

予測変換を用いた初心者向け作曲支援システムの改善と評価

山下峻^{†1} 坂本大介^{†1} 小野哲雄^{†1}

概要: アマチュア作曲の普及を背景として作曲に興味を持つ人が増えている。しかし、音楽知識に乏しい初心者にとって作曲は容易ではないため、初心者を対象とした様々な作曲支援システムが研究されてきた。我々は、入力メロディの先に続く候補メロディを提示することで、ユーザの作曲を支援するシステムを開発してきた。本研究では、システムの実用性向上を目的として、候補メロディの提示アルゴリズムの改善に取り組んだ。評価実験として、改善前後のシステムで生成したメロディの比較実験を行ったところ、改善後のメロディに対して評価得点が向上し、アルゴリズム改善の有効性を確認することが出来た。

Improvement and Evaluation of Composition Support System Using Prediction Conversion

SHUN YAMASHITA^{†1} DAISUKE SAKAMOTO^{†1} TETSUO ONO^{†1}

Abstract. With the popularization of amateur composition, the number of people who are interested in music composition is increasing. However, music composition is not easy for beginners who are poor in music knowledge, so various composition support systems for beginners have been studied. We have developed a system that supports user's composition by proposing candidate melodies that follows from the input melody. In this research, we worked on improving the proposal algorithm of candidate melodies with the aim to improve practicality of our system. As an evaluation, we conducted experiment to compare melodies generated in the system before and after improvement. As a result, the evaluation score was improved for melodies generated by the system after improvement, and the effectiveness of the algorithm improvement could be confirmed.

1. はじめに

近年、DTM (Desk Top Music) による音楽制作や動画投稿サイト等での自作楽曲の公開といった活動が身近なものとなり、作曲に興味を持つ人が増えている。しかし、音楽知識に乏しい初心者にとって作曲は決して容易ではない。このような背景から、作曲未経験者や初心者を対象とした様々な作曲支援システムが研究されてきた。代表的な例としては、歌詞のみを入力として与えることで未経験者でも作曲を楽しめる Orpheus[1]がある。また、ユーザが積極的に作曲過程に介入可能なインタラクティブな作曲支援システムも研究されている。例としては、入力歌詞から生成されたメロディをユーザが編集出来る OrpheusBB[2]がある。このシステムは、一部の音高やコード (和音) がユーザによって編集されると、音楽的に不自然にならないよう残りの箇所を自動変更する機能を持ち、初心者でもメロディの編集を楽しむことが出来る。スマートフォンやタブレット向けの作曲支援アプリケーションもあり、例としては、2小節のメロディを入力するだけで楽曲を自動生成する Chordana Composer[3]がある。

我々は、メロディの一部は作れたがその先に続くメロディが浮かばないという初心者によくみられる問題に着目し、

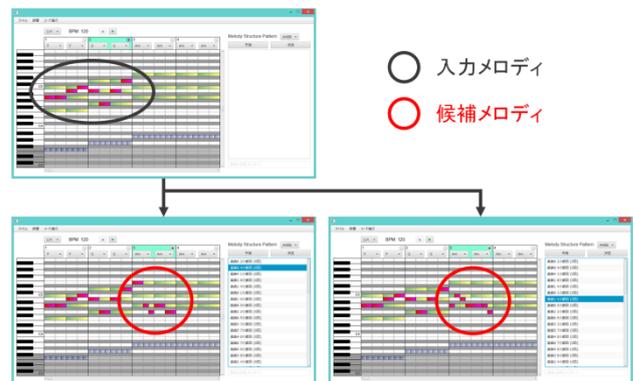


図1 予測変換を用いた作曲支援システム
先に続く多様な候補メロディを提示する

予測変換のアイデアを用いた作曲支援システム[4]を提案した (図1)。提案システムは、ユーザのメロディ展開発想を支援するために、入力されたメロディの先へ続く候補メロディを複数提示するものである。提案システムを実装し、被験者による試用実験でシステム評価も行ったところ、高評価を得られた一方で改善の余地もあることがわかった。

