

ポピュラー音楽クラスタリングのための近親調を用いた コード進行類似度の提案

長澤 槇子[†] 渡辺 知恵美[‡] 伊藤 貴之[‡] 増永 良文^{‡‡}

[†]お茶の水女子大学大学院 人間文化研究科 数理・情報科学専攻

[‡]お茶の水女子大学大学院 人間文化創成科学研究科基幹部門 自然・応用科学科

〒112-8610 東京都文京区大塚 2-1-1

^{‡‡}青山学院大学 情報科学研究センター

〒229-8558 神奈川県相模原市淵野辺 5-10-1

E-mail: [†]makiko@db.is.ocha.ac.jp, [‡]{chiemi, itot}@is.ocha.ac.jp, ^{‡‡}masunaga@irc.aoyama.ac.jp

あらまし 近年 MP3 プレーヤの流行により、多くの人がたくさん楽曲を聞く機会が増えてきている。楽曲には様々な音楽的特徴があり、楽器の種類、ハーモニーの変化、コード進行、リズムパターン、モチーフなどが挙げられる。そこで我々は、これらの音楽的特徴と作曲者、年代、ヒットチャートとの関係や、あるユーザの好みの楽曲の音楽的特徴の関係やユーザ同士の好みの楽曲の音楽的特徴の関係といったマイニングを行う。本稿では、以上に述べたことのマイニングを行うために、ポピュラー音楽を対象としたコード進行に着目した楽曲クラスタリングシステムを提案する。コード進行の類似度に関して、我々は音楽理論にて定義されている近親調をもとに定義した。

キーワード マルチメディア処理, マイニング, 音楽情報処理, クラスタリング, 近親調

Proposal on Chord Progression Similarity using Relative Key for Popular Music Clustering

Makiko NAGASAWA[†] Chiemi WATANABE[‡] Takayuki ITO[‡] and Yoshifumi MASUNAGA^{‡‡}

[†]Graduate Division of Mathematics and Computer Science (Master's Program), Ochanomizu University

[‡]Department of Natural and Advanced Science, Ochanomizu University

2-1-1 Otsuka, Bunkyo-ku, Tokyo, 112-8610 Japan

^{‡‡}Information Science Research Center, Aoyama Gakuin University

5-10-1 Fuchinobe, Sagamihara-shi, Kanagawa, 229-8558 Japan

E-mail: [†]makiko@db.is.ocha.ac.jp, [‡]{chiemi, itot}@is.ocha.ac.jp, ^{‡‡}masunaga@irc.aoyama.ac.jp

Abstract Recently we have more chances to listen to a lot of music due to the spread of MP3 players. There are various features in music, including musical instruments, harmony, chord progression, rhythm patterns, and motifs. Target of our study is mining of relativity among various features of music and their attributes, such as composers, date of release, ranking of hit charts, and among features of listeners' favorite music. In this paper we propose a music clustering system focusing on the chord progression of popular music for mining of the above relativity. We define similarity of chord progression according to relative keys defined by music theory.

Keyword Multimedia processing, Mining, Music information processing, Clustering, Relative key

1. はじめに

近年 MP3 プレーヤの流行により、多くの人がたくさん
の楽曲を聞く機会が増えてきている。楽曲には様々
な音楽的特徴がある。例えば以下のようなものが挙げ
られる。

- (1) 楽器の種類
- (2) ハーモニーの変化
- (3) コード進行
- (4) リズムパターン
- (5) モチーフ

(1)では、楽曲の楽器の種類を判別することが可能で
ある。(2)では、作曲者により特定のハーモニーパター
ンが頻出する場合がある。(3)では、楽曲の年代や歌手
により特定のコード進行パターンが頻出する場合があ
る。(4)では、音楽ジャンルによってリズムパターンが
決まっており、音楽ジャンルを特定することが可能で
ある。(5)のモチーフは、音楽では動機とも呼ばれてお
り、いくつかの音符ないしは、休符の特徴的な連なり
のことであり、楽曲の中で繰り返し用いられる。

本研究ではこれらの音楽的特徴と作曲者、年代、ヒ
ットチャートとの関係や、あるユーザの好みの楽曲の
音楽的特徴の関係やユーザ同士の好みの楽曲の音楽的
特徴の関係といったマイニングを行う。

本稿では、以上に述べたことのマイニングを行うた
めに、上に挙げた音楽的特徴の中の(3)のコード進行に
着目し、ポピュラー音楽を対象とした楽曲のクラスタ
リングを行う。コード進行とは楽曲の旋律に伴う和音
(コード)の変化のことである。コード進行が同一でリ
ズムパターンやメロディをアレンジすることだけで同一
の楽曲でも印象は異なったものになる。このことから、
音楽ではコード進行がベースとなっており、リズムパ
ターンやメロディに比べ不変性の高いものであるとい
える。よって、楽曲の特徴要素としてコード進行は重
要なものであると考え、コード進行に着目した。クラ
スタリングを行う際のコード進行の類似度に関して、
我々は音楽理論にて定義されている近親調をもとに定
義した。近親調に関しては、第 4.4.1 節にて述べる。本
稿では、コード進行変換についての定式化を重点に述
べる。

