

対話型遺伝アルゴリズムを用いた2旋律間の相性評価関数の生成

田島 雅斗 荒川 達也[†]
群馬工業高等専門学校 生産システム工学専攻[‡]

1 はじめに

本研究の目的は、2つの旋律の相性を定量的に評価する評価関数を作成することである。単体で聴くと良い旋律であっても、2つ同時に鳴らしたとき互いを妨害して良さを失うことは多い。一方、互いを引き立てあい、より高い音楽性を引き出すこともある。

このような旋律間の相性の良し悪しは聴者の感覚に依存する主観的なものであるが、実際には個人間で大きく評価が異なることは少なく、ある程度客観的な評価も可能だと考えられる。そこで本研究では、2つの旋律に対し、それらを同時に演奏したときの相性を評価値として数値で示す評価関数を提案し、遺伝アルゴリズム (GA) を用いて試作する。

現時点では人間の感覚の再現までは完成していないが、このような評価関数が実現できれば、自動作曲において自然な副旋律や和声付けを生成するための有効な方法になりうると考えられる。また将来的には、人工知能が音楽を理解するための一要素になると期待する。

2 先行研究との関係

コンピュータによる自動作曲の試みは多く、その中で複数の旋律間の関係も研究されてきた ([1] [2] など)。しかしこれらの手法は主に多声部を持つ楽曲を生成するための一つの工程として使われるため、必ずしも旋律間の相性を正確に評価することを目的としたものではない。

それに対し本研究では、2つの旋律の相性に対して人間の感性にできるだけ近い評価を行うこと自体を目的としている点でこれらの既存研究と異なる。なお将来的にはこれらの先行研究と連携することにより、本研究の提案手法は自動作曲のための有用な要素になると期待で

きる。

3 提案手法

2つの旋律の相性を評価するために、専用の評価関数 f を作成する。 f の入力は2つの旋律、出力は数値化された2旋律間の相性とする。評価値が大きいほど、同時に鳴らしたときにアンサンブル感に富む良い旋律の組み合わせであることを表す。一方負の値は2旋律が互いに邪魔しあう場合に対応する。基準として、入力された2旋律が同一の場合は評価値が0となるように設計する。具体的な実装は次節に示す。

4 評価関数の作成と考察

4.1 概要

まず評価関数の大まかな形を定め、その後旋律サンプルを用意し、対話型遺伝アルゴリズム (IGA) を用いて評価関数内のパラメータを決定する。パラメータを決定後、サンプルとは異なる旋律を入力してその評価値を計算し、人間の評価値と比較して有効性を検証する。

4.2 旋律の表現および評価関数の構造

旋律のモデル

旋律を表すデータ構造は先行研究でいくつか考えられているが、今回はベクトルで表現する方法を採用する。

具体的には小節を強制的に8等分し、音符(音高)・タイ(前の音符が続いていることを表す)・休符の3要素を旋律の先頭から順に並べることにより、旋律をベクトルとして表現する。音高はMIDIのノートナンバーを採用し、タイや休符の場合は例外的な値を設定する。

今回のサンプルでは8小節の旋律を8分音符で区切った次元数 $N = 64$ のベクトルとする。

評価関数の定義

評価関数はさまざまな形を考えることができるが、今回は試験的に以下のように2要素間の評価値への影響度

Evaluation Function of the Compatibility between Two Melodies with Interactive Genetic Algorithm

[†] Masato Tajima and Tatsuya Arakawa

[‡] National Institute of Technology, Gunma College: Advanced Production Systems Engineering Course

をパラメータとする形について実験を行った。

$$f(\mathbf{a}, \mathbf{b}) = \sum_{i,j=1}^N c_{ij} |a_i - b_j| \quad (1)$$

ここで \mathbf{a}, \mathbf{b} はそれぞれベクトル表現された旋律である。 c_{ij} は重みパラメータであり、この値により評価関数の値が決定される。なお、 $-1 \leq c_{ij} \leq 1$ とする。

a_i や b_j が休符の場合、その項は演算から除外する。また、 a_i や b_j がタイの場合、伸ばされている音符の音高を参照する。

4.3 対話型遺伝アルゴリズム

式 (1) の c_{ij} を定めるため、今回は実験的に対話型遺伝アルゴリズムを用いる。式 (1) においてパラメータ値 c_{ij} を任意に設定したものを個体とし、 c_{ij} を格納した長さ 64 の配列を遺伝子とする。初期世代はランダムに生成し、その後はいくつかのサンプル旋律に対して聴者（人間）による評価との誤差が小さいものを選択することにより世代交代を行う。今回行った評価実験での詳しい設定は次節で述べる。