本研究では、先の評価実験から得られた結果を基に、提案システムの改善に取り組んだ。候補メロディの提示アルゴリズムを改善することでより良い候補メロディを提示可能にし、システムの実用性向上を試みる。また、改善の有効性を検証するために、改善前後のシステムによって生成されたメロディを聴き比べてもらう評価実験も行った。本

^{†1} 北海道大学 大学院情報科学研究科
Graduate School of Information Science and Technology, Hokkaido University

論文では評価実験の結果についても報告する。

2. システムの概要

本節では提案システムの概要を述べる。

2.1 予測変換のアイデア

ユーザのメロディ展開発想を支援するためには、入力メロディの先に続く多様な候補メロディを提示する必要がある。多様な候補の提示を実現するために、我々は文章入力の予測変換に着目した。予測変換とは、現在入力中の文字から始まる単語を複数提示する機能であり、代表的な予測変換システムとしては POBox (Pen-Operation Based On eXample) [5]がある。POBox は単語辞書と例文辞書の2種類の辞書を用いて変換候補を検索し、妥当性の高い候補から順にユーザへ提示するシステムである。候補を検索する際、入力中の文字と辞書の登録内容が完全に一致していなくても、マッチングによって類似度を計算することで曖昧な入力にも対応している。

我々は、上記の POBox のアイデアを作曲に適用し、多様な候補メロディをユーザが取捨選択して作曲を進められるシステムを提案した。

2.2 辞書を用いた候補提示

提案システムでは、既存楽曲のメロディを登録した辞書を用いる。メロディを登録する際は、各音符から相対音高 x (直前音符に対する音高変化量)、発音時刻 p 、音価 d を抽出し、 $\{x, p, d\}$ の3つ組の集合から構成される相対メロディ RM として登録しておく。辞書は POBox を参考にし、単語辞書と例文辞書の2種類を用意する。単語辞書にはエントリとして1小節のメロディを登録し、例文辞書にはエントリとして1小節のメロディのペアを登録する。候補メロディを提示する際に、その時の状況に応じて単語辞書と例文辞書のどちらを使うのかを決定する。提案システムは辞書に登録された1小節分の相対メロディを参考にして候補メロディを生成するため、候補メロディの提示は1小節単位で行われる。

2.3 候補メロディ提示の流れ

候補メロディの提示を行う際、ユーザは入力としてコード進行と提示対象小節の番号を与える。以下、候補メロディ群の提示までの流れを説明する。

2.3.1 辞書の選択

最初に、提案システムは提示に使用する辞書の選択を行う。入力された対象小節番号を i とした場合、システムは $i-1$ 小節目の相対メロディ RM_{i-1} と、 i 小節目の相対メロディ RM_i を抽出する。 RM_{i-1} が存在しない場合は単語辞書を使用する。 RM_{i-1} が存在する場合は、1小節前の文脈を考慮するために例文辞書を使用する。

2.3.2 候補メロディ群の生成

選択した辞書を基に、辞書の全エントリに対応する候補メロディを生成する。その際、音域や跳躍度数、コードに

よる音高の遷移確率と出現確率も考慮するために、経路探索による旋律生成手法[6]を用いる。辞書の各エントリに登録された相対メロディの音高変化とリズムを参考にし、確率パラメータも考慮して候補メロディ群を生成する。

2.3.3 候補メロディ群の順位付け

生成した候補メロディ群に対して、文脈による順位付けを行う。辞書の各エントリについて、登録されたメロディと入力メロディとの一致度をマッチングスコアとして計算する。マッチングスコアは、そのエントリに対応する候補メロディと入力メロディとの文脈の一致度を意味するため、マッチングスコアの大きい順に候補メロディ群をソートすることで妥当性の高い候補から順にユーザへ提示することが出来る。マッチングスコアの計算は、DP マッチングを用いてメロディ同士の音高とリズムの類似度を計算し、それらの和をとることで実現する。