本システムが実現すれば、「作曲者がよく使うコード
進行パターン」や「異なる楽曲間で類似したコード進
行パターンを用いている」といった情報を抽出するこ
とが可能となる。また、クラスタリングの結果をもと
に、楽曲名、作曲者、発売日、その楽曲のヒットチャ
ートといった属性を追加し、「ヒット曲とコード進行と
の関係」や「作曲者とコード進行の関係」といった、
他属性との相関関係を抽出することが可能となる。

2. 関連研究

楽曲クラスタリングシステムに関する研究について、
松田ら[1]は、テキストマイニング技術を音楽情報へ適
用し音楽解析・音楽分類について提案した。音やコー
ドに適切なテキストを割り当て、メロディとコードに
ついての分析を行うというものである。梶[2]は、楽曲
に対する多様な解釈を Web 上から収集する音楽アノ
テーションシステムを構築し、楽曲検索システムとプレ
イリスト推薦システムを構築した。楽曲検索システム
では、タイトルや作詞・作曲者、アノテーションを用
いたキーワード検索、コード進行による検索、曲の構
成による絞込み検索が可能である。コード進行による
検索に関してはコードに数値を割り当て、各音楽コン
テンツに対してコードのアノテーションを取得し曲ご
とにコード進行を作成し、DP マッチングを行い検索要
求との距離を測るというものである。吉原ら[3]は、旋
律に対する和声付けに焦点を絞り、大量の雑多なデー
タから優位な情報を取り出し、取り出した情報を利用
した半自動和声付けシステムを構築した。その際、様々
なタイプの和声付けの実現のためにあらかじめ進行に
特徴のある楽曲同士のクラスタリングを行っている。

以上の論文では、コード進行の類似度を測る際に近
親調を考慮していないため、コード進行理論の観点か
ら類似したものが同じクラスタに含まれているかは明
確ではない。そこで我々は近親調を考慮しコード進行
に関しての距離を定義し、クラスタリングを行う。本
稿では、コード進行のみに着目しクラスタリングを行
ったが、今後、メロディやリズムパターンといった要
素についてもクラスタリングを行う予定である。

3. 楽曲クラスタリングの概要

本章では、近親調を用いた楽曲クラスタリングにお
けるクラスタリング処理の概要について述べる。全体
の処理の流れは以下の通りである。図 1 に処理全体の
流れを示す。

- (1) コード進行の数値化
クラスタリングを行うにあたり、コード進行デー
タを数値化する。数値化の手法については第 4.1
節にて述べる。
- (2) コード進行の正規化
コード進行の類似パターンを抽出するにあたり、
コード進行を正規化する必要がある。正規化につ
いては、第 4.2 節にて述べる。
- (3) 楽曲の分割
本システムでは、楽曲の部分要素をクラスタリン
グしていくため、楽曲を分割する必要がある。楽
曲の分割については、第 4.3 節にて述べる。
- (4) 距離の計算

分割した要素ごとの類似度を測るために、第4章で述べる定義に従って距離の計算を行う。

(5) 類似パターン抽出

手順(4)にて求めた距離が近いものを類似しているとみなし類似パターンを抽出する。

(6) クラスタリング

手順(5)にて抽出した類似パターンのクラスタリングを行う。第5.1節にて、クラスタリング手法について述べる。

(7) 他属性との相関抽出

クラスタリング結果をもとに、楽曲名、作曲者、発売日、その楽曲のヒットチャートといった属性を追加し、「ヒット曲とコード進行との関係」や、「作曲者とコード進行との関係」といった、他属性との相関関係をマイニングする。

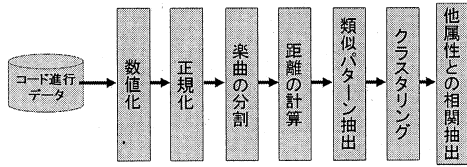


図1 楽曲クラスタリングの流れ

4. コード進行に関する諸定義

4.1. コード進行の数値化

コード進行一覧を図2に示す[4]、本システムでは、図中の第一行目の「C, C[#]・D^b, ..., B」をトーン、第一列目の「M, m, ..., mM7」をコードと名づける。トーンは図2で左から右に0, ..., 11, コードは図2で上から下に0, ..., 13と数値を割り当てる。また、音楽は時系列データであるので時間の情報を考慮する必要がある。本システムでは、四分音符を1拍として、一拍目を0とする。トーン、コード、時間の3つの要素で表されるものをコード進行と呼び、以下のように定義する。

【定義1】コード進行 Chord

Chord = (トーン, コード, 時間)

