり関連性がないものと考えられる。実際に各作曲者やジャンルごとに小節線が頻出であるか否かは分かれるという結果になった。

今回は三和音での終止形のみを抽出し検討を行ったが、四和音を含む終止形もコードの役割はほぼ同じであるため、今回の結果に近いものであると予想される。ただし、抽出できるパターン数が増えるため、さらにバリエーションに富む新しい知見を得られる可能性はある。

表 7.6: 最小 support 値 10 を満たす全終止 $\langle V', I' \rangle$ のパターンと support 値

	$\langle V', I' \rangle$:194 $\langle V, I \rangle$:169	$\langle V', \text{bar}, I' \rangle$:658 $\langle V, \text{bar}, I \rangle$:613	$\langle V', \text{bar}, \text{bar}, I' \rangle$:20 $\langle V, \text{bar}, \text{bar}, I \rangle$:17
John Lennon	なし	$\langle V', \text{bar}, I' \rangle$:67 $\langle V, \text{bar}, I \rangle$:67	なし
北川 悠仁	なし	$\langle V', \text{bar}, I' \rangle$:45 $\langle V, \text{bar}, I \rangle$:45	なし
小室 哲哉	$\langle V', I' \rangle$:42 $\langle V, I \rangle$:36	なし	なし
坂本 龍一	$\langle V', I' \rangle$:32 $\langle V, I \rangle$:32	なし	なし
藤巻 亮太	$\langle V', I' \rangle$:31 $\langle V, I \rangle$:29	なし	なし
草野 正宗	なし	$\langle V', \text{bar}, I' \rangle$:29 $\langle V, \text{bar}, I \rangle$:29	なし
小淵 健太郎	なし	$\langle V', \text{bar}, I' \rangle$:28 $\langle V, \text{bar}, I \rangle$:27	なし
織田 哲郎	なし	$\langle V', \text{bar}, I' \rangle$:27 $\langle V, \text{bar}, I \rangle$:27	なし
桑田 佳祐	なし	$\langle V', \text{bar}, I' \rangle$:25 $\langle V, \text{bar}, I \rangle$:25	なし
大塚 愛	なし	$\langle V', \text{bar}, I' \rangle$:24 $\langle V, \text{bar}, I \rangle$:20	なし
宮沢 和史	なし	$\langle V', \text{bar}, I' \rangle$:23 $\langle V, \text{bar}, I \rangle$:23	なし
Enya	なし	$\langle V', \text{bar}, I' \rangle$:21 $\langle V, \text{bar}, I \rangle$:21	なし
平井 堅	なし	$\langle V', \text{bar}, I' \rangle$:21 $\langle V, \text{bar}, I \rangle$:20	なし
J.Pop	$\langle V', I' \rangle$:172 $\langle V, I \rangle$:154	$\langle V', \text{bar}, I' \rangle$:457 $\langle V, \text{bar}, I \rangle$:426	$\langle V', \text{bar}, \text{bar}, I' \rangle$:12 $\langle V, \text{bar}, \text{bar}, I \rangle$:10
Rock	なし	$\langle V', \text{bar}, I' \rangle$:81 $\langle V, \text{bar}, I \rangle$:81	なし
Pops (洋楽)	なし	$\langle V', \text{bar}, I' \rangle$:39 $\langle V, \text{bar}, I \rangle$:32	なし

表 7.7: 最小 support 値 10 を満たす全終止 $\langle V', I_m' \rangle$ のパターンと support 値

	$\langle V', \text{bar}, I_m' \rangle:82$ $\langle V, \text{bar}, I_m \rangle:68$
松本 孝弘	$\langle V', \text{bar}, I_m' \rangle:30$ $\langle V, \text{bar}, I_m \rangle:30$
椎名 林檎	$\langle V', \text{bar}, I_m' \rangle:12$ $\langle V/\text{VII}, \text{bar}, I_m \rangle:12$
中島 みゆき	$\langle V', \text{bar}, I_m' \rangle:10$ $\langle V, \text{bar}, I_m \rangle:10$
J.Pop	$\langle V', \text{bar}, I_m' \rangle:42$ $\langle V, \text{bar}, I_m \rangle:30$
Rock	$\langle V', \text{bar}, I_m' \rangle:28$ $\langle V, \text{bar}, I_m \rangle:28$