2.4 システム実装

提案システムを GUI アプリケーションとして実装した。実装には Java を用いた。システムはピアノロールによって音符を入出力する。メロディは赤色の MIDI ノートとして表示され、ユーザはノートを編集 (追加, 削除, 伸縮, 移動) することが出来る。作曲で使用可能な音域は G3~B5 (MIDI ノート番号 55~83) とし、さらに C メジャースケールの音のみを使用可能としている。また、2 拍単位でコードの指定が可能であり、コードを指定すると伴奏が出力される。

ピアノロールの上部には小節毎にラジオボタンが配置してあり、選択するとその小節が候補メロディ提示の対象小節となる。予測ボタンをクリックすると候補メロディ群の生成が実行され、順位付けされた状態で画面右側のリストに一覧表示される。各候補メロディを選択すると対象小節に MIDI ノートが出力されるため、ユーザは候補メロディを試聴して試行錯誤しながら作曲を進めることが出来る。

3. システムの改善

先の被験者による試用実験では、作曲したメロディおよびシステムのユーザビリティについてほとんどの被験者が高評価を与えていた。その一方で、成果物のメロディに対して低評価を与えた被験者もあり、「良いと思える候補が少なかった」「音高の跳躍が大きすぎる候補があった」といった意見を述べていた。低評価の原因としては単純に個人の好みに合う候補メロディが少なかったことも考えられるが、上記の意見については、例えば非和声音の頻出による不協和感や、大きすぎる音高変化によって引き起こされるメロディの自然さの欠如が原因になっているとも考えられる。このような問題については、候補メロディの提示アルゴリズムを工夫することで改善を期待出来ると考えられる。

以上の結果と考察を踏まえ、本研究では候補メロディの生成・順位付けのアルゴリズムを改善し、より良い (違和

感が少なく、一般に好まれやすい) 候補メロディを提示することを試みる。以下、改善内容について述べる。

3.1 生成アルゴリズムの改善

本節では候補メロディ群の生成に関わる改善内容について述べる。本節で述べる改善は、経路探索による候補メロディ生成に影響を与えるものである。

音符数が I 個の候補メロディを生成する際、各音符 $i = 1, \dots, I$ 、各音高 $j = 0, \dots, P$ 、 $k = 0, \dots, P$ について、以下の式に従って再帰計算を行う。

$$\delta(i, k) = \max_j [\delta(i-1, j) a_{jk}] b_k \quad (1)$$

ここで、 $\delta(i, k)$ は i 番目の音高が k の時の $i-1$ 番目の音符までのメロディ最大確率を示し、 a_{jk} は音高 j から音高 k への遷移確率、 b_k は音高 k の出現確率を示す。 a_{jk} は、音域に対する遷移確率 a_{jk}^{range} および跳躍度数に対する遷移確率 a_{jk}^{jump} と、係数 c_{jk} により計算される。 b_k は、音域に対する出現確率 b_k^{range} およびコードに対する出現確率 b_k^{chord} と、3つの係数 c_{k1} 、 c_{k2} 、 c_{k3} により計算される。

$$a_{jk} = a_{jk}^{range} a_{jk}^{jump} c_{jk} \quad (2)$$

$$b_k = b_k^{range} b_k^{chord} c_{k1} c_{k2} c_{k3} \quad (3)$$

(1)式による再帰計算終了後にバックトラックを行うことで、最大確率を与えるメロディが得られるため、これを候補メロディとする。

3.1.1 音高遷移制約の緩和

提案システムでは、確率最大の経路を探索する際に経路制約[7]を適用し、参考にする相対メロディの音高変化をなぞるように候補メロディを生成する。 $i-1$ 番目と i 番目の音符について考える時、相対音高 x の変化量に応じて以下の経路制約を適用し、 c_{jk} を計算する。