例えば、一拍目がD^bであった場合、D^b=(1, 0, 0)、四拍目がF_mであった場合、F_m=(5, 1, 3)となる。

	C	C [#] ・D ^b	D	D [#] ・E ^b	E	F	F [#] ・G ^b	G	G [#] ・A ^b	A	A [#] ・B ^b	B
M	C	C [#] ・D ^b	D	D [#] ・E ^b	E	F	F [#] ・G ^b	G	G [#] ・A ^b	A	A [#] ・B ^b	B
m	C _m	C [#] ・D ^b _m	D _m	D [#] ・E ^b _m	E _m	F _m	F [#] ・G ^b _m	G _m	G [#] ・A ^b _m	A _m	A [#] ・B ^b _m	B _m
7	C ₇	C [#] ・D ^b ₇	D ₇	D [#] ・E ^b ₇	E ₇	F ₇	F [#] ・G ^b ₇	G ₇	G [#] ・A ^b ₇	A ₇	A [#] ・B ^b ₇	B ₇
M7	C _{M7}	C [#] ・D ^b _{M7}	D _{M7}	D [#] ・E ^b _{M7}	E _{M7}	F _{M7}	F [#] ・G ^b _{M7}	G _{M7}	G [#] ・A ^b _{M7}	A _{M7}	A [#] ・B ^b _{M7}	B _{M7}
m7	C _{m7}	C [#] ・D ^b _{m7}	D _{m7}	D [#] ・E ^b _{m7}	E _{m7}	F _{m7}	F [#] ・G ^b _{m7}	G _{m7}	G [#] ・A ^b _{m7}	A _{m7}	A [#] ・B ^b _{m7}	B _{m7}
dim	C _{dim}	C [#] ・D ^b _{dim}	D _{dim}	D [#] ・E ^b _{dim}	E _{dim}	F _{dim}	F [#] ・G ^b _{dim}	G _{dim}	G [#] ・A ^b _{dim}	A _{dim}	A [#] ・B ^b _{dim}	B _{dim}
m7 ⁻⁹	C _{m7⁻⁹}	C [#] ・D ^b _{m7⁻⁹}	D _{m7⁻⁹}	D [#] ・E ^b _{m7⁻⁹}	E _{m7⁻⁹}	F _{m7⁻⁹}	F [#] ・G ^b _{m7⁻⁹}	G _{m7⁻⁹}	G [#] ・A ^b _{m7⁻⁹}	A _{m7⁻⁹}	A [#] ・B ^b _{m7⁻⁹}	B _{m7⁻⁹}
aug	C _{aug}	C [#] ・D ^b _{aug}	D _{aug}	D [#] ・E ^b _{aug}	E _{aug}	F _{aug}	F [#] ・G ^b _{aug}	G _{aug}	G [#] ・A ^b _{aug}	A _{aug}	A [#] ・B ^b _{aug}	B _{aug}
sus4	C _{sus4}	C [#] ・D ^b _{sus4}	D _{sus4}	D [#] ・E ^b _{sus4}	E _{sus4}	F _{sus4}	F [#] ・G ^b _{sus4}	G _{sus4}	G [#] ・A ^b _{sus4}	A _{sus4}	A [#] ・B ^b _{sus4}	B _{sus4}
7sus4	C _{7sus4}	C [#] ・D ^b _{7sus4}	D _{7sus4}	D [#] ・E ^b _{7sus4}	E _{7sus4}	F _{7sus4}	F [#] ・G ^b _{7sus4}	G _{7sus4}	G [#] ・A ^b _{7sus4}	A _{7sus4}	A [#] ・B ^b _{7sus4}	B _{7sus4}
6	C ₆	C [#] ・D ^b ₆	D ₆	D [#] ・E ^b ₆	E ₆	F ₆	F [#] ・G ^b ₆	G ₆	G [#] ・A ^b ₆	A ₆	A [#] ・B ^b ₆	B ₆
add9	C _{add9}	C [#] ・D ^b _{add9}	D _{add9}	D [#] ・E ^b _{add9}	E _{add9}	F _{add9}	F [#] ・G ^b _{add9}	G _{add9}	G [#] ・A ^b _{add9}	A _{add9}	A [#] ・B ^b _{add9}	B _{add9}
m6	C _{m6}	C [#] ・D ^b _{m6}	D _{m6}	D [#] ・E ^b _{m6}	E _{m6}	F _{m6}	F [#] ・G ^b _{m6}	G _{m6}	G [#] ・A ^b _{m6}	A _{m6}	A [#] ・B ^b _{m6}	B _{m6}
mM7	C _{mM7}	C [#] ・D ^b _{mM7}	D _{mM7}	D [#] ・E ^b _{mM7}	E _{mM7}	F _{mM7}	F [#] ・G ^b _{mM7}	G _{mM7}	G [#] ・A ^b _{mM7}	A _{mM7}	A [#] ・B ^b _{mM7}	B _{mM7}

図2 コード進行一覧

4.2. コード進行の正規化

あるBlock(定義2)同士のChordを比較した際に、Chordは違うが、Chordの遷移は同じである場合がある。このような場合にも、類似したコード進行パターンとみなして抽出も行うために、コード進行の正規化を行う。楽曲の調によって、Chordのトーンの値を変える。正規化の手法は以下の通りである。

- ・「ハ長調・イ短調」の場合：トーンのまま
- ・「ト長調・ホ短調」の場合：トーンの半音7個下
- ・「ニ長調・ロ短調」の場合：トーンの半音2個下
- ・「イ長調・嬰へ短調」の場合：トーンの半音9個下
- ・「ホ長調・嬰ハ短調」の場合：トーンの半音4個下
- ・「ロ長調・嬰ト短調」の場合：トーンの半音11個下
- ・「嬰へ(変ト)長調・嬰ニ(変ホ)短調」の場合：トーンの半音6個下
- ・「へ長調・ニ短調」の場合：トーンの半音5個下
- ・「変ロ長調・ト短調」の場合：トーンの半音10個下
- ・「変ホ長調・ハ短調」の場合：トーンの半音3個下
- ・「変イ長調・へ短調」の場合：トーンの半音8個下
- ・「変ニ長調・変ロ短調」の場合：トーンの半音1個下