4.4 評価実験

4.1-4.3 節で述べた提案手法の有効性を検証するために、python 3 で自作した簡易的な GUI アプリケーションを用いて評価実験を行なった。

- サンプル音源として 8 種類の旋律を自作した。
この中から 2 つ選んで式 (1) の入力 \mathbf{a}, \mathbf{b} とする。なお、これらのサンプル旋律のコード進行はすべて同一とした (I→V→VIIm→IV)。
- 1 世代は 16 個体とし、各サンプル音源に対する人間の評価値との 2 乗誤差の和が小さい 4 個体を選択して次の親とする。選ばれた 4 個体とそれらから定点点交叉（総当たり）によって生成される 12 個体の合計 16 個体を次の世代とする。
- 突然変異として、世代交代時に低確率でパラメータ値をランダムに変化させる。
- 2 乗誤差の和が 100 以下となる個体が 4 個以上見つかったら収束と判定する。

4.5 評価実験の結果および考察

実験はサンプル音源を変えて 2 種類行った。

[A] 8 種類のサンプル旋律のうち 4 種類を GA に用い、残り 4 種類を生成された評価関数のテストに使用した場合の実験結果を図 1 に示す。[B] f の一つめの入力 \mathbf{a} を固定し、2 つめの入力 \mathbf{b} のみを変える設定とした場合

の試行結果の一つを図 2 に示す（収束直前を拡大して示す）。実線は GA に用いた旋律に対する 2 乗誤差の和を、点線はテスト用の旋律に対する 2 乗誤差の和を示す。

[A] では、世代が進むにつれて GA に用いた旋律の組に対しては人間に近い評価値を出力したが、テスト用の旋律に対しては人間の評価値に近づくことはなかった。[B] では、テスト用の旋律に対しても人間に近い評価を出力する世代が見られた。同条件の試行において同様の傾向を示す場合がいくつか見られたが、毎回同様の結果が再現されるわけではなかった。図 2 は過学習に似た挙動を示しているが、[A] の結果や評価実験の設定を考慮すると偶然の結果だと考えられる。

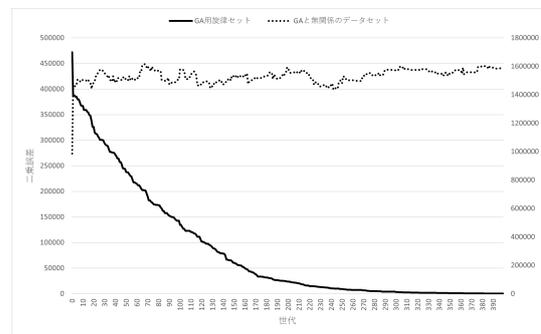


図 1 4 種類の旋律を GA に入力した場合

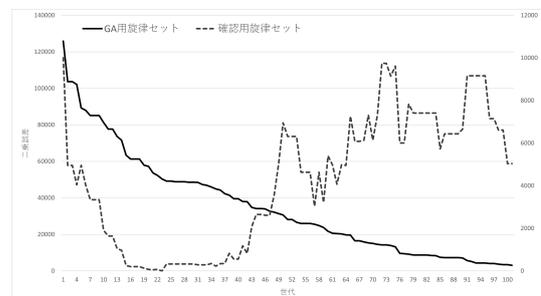


図 2 GA 入力の片方を固定とした場合

5 まとめと今後の予定

今回使用したモデルでは多様な旋律に対し人間に近い値を得るのは難しいとわかった。今後は人間の評価手法を考慮し、式 (1) の形を見直す予定である。

参考文献

- [1] 田中翼, 古川聖, 多声音楽の旋律スタイルの文法的なモデル化および自動生成, 情報処理学会論文誌, Vol.54, No.4, pp1308-1318, 2013.
- [2] 梅本あずさ *et.al.*, 音楽理論と経験的知識を整合活用した作曲支援システム, 情報処理学会研究報告音楽情報科学 (MUS), Vol.1997, No.122(1997-MUS-023), pp57-62, 1997.