- $x_{i-1} = x_i$ の場合、制約(1) : $j = k$ 以外の経路を抑制
- $x_{i-1} > x_i$ の場合、制約(2) : $j > k$ 以外の経路を禁止
- $x_{i-1} < x_i$ の場合、制約(3) : $j < k$ 以外の経路を禁止

$$c_{jk} = \begin{cases} 1.0 & \text{適用された制約に反しない時} \\ 0.5 & \text{制約が適用され、反する時} \\ 0.0 & \text{制約が適用され、反する時} \end{cases} \quad (4)$$

従来は制約(1)に反する場合も(2)(3)と同様に $c_{jk} = 0.0$ としていたが、その状態で制約(1)が適用された場合、同じ音高が続けて出力されることになる。これにより非和声音が連続して現れる候補メロディが生成されることがあり、結果として不協和感の原因になりうるという問題がある。よって、制約(1)に反する場合は $c_{jk} = 0.5$ とするよう変更し、音高の上昇・下降の禁止度合いを緩和した。これにより、相対メロディをなぞりつつも、必要な場合は従わないことで、不協和感の大きい候補メロディが生成されることを防ぐ。

3.1.2 非和声音制約の導入

非和声音は経過音や刺繍音[8]として出現した場合は不協和感が小さいが、倚音(コードの冒頭部分に出現する非和声音)や音価が大きい音符で出現した場合は不協和感を

感じやすいと考えられる。

この問題の対策として、以下に示す非和声音制約を新たに導入し、 c_{k1} および c_{k2} を計算する。ここでは、音価が大きい音符は4分音符よりも長い音符とした。

- 倚音の制約

$$c_{k1} = \begin{cases} 0.1 & k \text{ が倚音の時} \\ 1.0 & \text{otherwise} \end{cases} \quad (5)$$

- 音価が大きい音符の制約

$$c_{k2} = \begin{cases} 0.1 & k \text{ が非和声音で4分音符より長い時} \\ 1.0 & \text{otherwise} \end{cases} \quad (6)$$

上記の制約により、不協和感が大きくなりやすい状況で非和声音が出現することを防ぐ。

3.1.3 音域バランス制約の導入

ポピュラー音楽における一かたまりのメロディ(例:Aメロ、Bメロ、サビ等)について考えると、その音域は1オクターブ程度であることが多い。例えば4小節や8小節のメロディの中で1オクターブを超える音高変化があった場合、激しい音高変化に対してリスナーは不自然さを感じやすいと考えられる。このような問題を解決するには、メロディ全体の音域を常に意識し、その音域から大きく外れないように続きのメロディを設計する必要がある。

そこで、正規分布による音域バランス制約を新たに導入し、上記の問題の解決を試みる。

$$c_{k3} = \begin{cases} \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(k-\mu)^2}{2\sigma^2}\right) & \sigma > 3.0 \\ 1.0 & \sigma \leq 3.0 \end{cases} \quad (7)$$

ここで、 μ と σ はそれぞれ、入力メロディの音高の平均と標準偏差を示す。 μ については、各音符の音高 $pitch$ と音価 d を用いた重み付き平均として計算する。入力メロディが N 個の音符から構成される時、 μ と σ は以下の式で計算される。

$$\mu = \frac{d_1 pitch_1 + d_2 pitch_2 + \dots + d_N pitch_N}{d_1 + d_2 + \dots + d_N} \quad (8)$$

$$\sigma = \sqrt{\frac{1}{N} \sum_{n=1}^N (pitch_n - \mu)^2} \quad (9)$$

(7)式の確率密度関数は、入力メロディの音域から外れる音高ほど出力値が小さくなる。この出力値を係数とすることで、音域から大きく外れる音符が候補メロディに出現することを防ぐ。ただしこの方法では、入力メロディの音高変化が小さい場合に、候補メロディもまた音高変化が小さいものが出現しやすくなり、結果として変化に乏しい単調なメロディが生成されやすくなる。その対策として、標準偏差が小さい場合 ($\sigma \leq 3.0$) は $c_{k3} = 1.0$ とすることで音域バランス制約の影響が出ないようにした。