表 7.8: 最小 support 値 10 を満たす偽終止 $\langle V', VI_m' \rangle$ のパターンと support 値

	$\langle V', VI_m' \rangle:25$	$\langle V', \text{bar}, VI_m' \rangle:264$
織田 哲郎	なし	$\langle V', \text{bar}, VI_m' \rangle:29$
松本 孝弘	なし	$\langle V', \text{bar}, VI_m' \rangle:28$
John Lennon	なし	$\langle V', \text{bar}, VI_m' \rangle:27$
小渕 健太郎	なし	$\langle V', \text{bar}, VI_m' \rangle:26$
草野 正宗	なし	$\langle V', \text{bar}, VI_m' \rangle:26$
北川 悠仁	なし	$\langle V', \text{bar}, VI_m' \rangle:13$
アンジェラ・アキ	なし	$\langle V', \text{bar}, VI_m' \rangle:13$
宮沢 和史	なし	$\langle V', \text{bar}, VI_m' \rangle:10$
J.Pop	なし	$\langle V', \text{bar}, VI_m' \rangle:208$
Rock	なし	$\langle V', \text{bar}, VI_m' \rangle:32$

表 7.9: 最小 support 値 10 を満たす偽終止 $\langle V', III_m' \rangle$, $\langle V', II_m' \rangle$ のパターンと support 値

	$\langle V', \text{bar}, III_m' \rangle:17$	$\langle V', \text{bar}, II_m' \rangle:14$
J.Pop	$\langle V', \text{bar}, III_m' \rangle:10$	なし

表 7.10: 最小 support 値 10 を満たす変終止のパターンと support 値

	$\langle IV', I' \rangle:290$	$\langle IV', \text{bar}, I' \rangle:208$	$\langle IV', \text{bar}, \text{bar}, I' \rangle:10$
Enya	なし	$\langle IV', \text{bar}, I' \rangle:43$	なし
BEGIN	$\langle IV', I' \rangle:34$	なし	なし
John Lennon	$\langle IV', I' \rangle:33$	なし	なし
アンジェラ・アキ	なし	$\langle IV', \text{bar}, I' \rangle:24$	なし
徳永 英明	$\langle IV', I' \rangle:23$	なし	なし
小淵 健太郎	$\langle IV', I' \rangle:22$	なし	なし
大塚 愛	なし	$\langle IV', \text{bar}, I' \rangle:21$	なし
松本 孝弘	$\langle IV', I' \rangle:14$	なし	なし
Richard Carpenter	なし	$\langle IV', \text{bar}, I' \rangle:14$	なし
井上 陽水	$\langle IV', I' \rangle:13$	なし	なし
J.Pop	$\langle IV', I' \rangle:214$	$\langle IV', \text{bar}, I' \rangle:126$	なし
Rock	$\langle IV', I' \rangle:47$	なし	なし
Pops (洋楽)	なし	$\langle IV', \text{bar}, I' \rangle:21$	なし

7.5 構造アイテムを含むパターンの抽出実験

構造アイテム“START”に対する正順走査結果と，“END”に対する逆順走査結果について，代表的な出力パターンとその support 値をそれぞれ表 7.11 と表 7.12 に示す．表の右側が提案手法による抽出パターンであり，表の左側はこれに合わせて構造アイテムを含まない場合の抽出結果を併記したものである．

表 7.11: 楽曲の開始を表す構造アイテムを含むパターンと support 値

構造アイテムなし		構造アイテムあり	
出力パターン	support	出力パターン	support
$\langle I, V/VII \rangle$	230	$\langle \text{START}, //, I, V/VII \rangle$	7
$\langle \text{bar}, I, V/VII \rangle$	200	$\langle \text{START}, //, \text{bar}, I, V/VII \rangle$	7
$\langle \text{bar}, \text{bar}, I, V/VII \rangle$	15	$\langle \text{START}, //, \text{bar}, \text{bar}, I, V/VII \rangle$	5
$\langle I, \text{bar}, IV/I \rangle$	22	$\langle \text{START}, //, I, \text{bar}, IV/I \rangle$	3
$\langle I, \text{bar}, V \rangle$	135	$\langle \text{START}, //, I, \text{bar}, V \rangle$	3