図3にへ長調の楽曲の例を示す[4]。この楽曲は、へ長調であるので、正規化を行うとトーンは半音5個下のトーンとなり、コード進行は、「C, G, Am, Em, F, C, Dm7, G7」となり、「(0, 0, 0), (7, 0, 2), (9, 1, 4), (4, 1, 6), (5, 0, 8), (0, 0, 10), (2, 4, 12), (7, 2, 14)」と数値化される。



図3 へ長調の楽曲

4.3. 楽曲のブロック分け

音楽では一般に4小節、8小節といった2累乗の小節数が一つのまとまりとして区切りが良いと考えられるため、本システムでは楽曲を4小節ごとに分割し、分割した4小節をブロックと名づける。小節を分割し

た際に、曲の最後や1番から2番に切り替わるときなどに小節が余る場合がある。その場合、余った小節や区切りの悪い小節は切り捨てて考える。そこで、ブロックおよび楽曲を以下のように定義する。

【定義2】ブロック **Block**

Block は n 個のコード進行 $\text{Chord}_i (0 \leq i < n)$ によって以下のように構成される。

$$\text{Block} = (\text{Chord}_0, \text{Chord}_1, \dots, \text{Chord}_n)$$

例) $\text{Block}_0 = [(1, 0, 0), (3, 0, 4), (8, 0, 8), (5, 1, 12)]$

【定義3】楽曲 **M**

楽曲 **M** は n 個の **Block** によって以下のように構成される。

$$\text{M} = (\text{Block}_0, \dots, \text{Block}_n)$$

4.4. 類似度

4.4.1. 近親調

コード進行の類似度に関して、我々は音楽理論にて定義されている近親調(Relative Key)をもとに定義することとした。図4に示す五度圏[5]を参考に、近親調について説明する。近親調は以下6種類に分けることが出来る。

(1) 平行調

同一の調号をもつ長調と短調のことである。もとなる調が長調ならば、短3度下の短調、もとなる調が短調ならば、短3度上の長調が平行調の関係である。例えば、CとA_mが平行調の関係である(図4①)。平行調が最も距離が近い関係にあり、互いをもう一方の代理コードとして利用することもある。

(2) 属調

もとなる調の完全5度上の調のことである。例えば、Cの属調はGである(図4②)。属調の距離関係については、例えば、CとG、CとDの距離を比較した際に、CとDの距離の方が大きく差がでて遠くなり、CとB、CとF#の距離を比較した際には、あまり差はないという関係にある。つまり、五度圏上で変換先(定義4)のコード進行が変換元(定義4)から離れるに従い、距離の差はなくなるという関係にある。

(3) 下屬調

もとなる調の完全4度上の調のことである。例えば、Cの下屬調はFである(図4③)。下屬調の距離関係は属調の場合と同様である。

(4) 属調平行調

属調の平行調のことである。もとなる調が長調ならば、長3度上の短調、もとなる調が短調ならば、長2度下の長調が属調平行調の関係である。例えば、Cの属調平行調は、E_mである(図4④)。

(5) 下屬調平行調

下屬調の平行調のことである。もとなる調が長調ならば、長2度上の短調、もとなる調が短調ならば、長3度下の長調が下屬調平行調の関係である。例えば、Cの下屬調平行調は、D_mである(図4⑤)。

(6) 同主調

同一の主音を持つ長調と短調のことである。例えば、CとC_mが同主調の関係である(図4⑥)。同主調が最も距離が遠い関係にあり、同主音の長調と短調という対称的な関係であるため、他の近親調と比較しても圧倒的に「遠い」関係にある。

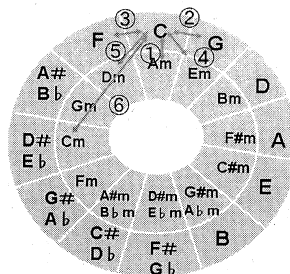


図4 五度圏

4.4.2. コードの類似度

コードは図2に示したように14種類あるが、「aug, 6, add9, sus4」は「M」から、「m7⁻⁵, m6, mM7」は「m」から、「7sus4」は「7」から派生したものと考えられる。「sus4」と「7sus4」に関して、これらはその次のコードが同音のマイナーである場合に限り、「sus4」は「m」、「7sus4」は「m7」から派生したものと考えられるが、このような用法は頻出しないので、上記のようにまとめる。また、「dim」、「m7⁻⁵」に関しては、「M」にも「m」にもまとめるににくい例外的なものなので、現時点では「M」にまとめる。以上のことより、「M」、「7」、「M7」、「m」、「m7」の5つのコードについての類似度を考える。

図5にコードの類似度の関係図を示す。コードは長調に使うコード同士、短調に使うコード同士は類似していると考えられる。よって、「M」、「7」、「M7」同士は類似度が高く、また、「m」、「m7」同士も類似度が高い関係にある。長調に使うコード同士をまとめたものを「メジャー群」、短調に使うコード同士をまとめたものを「マイナー群」と呼ぶ。「メジャー群」のコードと「マイナー群」のコードの類似度は低いため「遠い」関係にある。