3.1.4 音高出現確率の見直し

2種の音高出現確率 b^{range} と b^{chord} の見直しを行った。

b^{range} については、従来は著者が独自に確率値を設定していたが、正規分布により確率値を得よう変更した。有

効な音高 (G3~B5) の平均と標準偏差を求め、それによって定まる正規分布から各音高の出現確率を得る。

b^{chord} については、従来は和声音の出現確率を高めに、非和声音 (7th ノートやテンションノート) の出現確率を低めに設定していた。これに加え、対象コードについてアポイドノートが考えられる場合は、その音の出現確率を最も低くするよう変更した。

3.2 順位付けアルゴリズムの改善

本節では候補メロディ群の順位付けに関わる改善内容について述べる。本節で述べる改善は、候補メロディ群を順位付けする際のスコア計算に影響を与えるものである。

K 個の候補メロディから成る候補メロディ群がある時、 $k = 1, \dots, K$ についてマッチングスコア M_k を計算する。 M_k は文脈を考慮して計算され、値が大きいくほど候補メロディとして妥当であることを意味する。 M_k を計算する際、従来は音高の類似度とリズムの類似度の和 S_{k1} のみを用いていたが、これに加えてメロディ構造スコア S_{k2} および音域バランススコア S_{k3} を新たに導入した。

$$M_k = S_{k1} + S_{k2} + S_{k3} \quad (10)$$

以下、新たに導入した S_{k2} および S_{k3} について説明する。

3.2.1 メロディ構造スコアの導入

ポピュラー音楽のメロディには、類似したリズムのフレーズを小節単位で繰り返す構造が見られることが多い。例えば、スピッツの楽曲『チェリー』のサビ部分 1 小節目から 4 小節目を見ると、1 小節目と 3 小節目、および 2 小節目と 4 小節目は完全に同一もしくは類似したフレーズから展開されている。このような繰り返し構造を持つメロディは、リスナーにとっても覚えやすく好まれやすい傾向があると考えられる。

上記のような構造を持つ候補メロディを提示するために、メロディへの構造付けを新たに導入する。作曲するメロディの長さを 4 小節に限定した上で、4 種のメロディ構造を定義する (表 1)。ここで、ダッシュなしのラベルの小節を参考小節、ダッシュ付きのラベルの小節を繰り返し小節と呼ぶ。繰り返し小節のメロディは、参考小節のメロディとリズムが類似するものを提示する。例えば構造 1 を適用した場合は、3 小節目は 1 小節目と、4 小節目は 2 小節目とリズムが類似するメロディを提示する。どのメロディ構造を適用するかは、ユーザが指定するものとする。

繰り返し小節のメロディを提示する際に、各候補メロディについてメロディ構造スコア S_{k2} を計算し、マッチングスコアに加算する。 S_{k2} は参考小節のメロディと候補メロディとのリズム類似度として計算する。これにより、指定した構造を持つ候補メロディを上位に提示することを試みる。リズム類似度は DP マッチングを用いて計算する。音符数 I の繰り返し小節のメロディ ref の音符インデックスを i 、音符数 J の候補メロディ $cand$ の音符インデックスを j とし、各音符の発音時刻 p と音価 d について、以下の式に従い局所距

表 1 導入するメロディ構造

| | 1 小節目 | 2 小節目 | 3 小節目 | 4 小節目 |
|------|-------|-------|-------|-------|
| 構造 1 | A | B | A' | B' |
| 構造 2 | A | A' | B | B' |
| 構造 3 | A | A' | B | C |
| 構造 4 | A | B | C | C' |