表 7.12: 楽曲の終了を表す構造アイテムを含むパターンと support 値

構造アイテムなし		構造アイテムあり	
出力パターン	support	出力パターン	support
$\langle V, \text{bar}, I \rangle$	613	$\langle V, \text{bar}, I, //, \text{END} \rangle$	12
$\langle V, \text{bar}, I, \text{bar} \rangle$	302	$\langle V, \text{bar}, I, \text{bar}, //, \text{END} \rangle$	11
$\langle V, \text{bar}, I, \text{bar}, \text{bar} \rangle$	77	$\langle V, \text{bar}, I, \text{bar}, \text{bar}, //, \text{END} \rangle$	8
$\langle I_{\text{sus}4}, \text{bar}, I \rangle$	53	$\langle I_{\text{sus}4}, \text{bar}, I, //, \text{END} \rangle$	7
$\langle I_{\text{sus}4}, \text{bar}, I, \text{bar} \rangle$	22	$\langle I_{\text{sus}4}, \text{bar}, I, \text{bar}, //, \text{END} \rangle$	7
$\langle II_{m7}/V, \text{bar}, I \rangle$	65	$\langle II_{m7}/V, \text{bar}, I, //, \text{END} \rangle$	4
$\langle II_{m7}/V, \text{bar}, I, \text{bar} \rangle$	30	$\langle II_{m7}/V, \text{bar}, I, \text{bar}, //, \text{END} \rangle$	3

“START”と“END”は1楽曲につき高々1個のみの出現であり，support 値もさほど高くはない．しかし，そこで最も頻出だったコード進行は構造アイテムを含まないパターンにおいても高い support 値をもつ頻出なコード進行であった．また，その全ての開始と終了がトニックコード I であった．その理由として，調性音楽において，楽曲の開始時から調を感じられないコードを用いることは望ましくないためであると考えられる．また，古典的な楽典では楽曲の終了を全終止とすることを定めており，ポピュラー音楽においても，調性の中で最も収束感のある

終わり方をすることが踏襲しているからであると考えられる。すなわち、楽曲の開始と終了においては、最も頻度が高く、安定したコード進行が多用される傾向にあるということがわかった。

第8章 おわりに

本章では、第7章での実験を踏まえ、結果のまとめと今後の課題について述べる。

8.1 まとめ

本稿では、シーケンシャルパターンマイニングの拡張によって、特徴あるコード進行を抽出することができることを報告した。提案手法では、アイテムに親子関係を導入して転回形を含むコード進行を抽出しやすくし、作曲者や小節線などの疑似アイテムを付加して楽曲情報や楽曲構造を考慮したパターンを抽出できるように拡張した。

本提案手法を楽曲 150 曲のコード進行データに適用した実験を行ったところ、親子関係の導入によって、support 値が少なく抽出しにくいパターンや既存手法では発見できないパターンが高い最小 support 値で提示できるようになった。また、疑似アイテムとして楽曲のメタ情報や小節線、楽曲の開始・終了を表すアイテムを付加することにより、作曲者・ジャンルごとのコードの使用傾向を明らかにし、出現位置やコードの音価の特徴を抽出パターンとして反映することができた。

8.2 今後の課題

実験により挙げられた課題や今後の展望として、次のような事項が考えられる。

マイニングに用いるコード進行データの増加

本研究の前提となる問題点に、マイニングに用いるコード進行データベースの量が少ないという問題がある。データベース量を増やす手法として、単体で楽曲を構成している MIDI データからコードを抽出する手法が考えられる。ネット上には有償無償を問わず、MIDI データで制作された楽曲が数多く存在するが、本研究では YAMAHA 社が販売するメタ情報があらかじめ含まれた MIDI (XF) データからコード進行データベースを生成した。このようなメタ情報がない場合、MIDI データに対して、楽曲の調の推定・コードを構成するトラックの推定・ベースノートなどを手がかりにしたコードの音価の

推定など、生成にはさまざまな工夫が必要になる。その他のアプローチとして、音楽・音声データである wav ファイル等からコードを推定・認識する方法も考えられるが、本研究で扱っている四和音や五和音、転回形まで正確に判定することは難しい。以上の問題があると考えられるが、これを解決することにより、特定の作曲家やジャンルの楽曲が大幅に増え、マイニング結果もより確かなものにする可以考虑。