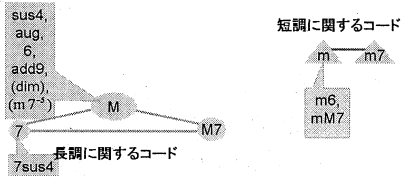


図5 コードの類似度

4.5. コード進行間の距離の定義

前節の近親調およびコード進行のコード間の関係をもとに、我々はコード進行間の距離に編集距離を利用することとした。編集距離とは、オブジェクト A にくつかの基本編集を適用し、オブジェクト B にした時に、その編集コストを距離とすることである。我々はコード進行の基本編集を定義するために「基本コード進行変換」を導入する。

【定義4】基本コード進行変換・基本変換コスト

コード進行 A をもう一方のコード進行 B に変換することを基本コード進行変換と呼び以下のように表す。また、コード進行 A を変換元、コード進行 B を変換先と呼ぶ。

$T(\text{type}, A \Rightarrow B)$

type は基本変換の種類であり、一回の変換で変換することを基本変換といい、調変換(定義6, tonetype)、コード変換(定義8, chordtype)がある。

例) C は T(平行調変換, $C \Rightarrow A_m$) で A_m に変換される。また、基本変換のコストは、

$\text{Cost}(T(\text{type}, A \Rightarrow B))$

と表す。コストは変換の種類によって決定される。

【定義5】コード進行間の距離

コード進行 A から B への変換は、コード進行 A に対して、n 回の調変換 $T_i(\text{tonetype}, a_i \Rightarrow a_{i+1}) (0 \leq i < n-1)$ と m 回のコード変換 $T_j(\text{chordtype}, a_j \Rightarrow a_{j+1}) (0 \leq j < m-1)$ によって行われる。また調変換の際コード A および B がメジャー群の場合はメジャー群の代表コードである M に、マイナー群の場合はマイナー群の代表である m に置き換え変換を行う。

このとき、コード進行 A, B 間の距離 $d(A, B)$ は調変換コストとコード変換コストの和で表され、以下ようになる。

$$\sum_{i=0}^{n-1} \text{Cost}(T(\text{tonetype}, a_i \Rightarrow a_{i+1})) + \sum_{j=0}^{m-1} \text{Cost}(T(\text{chordtype}, a_j \Rightarrow a_{j+1}))$$

例) C_7 と D_{m7} の場合、 C_7 調変換については C から D_m の変換コスト、コード変換については 7 から m7 の変換コストを計算し、距離は以下ようになる。

$$\text{Cost}(T(\text{tonetype}, C \Rightarrow D_m)) + \text{Cost}(T(\text{chordtype}, 7 \Rightarrow m7))$$

4.6. 基本編集

基本編集は、調変換とコード変換とで構成する。調変換は第 4.4.1 項をもとに定義する。まず、(3)下調は(2)属調の距離と同じものと考えられ、(4)属調平行調は属調と平行調、(5)下調平行調は下調と平行調を組み合わせたものである。また、同主調変換は平行調変換と属調変換で表すことができる。例えば、T(同主調, $C \Rightarrow C_m$) は、 $[T_0(\text{平行調変換}, C \Rightarrow A_m), T_1(\text{属調変換}, A_m \Rightarrow C_m)]$ となる。なお、平行調変換を属調変換と同主調変換で表すこともできるが、この場合、最も類似した平行調を最も「遠い」関係である同主調と属調で表すため、コスト加算による距離が成立しない。そのため、平行調を基本編集とする。

【定義6】調変換・調変換コスト

(6.1) 平行調変換

あるコード進行 A が「M」の場合、短3度下の「m」に変換し、「m」の場合、短3度上の「M」に変換する。平行調変換のコストは c とする。c は任意の数とする。

(6.2) 属調変換

あるコード進行 A の完全5度上のトーンに変換する。属調変換のコストは定義7の距離に従う。

例) A^b は $[T_0(\text{平行調変換}, A^b \Rightarrow F_m), T_1(\text{属調変換}, F_m \Rightarrow C_m)]$ で C_m に変換される。

【定義7】属調の距離

属調の距離に関しては、変換先が変換元から離れるに従って、距離の差はなくなっていくので式を以下のように設定する。また、属調か下調かによらずこれらの距離は対称に計算されなければならない。そこで、変換元から6つ進むと距離がマイナスに転じるように設定し、累積しても加算した値が相殺され、下調の計算が可能となるように定義した。コード進行 A と B の属調の距離は以下の通りである。

$$\sum_{i=0}^{a-1} (11 - 2i) \times b$$

ただし、b は任意の数、a は以下の式を満たす値である。

$$B = (A + 7a) \bmod 12 \quad (1 \leq a)$$

例) 図4の C から E^b の距離に関して、属調方向に9つ移動した距離は $27b$ 、下調方向に3つ移動した距離も $27b$ となり等しくなる。

【定義8】コード変換・コード変換コスト

(8.1) 初期変換

あるコード進行のコードが「aug」、「6」、「add9」、「sus4」、「dim」、「m7-5」ならば「M」に、「7sus4」ならば「7」に、「m6」、「mM7」ならば「m」に変換する。初期変換のコストは ϵ とする。