離 $D(i, j)$ と累積距離 $G(i, j)$ を計算する。そして、 $G(I, J)$ よりリズム類似度を得て、これを S_{k2} とする。

$$D(i, j) = \sqrt{(p_i^{ref} - p_j^{cand})^2 + (d_i^{ref} - d_j^{cand})^2} \quad (11)$$

$$G(i, j) = \min \begin{cases} G(i-1, j) + D(i, j) \\ G(i-1, j-1) + 2D(i, j) \\ G(i, j-1) + D(i, j) \end{cases} \quad (12)$$

3.2.2 音域バランススコアの導入

候補メロディ群の順位付けにおいても、3.1.3 節で述べた音域バランス制約を導入する。各候補メロディについて音域バランススコア S_{k3} を計算し、マッチングスコアに加算する。これにより、入力メロディの音域から外れすぎない候補メロディを上位に提示し、メロディ全体の音域のバランスを保つことを試みる。 S_{k3} は以下に示す式で計算する。

$$S_{k3} = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(\bar{\mu}_k - \mu)^2}{2\sigma^2}\right) \quad (13)$$

ここで μ と σ はそれぞれ(8)式と(9)式により計算する。 $\bar{\mu}_k$ は k 番目の候補メロディの音高の平均を示し、各音符の音高 \overline{pitch} と音価 \bar{d} を用いた重み付き平均として計算する。候補メロディが M 個の音符から構成される時、 $\bar{\mu}_k$ は以下の式で計算される。

$$\bar{\mu}_k = \frac{\bar{d}_1 \overline{pitch}_1 + \bar{d}_2 \overline{pitch}_2 + \dots + \bar{d}_M \overline{pitch}_M}{\bar{d}_1 + \bar{d}_2 + \dots + \bar{d}_M} \quad (14)$$

4. 評価実験

3 章で述べたアルゴリズム改善の有効性を検証するため、被験者実験を行った。

4.1 実験内容

改善前と改善後のそれぞれのシステムで生成したメロディを被験者に聴き比べてもらう実験を行った。聴き比べの設定は合計 8 問用意した。各設問で 2 つのメロディ (A と B) を提示し、被験者は気が済むまで A と B を試聴した後アンケートに回答する (表 2)。

4.2 実験設定

実験に用いる全てのメロディについて、長さは 4 小節、BPM は 120 とした。コード進行は設問毎に異なるものを著者が設定した。1 小節目のメロディは著者が入力して与え、2~4 小節目のメロディはシステムが出力する候補メロディを用いた。候補メロディは編集を加えず、そのままの形で用いた。また、メロディはピアノ、ベース、ドラムスによる簡易な伴奏と共に演奏される。実験では既存楽曲 8 曲

表2 アンケート項目

| | 内容 | 評価 |
|----|----------------------------------|--|
| Q1 | メロディ A に点数を付けるなら何点か？ | 7 点満点 |
| Q2 | メロディ B に点数を付けるなら何点か？ | |
| Q3 | メロディ A と B とではどちらがより良いと思うか？ | 7 段階 1 : A が良い (自然) 7 : B が良い (自然) |
| Q4 | メロディ A と B とではどちらがより自然に聴こえたと思うか？ | |

表3 評価実験で用いた既存楽曲

| | アーティスト名 | 曲名 |
|------|---------------------|----------|
| 楽曲 1 | スピッツ | ヒバリのころ |
| 楽曲 2 | スピッツ | 空も飛べるはず |
| 楽曲 3 | スピッツ | 春の歌 |
| 楽曲 4 | スピッツ | 魔法のコトバ |
| 楽曲 5 | BUMP OF CHICKEN | 天体観測 |
| 楽曲 6 | KANA-BOON | シルエット |
| 楽曲 7 | supercell | 君の知らない物語 |
| 楽曲 8 | supercell feat.初音ミク | メルト |