時間情報を用いたマイニングの検討

今回は楽曲のコード進行に小節線を付加することにより、各コードアイテムに一定の音価の概念を与えたが、今後の課題として、さらに厳密に音価を制御できるよう拡張する。その手法として、アイテムに継続時間や時間間隔を持たせ、これを抽出条件として適用する拡張法を検討する。これにより、同じコード進行でも音価の違うものを正確に区別することができる。

疑似アイテムの追加導入

今回導入した疑似アイテムの他にも、構造アイテムに MIDI データにおける楽曲のマーカを導入したり、拍子・調の情報や演奏楽器情報を取り入れることで、さらに楽曲情報や楽曲構造と関連性の高いコード進行の抽出が可能になると考える。

また、このような疑似アイテムを順次導入していくことにより、MIDI データや MusicXML といった楽譜情報全体をマイニングすることにも繋がり、抽出できる情報を大幅に拡大できると考えられる。

メロディーを対象としたマイニング

本研究の応用事項として、頻出なメロディーを提示したり、コード進行に合ったメロディーの提示を行うことが考えられる。

前者の場合は、メロディーを構成する音符をアイテムとし、疑似アイテムとの関連性などを調べたりすることで応用できると考えられる。ただし、コードアイテムを扱うよりも、音符間の隣接度合いは少ないため、ギャップによる制御がより一層重要になると考えられる。

後者の場合は、鳴っているコードに合わせ、和声内音からできるだけ逸脱のないようにする必要があり、コードに対応した確率モデルを利用してメロディーを生成するなどのインターフェースの導入を検討すべきである。また古典的な音楽理論では、和声法に対応するものとして対位法が確立されている。旋律を無から創り出すことそのものが困難であるならば、計算機によってランダムかつ対位法などの理論に即した旋律を生成するなどのシステムの

構築が考えられる。

いずれの解釈においても、本システムで用いた手法と同様に、マイニング手法によって導出したメロディーを提示するシステムを導入することは、本研究の発展として望ましいと考えられる。

アプリケーションのインターフェースの向上

本研究では、マイニング手法を用いて抽出されたコード進行を利用した作曲支援システムも並行して開発して行ってきた。しかし、マイニング手法の拡大に伴い、操作をより直感的に行えるよう改良する必要がある。特に、どのコードからも終止形にたどり着けるような探索を検討するなどのインターフェースの実装が望まれる。

また、提示されたコード進行が実際に演奏したい楽器で演奏できるかどうかについては考慮されていない。対策としては、コード進行データベースに演奏楽器情報を表す疑似アイテムを付加するなど、何からの方法でコード進行を区別する必要がある。ただし、実際の MIDI データにはどのコードをどの楽器で演奏するかという情報は含まれていないため、手動での分析・入力求められる。

以上の検討を総合的に踏まえ、より興味深い特徴の抽出・分析を進める。また、多彩なコード進行をより直感的かつ適切に構成できる作曲支援システムの構築を目指す。

参考文献

- [1] H. Kenmochi and H. Ohshita. Vocaloid–commercial singing synthesizer based on sample concatenation. *Proc. Interspeech 2007*, pp. 4011–4010, 2007.
- [2] Steinberg Media Technologies GmbH. Cubase 7, 2012. <http://japan.steinberg.net/jp/products/cubase/>.
- [3] 篠原透, 大野将樹, 沼尾雅之. 作曲支援のためのマイニングによる和声進行提示システムの提案. 情報処理学会第 75 回全国大会, pp. 2–285–2–286, 2013.
- [4] C. Roads. *The Computer Music Tutorial*. MIT press, 1996.
- [5] 深山寛, 中妻啓, 米林裕一郎, 酒向慎司, 西本卓也, 小野順貴, 嵯峨山茂樹. Orpheus: 歌詞の韻律に基づいた自動作曲システム. 情報処理学会研究報告, pp. 179–184, 2008.
- [6] 徳丸正孝, 村中徳明, 今西茂. 「記憶と忘却」を繰り返し発想し続けるシステム: 記憶パターン更新型カオスニューラルネットワークと作曲システムへの応用. 知能と情報: 日本知能情報ファジィ学会誌: journal of Japan Society for Fuzzy Theory and Intelligent Informatics, Vol. 19, No. 3, pp. 299–312, 2007.
- [7] 安藤大地, P. Dahlstedt, M. Nordahl, 伊庭斉志. 対話型 GP を用いたクラシック音楽のための作曲支援システム. 芸術科学会論文誌, Vol. 4, No. 2, pp. 77–86, 2005.
- [8] E. Cambouropoulos, T. Crawford, and C.S. Iliopoulos. Pattern processing in melodic sequences: Challenges, caveats and prospects. *Computers and the Humanities*, Vol. 35, No. 1, pp. 9–21, 2001.
- [9] 藤川純平, 喜田拓也. 頻出直列エピソードマイニング手法を用いた音楽データからの繰り返し部分抽出. *DEIM Forum 2012 F6-4*, 2012.
- [10] A. Jiménez, M. Molina-Solana, F. Berzal, and W. Fajardo. Mining transposed motifs in music. *Journal of Intelligent Information Systems*, Vol. 36, No. 1, pp. 99–115, 2011.