(8.2) Mm 変換

メジャー群のコードは M に、マイナー群のコードは m に変換する。 Mm 変換コストは以下のように定義する。

$$\cdot \text{Cost}(M \Leftrightarrow M7) = \text{Cost}(M \Leftrightarrow 7) = \alpha$$

$$\cdot \text{Cost}(7 \Leftrightarrow M7) = \beta$$

$$\cdot \text{Cost}(m \Leftrightarrow m7) = \gamma$$

また M と m の変換に関しては、平行調による変換で行う。コードの距離は、図 6 に示したパスを通じて、変換元から変換先へ到着する最短パスを求め、それぞれのパスの重みの合計値とする。 α, β, γ は任意の数とし変換にかかるコストの大小関係は、 $\varepsilon < \gamma < \alpha < \beta$ となる。図 6 にコード間の距離の図を示す。

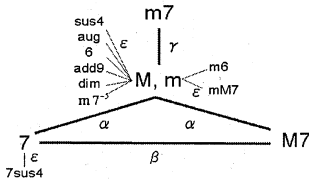


図 6 コード間の距離

4.7. 距離の計算例

第 4.5 節、第 4.6 節で述べた定義に従って、コード進行間の距離計算の具体例を示す。

① コードがメジャー群同士の場合

例)コード進行 C_7 から D_{M7} の変換は以下のように表される。

$$T(\text{tonetype}, C \Rightarrow D) = [T_0(\text{属調変換}, C \Rightarrow D)]$$

$$T(\text{chordtype}, 7 \Rightarrow M7) = [T_0(Mm \text{ 変換}, C_7 \Rightarrow C_{M7})]$$

このときにかかるコストは以下ようになる。

$$\text{Cost}(T(\text{tonetype}, C \Rightarrow D)) + \text{Cost}(T(\text{chordtype}, 7 \Rightarrow M7)) \\ = \text{Cost}(T(\text{属調変換}, C \Rightarrow D)) + \text{Cost}(7 \Leftrightarrow M7)$$

$$= \sum_{i=0}^1 (11 - 2i) \times b + \beta = 20b + \beta$$

② コードがメジャー群とマイナー群の場合

例)コード進行 C から $Fm6$ の変換は以下のように表される。

$$T(\text{tonetype}, C \Rightarrow F_m) = [T_0(\text{平行調変換}, C \Rightarrow A_m), T_1(\text{属調変換}, A_m \Rightarrow F_m)]$$

$$T(\text{chordtype}, M \Rightarrow m6) = [T_0(\text{初期変換}, F_{m6} \Rightarrow F_m)]$$

このときにかかるコストは以下ようになる。

$$\text{Cost}(T(\text{tonetype}, C \Rightarrow F_m)) + \text{Cost}(T(\text{chordtype}, M \Rightarrow m6)) \\ = \text{Cost}(T(\text{平行調変換}, C \Rightarrow A_m)) + \text{Cost}(T(\text{属調変換}, A_m \Rightarrow F_m))$$

$$+ \text{Cost}(m6 \Leftrightarrow m) = c + \sum_{i=0}^7 (11 - 2i) \times b + \varepsilon = c + 32b + \varepsilon$$

5. 比較予備実験

第 3 章、第 4 章で述べた手順・定義に従って楽曲クラスタリングシステムを構築し、J-POP の楽曲を対象に近親調を用いた場合と用いなかった場合の比較予備実験を行った。今回の予備実験では、3 人の作曲者の楽曲を 2 曲ずつ用意し、合計 6 曲、Block 数 77 個を対象にクラスタリングを行い比較予備実験を行った。扱ったデータに関しては、第 4.1 節で述べた定義に従って、楽譜を見ながら著者が手動で作成した。

5.1. クラスタリング手法

今回の予備実験では代表的な非階層的クラスタリング手法である k-means 法を用いた。以下にアルゴリズムを述べる。

- (1) K 個のクラスタの中心をランダムに設定する
- (2) それぞれの Block を最も近い中心に割り当てる
- (3) 全てのクラスタごとに中心を再定義
- (4) 全てのクラスタの中心が変化しなくなるまで、(2)に戻り繰り返す

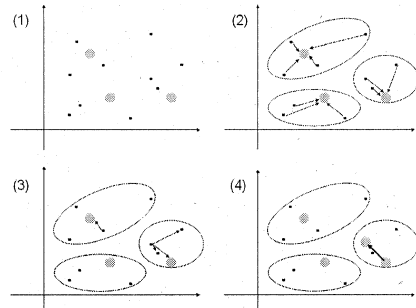


図 7 k-means 法 (k=3 の場合)

図 7 に k=3 の場合についての図を示す。

本システムにおいては、それぞれの Block 間の相対的な距離は分かるが、距離を求める際に編集距離を用いているため、平均値などは求められない。クラスタの中心を再定義する際に centroid は利用できないため、クラスタの中心を再定義する際に medoid を用いた。medoid とは、クラスタ内の最大距離が最小である点のことである。

5.2. 比較予備実験の考察

今回の予備実験ではクラスタの数を k=36 とし行った。近親調を考慮した場合と考慮しなかった場合のクラスタリング結果を図 8 に示す。図中の「距離の平均」は、それぞれのクラスタごとのクラスタの中心との距離の平均であり、「評価」はある一人の音楽に詳しい者による、類似したコード進行で構成されたクラスタか否かの主観による評価である。「○」は類似したもので