のサビ区間 8 小節分を用いて作成した辞書を用いた(表 3)。

各設問に対して改善前と改善後のメロディのペアを用意する際、コード進行、1 小節目のメロディ、そして 2~4 小節目の各小節で選択する候補メロディの順位をシステム間で統一した。A と B のどちらに改善前と改善後のメロディを割り当てるかは、ランダムに決定した。

4.3 実験結果

20 代~30 代の男女 20 名 (男性 15 名, 女性 5 名) から回答を得られた。

4.3.1 Q1 と Q2 による評価

Q1 と Q2 の回答から、改善前・改善後のメロディに対する評価の平均点を被験者毎に求めた(図 2)。被験者毎の平均点に対してウィルコクソンの符号順位検定を行ったところ、両側検定と片側検定の両方で有意差が見られた ($p < 0.01$)。被験者毎の平均点の平均は、改善前が約 4.238、改善後が約 4.806 となった。この結果から、改善前と比べて改善後の方が高評価であることが示された。図 2 の結果を見ると、全ての被験者について平均点が改善前 \leq 改善後となっていることがわかる。

4.3.2 Q3 と Q4 による評価

Q3 と Q4 で得られた回答を、改善後のメロディに対する点数 (7 点満点) に変換した。変換点の平均を被験者毎に求め、平均点 > 中間点 (4 点) となった被験者を改善後の

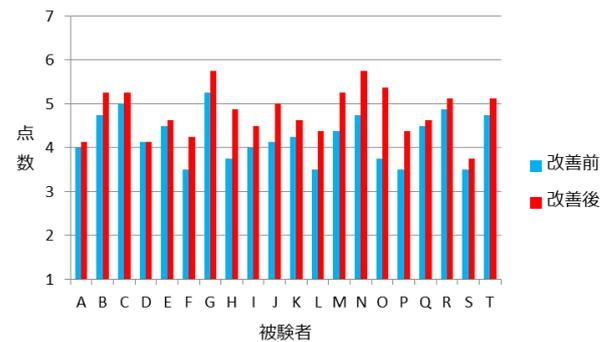


図2 各被験者の改善前・改善後の平均点

メロディに対して高評価を与えた者と見なし、その人数を Q3 と Q4 について数えた。結果は、Q3 は 17 人 (被験者全体の 85%)、Q4 は 19 人 (被験者全体の 95%) となった。被験者毎の平均点の平均は、Q3 が約 4.494、Q4 が約 4.788 となった。この結果からも、改善前と比べて改善後の方が高評価であることがわかる。

4.4 考察

4.3 節で述べた結果から、全体として改善前よりも改善後のメロディの方が高評価であり、より多くの人に好まれやすいメロディを生成出来ていると考えられる。

実験で用いたメロディを確認したところ、改善前のメロディの多くには以下の特徴が見受けられた。

- (1) 連続した非和声音の出現
- (2) 倚音の出現
- (3) 音価の大きな非和声音の出現
- (4) 広域な音域 (1 オクターブ前後)
- (5) 非和声音同士の跳躍

これらの特徴が不協和感や不自然さの要因となり、改善前のメロディに高評価を与える被験者が少なかったと考えられる。一方、改善後のメロディについては上記の特徴が見受けられたものは少なく、多くが和声音を多用したメロディとなっていた。改善後のメロディにも非和声音は出現していたが、ほとんどは経過音や刺繍音として出現していたため、不協和感や不自然さを感じない範囲で非和声音を使用し、和声音のみの単調なメロディになることも防ぐことが出来ていたと考えられる。結果として、改善後のメロディに対して良さや自然さを感じ、高評価を与える被験者が多かったと考えられる。