- [11] J.L. Koh and W. Yu. Efficient feature mining in music objects. In *Database and Expert Systems Applications*, pp. 221–231. Springer, 2001.
- [12] M. Ogihara and T. Li. N-gram chord profiles for composer style representation. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR) 2008-Session 5d-MIR Methods*, 2008.
- [13] Carlos Pérez-Sancho, David Rizo, and José M Inesta. Genre classification using chords and stochastic language models. *Connection science*, Vol. 21, No. 2-3, pp. 145–159, 2009.
- [14] Ricardo Scholz, Emmanuel Vincent, and Frédéric Bimbot. Robust modeling of musical chord sequences using probabilistic n-grams. In *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on*, pp. 53–56. IEEE, 2009.
- [15] 北川祐. ポピュラー音楽理論 改訂版. リットーミュージック, 2004.
- [16] R. Agrawal and R. Srikant. Mining sequential patterns. In *Data Engineering, 1995. Proceedings of the Eleventh International Conference on*, pp. 3–14. IEEE, 1995.
- [17] J. Pei, J. Han, B. Mortazavi-Asl, H. Pinto, Q. Chen, U. Dayal, and MC Hsu. PrefixSpan: Mining sequential patterns efficiently by prefix-projected pattern growth. *ICDE'2001, April*, pp. 215–24, 2001.
- [18] Hajime Kitakami, Tomoki Kanbara, Yasuma Mori, Susumu Kuroki, and Yukiko Yamazaki. Modified prefixspan method for motif discovery in sequence databases. In *PRICAI 2002: Trends in Artificial Intelligence*, pp. 482–491. Springer, 2002.
- [19] Cláudia Antunes and Arlindo L Oliveira. Generalization of pattern-growth methods for sequential pattern mining with gap constraints. In *Machine Learning and Data Mining in Pattern Recognition*, pp. 239–251. Springer, 2003.
- [20] 浅原正幸. PrefixSpan-rel, 系列パターンマイニングツール, 2007. <http://prefixspanrel.sourceforge.jp/>.
- [21] ヤマハ株式会社. XF フォーマット仕様書 v2.03, 2000. <http://download.yamaha.com/file/45941>.

-
- [22] Issei Sato, Yu Hirate, and Hayato Yamana. Text mining using prefixspan constrained by item interval and item attribute. In *Data Engineering Workshops, 2006. Proceedings. 22nd International Conference on*, pp. 35–38. IEEE, 2006.
- [23] Ramakrishnan Srikant and Rakesh Agrawal. Mining generalized association rules. In *VLDB*, Vol. 95, pp. 407–419, 1995.
- [24] 篠原透, 沼尾雅之. シーケンシャルパターンマイニング拡張による特徴的なコード進行の抽出手法. 情報処理学会 音楽情報科学研究会 研究報告 2014-MUS-104-3, August 2014.
- [25] 篠原透, 沼尾雅之. シーケンシャルパターンマイニング拡張による特徴的なコード進行の抽出手法. *DEIM Forum 2015 F1-2*, 2015.

謝辞

本研究を進めるにあたり，データマイニングの手法や意義，研究の指針や評価方法など，全面的な支援をしてくださった指導教員の沼尾雅之教授に深謝いたします。また，日常の議論を通じて多くのご助言を頂いた沼尾研究室の皆様にご感謝いたします。

最後に，生活を支えてくださった家族にご感謝し，謝辞といたします。