構成されたクラスタ、「×」は類似していないもので構成されたクラスタ、「△」どちらともいえない場合の評価、「-」はクラスタ内の要素が一つである場合の評価である。

近親調を考慮した場合のクラスタリング結果について、評価が「-」であるものを除いて、上位6件までのクラスタは、トーンが同じもので構成され、コードだけが異なるBlockで構成されていた。次の5件は、Block内に平行調や属調の関係にある要素が2、3個含まれるBlockで構成されていた。その他のものは、Block内の要素そのものは異なるもので構成されているが、音楽的観点からは類似しているBlockで構成されたクラスタが出来上がっているという結果になった。図9に近親調を考慮した場合のCluster3の結果を示す。図中「★」はクラスタの中心のBlockである。Cluster3は一見異なるコード進行が集まっているように見えるが、FとCは属調の関係、CとA_m、FとD_mは平行調の関係にあり近親調を考慮した場合に距離が近くなるBlockで構成されていることが分かる。なお、本クラスタ内のMusic1-Block14とMusic2-Block5は近親調を考慮しなかった場合は、Cluster0とCluster6というように別々のクラスタに含まれるという結果になっている。また、図10に近親調を考慮しなかった場合のCluster24の結果を示す。Cluster24のMusic1-Block11とMusic0-Block9について、CとD^b_{M7}は近親調を考慮しなければ、トーンが半音違うだけの関係にあり距離は近くなるが、実際にこの二つは音楽的意味の近いものではないので同じクラスタに含まれるべきではない。この二つのBlockは近親調を考慮した場合には、Cluster8とCluster17というように別々のクラスタに含まれ、音楽的観点から見て正しくクラスタリングされていた。

この比較予備実験により、近親調を考慮しない場合はクラスタの中心との平均距離が近いクラスタであっても、音楽的意味を逸脱したもので構成されたクラスタが出来上がり、類似していないコード進行同士でも

同じクラスタに含まれるという結果になった。よって、近親調を考慮した方が、音楽的観点から見て正しくクラスタリング出来る事が分かる。

また、評価の方法は今回のように音楽的観点からコード進行が類似しているか否かを評価する方法と、リズムやメロディや歌詞を評価基準として類似しているか否かを判断する方法があり人により判断基準は異なる。今回の予備実験の評価は音楽に詳しい者の一人の主観に基づくものであったが、今後、様々な評価基準から複数名の評価が必要であると考えられる。

Cluster No	距離の平均	評価
0	0	-
2	0	-
5	0	-
8	0	-
14	0	-
16	0	-
18	0	-
19	0	-
22	0	-
25	0	-
26	0	-
27	0	-
28	0	-
33	0	-
34	0	-
35	0	○
32	4	○
15	7	○
11	20.2	○
4	27.5	○
20	38.4	○
9	102.6	○
10	106	○
1	130.2	○
24	137.6	○
31	166	○
23	210.2	○
3	232	○
21	284.1	○
30	351.7	△
6	377.5	△
29	412	△
17	448.0	△
12	582.7	△
13	700.4	△
7	887	△

Cluster No	距離の平均	評価
5	0	-
10	0	-
11	0	-
16	0	-
22	0	-
25	0	-
28	0	-
31	0	-
34	0	-
35	0	-
21	4.8	○
18	13.6	×
9	17.2	○
23	17.6	○
3	19.5	×
24	20.4	×
8	20.7	○
0	21.6	△
6	24.3	×
12	24.3	×
4	24.8	×
2	24.9	×
15	27.2	×
1	28.4	○
13	28.5	△
32	29.2	○
7	30.6	×
30	31.2	×
20	32.6	×
17	33.6	○
33	33.6	△
27	34.2	○
26	36	○
29	37.4	×
19	39.0	○
14	46.4	○

(a)近親調を考慮した場合

(b)近親調を考慮しない場合

図8 クラスタリング結果の評価

Cluster No	Music No	Block No	コード進行
3	1	14	F F A b 7 A b 7 C C D D F F F F G G G G
3	2	5	C C C C A m A m A m A m D m D m D m G G G G
3	0	10	D m 7 D m 7 D m 7 D m 7 C C C C F F F F E m E m E m
3	3	4	F F F F C C A m A m D m D m D m G G G G
★	3	3	F F F F C C A m A m D m 7 D m 7 D m 7 D m 7 G G G G

図9 近親調を考慮した場合のクラスタリング結果の例

Cluster No	Music No	Block No	コード進行
24	1	12	D b M 7 D b M 7 D b M 7 D b M 7 D 7 D 7 D 7 D 7 G 7 G 7 G 7 D 7 D 7 D 7
24	1	11	D b M 7 D b M 7 D b M 7 D b M 7 D m 7 D m 7 D m 7 D m 7 D m 7 D m 7 D m 7 D m 7 D m 7
24	1	5	F M 7 F M 7 F M 7 F M 7 E b 6 E b 6 E b 6 E b 6 D m 7 D m 7 D m 7 G G G G
24	1	3	F M 7 F M 7 F M 7 F M 7 E b 6 E b 6 E b 6 E b 6 D m 7 D m 7 D m 7 D m 7 G G F E m
24	0	4	E m E m A m A m G m 7 G m 7 C 7 C 7 F F C C D m 7 D m 7 D m 7
24	0	9	F F C C D m 7 D m 7 D m 7 D m 7 C B b B b A b F F G G
★	24	0	E m E m A m A m D m 7 D m 7 D m 7 D m 7 E m E m A m A m D m 7 D m 7 D m 7

図10 近親調を考慮しない場合のクラスタリング結果の例

6. クラスタリング結果の応用例

本章では、クラスタリング結果を利用し、様々な関係をマイニングすることやクラスタリングの結果の可視化について詳細に述べる。

6.1. 相関ルール抽出

クラスタリング結果に基づき、Cluster.No や Block.No と他属性との相関抽出を考える。他属性として考えられるものは、「楽曲名」、「歌手名」、「作曲者」、「発売日」、「ヒットチャート」などである。Cluster.No や Block.No とこれらの相関関係をマイニングし、コード進行と他属性との関係のマイニングを目指す。例えば、「ヒット曲とコード進行との関係」や「作曲者とコード進行の関係」のマイニングを考えている。

6.2. 可視化

可視化には平安京ビュー[6]を利用する。平安京ビューとは、階層型データを構成する葉ノードをアイコンで表示し、枝ノードを入れ子状の長方形の枠で表現するというものであり、階層型データを可視化するための手法である。図 11 は平安京ビューによる階層型データの可視化の例である。平安京ビューは階層構造の葉ノードを塗りつぶされたアイコンで、枝ノードを長方形の枠で表現し、長方形の入れ子構造で階層構造を表現する。平安京ビューは、階層構造をもつウェブサイトのアクセス傾向や計算機ネットワークの不正侵入検知情報などの可視化、原子力システムの計測情報や薬物実験情報の可視化など、多くの分野に適用が試みられている汎用的な可視化手法である。

我々は現在、この可視化手法を用いて楽曲のクラスタリング結果の可視化を試みている。一例として我々は、同一クラスタに所属するコード進行をアイコンで表現し、クラスタを長方形の枠で表現し、そして作曲者あるいは楽曲でアイコンを色分けする、という方法を試している。これによって、「ある特定の楽曲のみで構成されるクラスタ」や「ある特定の作曲者や歌手の楽曲のみで構成されたクラスタ」の可視化が可能となり、「同一の作曲者、楽曲、歌手の作品の中に見られるコード進行が、同一のクラスタに集中している」というような傾向を発見できると考えられる。

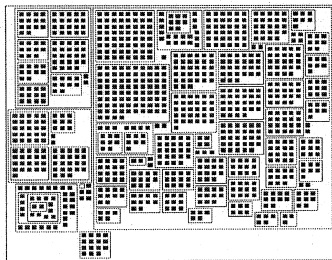


図 11 平安京ビューによる可視化の例

7. まとめと今後の課題

本稿では、楽曲中の音楽的特徴であるコード進行に着目し、ポピュラー音楽を対象とした楽曲のクラスタリングを行うために、コード進行の類似度に関する様々な定義、定式化を行った。また、定義に従って、k-means 法を用いたクラスタリングシステムを実装し、近親調を考慮した場合と、考慮しない場合の比較予備実験を行った。クラスタリング結果より、近親調を考慮した方が音楽的観点から類似しているもので構成されたクラスタが出来上がった。

今後は楽曲データ数をさらに増やし、楽曲のクラスタリングに最適なクラスタリング手法を用いて、より性質の良いクラスタリングを行う。クラスタリング手法には、階層的クラスタリングの適用を考えている。階層的クラスタリングは、クラスタの数を指定する必要はなく、距離が近いもの同士を同じクラスタに統合していくため、k-means 法よりも、より良い分類結果を得ることが期待できる。また、コード進行の数値化に関して、現時点では四分音符を 1 拍と定義しているが、今後 1 拍の定義に関して最適な長さについて検討していく。また、人間が耳にした際に似たコードであると感ずるかについて、多くの人に聞いてもらい実験を行う。

さらに、クラスタリング結果をもとに、作曲者、楽曲名、発売日、その楽曲のヒットチャートといった属性を追加し、「ヒット曲とコード進行の関係」、「年代とコード進行の関係」といった他属性との相関関係のマイニングやクラスタリング結果の可視化を行っていく予定である。また、コード進行以外のメロディやリズムパターンといった音楽的特徴についてもクラスタリングを行い、楽曲傾向をマイニングしていく。

文 献

- [1] 松田卓久, 飯島淳一, “テキストマイニング技術の音楽情報への適用”, 経営情報学会 2002 年春季全国研究発表大会予稿集, pp.186-189, 2002,
- [2] 梶克彦, “音楽におけるアノテーションとその応用に関する研究”, 名古屋大学大学院修士論文, 2004
- [3] 吉原一期, 近山隆, “Web 上の演奏データを利用した作曲支援システム”, 情報処理学会第 65 回全国大会, pp.2-197-2-198, 2003
- [4] 藤井英一, “藤井英一のコード進行講座”, 株式会社ヤマハミュージックメディア, 103p., 2006
- [5] 北川祐, “ポピュラー音楽理論”, リットーミュージック, 278p., 2004
- [6] 伊藤貴之, 山口裕美, 小山田耕二, “長方形の入れ子構造による階層型データ可視化手法の計算時間および画面占有面積の改善”, 可視化情報学会論文集, Vol.26, No.6, pp.51-61, 2006