改善後のメロディに特徴(1)~(3)が表れなかったのは、3.1.1 節と 3.1.2 節で述べた、非和声音の対策を目的とした改善が機能しているためだと考えられる。音域については、特徴(4)にも示したように改善前のメロディには音域が 1 オクターブ以上のももあったのに対し、改善後のメロディは全て音域が 1 オクターブ以内に収まっていた。これは、3.1.3 節と 3.2.2 節で述べた、音域のバランスを保つことを目的とした改善が機能しているためだと考えられる。

実験では4個の改善後のメロディに対し、3.2.1節で述べた構造付けを適用した。構造1～構造4の4種を1つずつ適用したが、はっきりと繰り返しを認識出来るような構造はメロディに現れなかった。原因としては、メロディ構造スコアが順位付けに与える影響が小さい等といったことが考えられる。解決策としては、 $S_{k1} \sim S_{k3}$ のそれぞれに対して重みを設定し、スコア毎に重視する度合いを調整するといったことが考えられる。

4.4.1 改善後が高評価だったケース

多くの被験者が改善後を高評価した設問について考察する。改善前、改善後のメロディを図3に示す。この設問の改善前と改善後の平均点はそれぞれ3.7と5.6、Q3とQ4の平均点はともに5.75となった。

改善前のメロディには非和声音が頻出している。より詳しく見ると、3小節目ではB音が比較的長い音で出現し、4小節目ではD音が連続して出現している。このような非和声音に不協和感を感じ、改善前に高評価を与えた被験者が少なかったと考えられる。

一方、改善後のメロディにも非和声音は出現しているが、いずれも経過音や刺繍音として出現しているため不協和感は小さい。また、メロディ全体が4分音符を中心とした似通ったリズムで構成されており、改善前と比べると全体にまとまりを感じられる。これらの要因から、改善後が高評価を与えた被験者が多かったと考えられる。

4.4.2 改善前が高評価だったケース

多くの被験者が改善前を高評価した設問について考察する。改善前、改善後のメロディを図4に示す。この設問の改善前と改善後の平均点はそれぞれ5.1と4.75、Q3とQ4の平均点はそれぞれ4.45、4.05となり、改善前の方が高評価傾向という結果になった。

改善前のメロディを見ると、非和声音は出現しているものの、長い音符や連続での出現は見受けられず、不協和感は小さい。また、4小節目に着目するとC音が全音符として出現しており、終止感を感じられる。これらの要因から、改善前に高評価を与えた被験者が多かったと考えられる。

5. 終わりに

本研究では、予測変換を用いた作曲支援システムの有用性向上を目的として、システムの改善に取り組んだ。候補メロディの生成・順位付けのアルゴリズムを改善し、従来と比べてより良い候補メロディを提示することを試みた。改善前後のシステムによって生成されたメロディを被験者に聴き比べてもらう実験を行ったところ、多くの被験者が改善後のメロディに対して高評価を与える結果となり、アルゴリズム改善の有効性を確認出来た。

今後は、システムを実際に試用してもらう被験者実験を行う予定である。アルゴリズムの改善がシステム全体の有用性向上に効果があるか検証するためには、実際に改善前



図3 改善後が高評価だったケース
改善前 (上) と改善後 (下)



図4 改善前が高評価だったケース
改善前 (上) と改善後 (下)

後のシステムを用いて作曲をしてもらい、評価に差が見られるか確認する必要がある。尚、実際にシステムを使用してもらう被験者実験は既に一度実施しているが、その際の被験者数は5人であり、決して十分とは言えない人数だった。今後被験者実験を行う際は被験者数を増やし、より妥当性の高い実験結果を集めたいと考えている。

参考文献

- [1] 深山覚, 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, 2008-MUS-76, pp.179-184, 2008.
- [2] 北原鉄朗, 深山覚, 片寄晴弘, 嵯峨山茂樹, 長田典子. OrpheusBB: Human-in-the-loop 型の自動作曲システム. インタラクシオン 2011, pp.57-64, 2011.
- [3] “作曲アプリ「Chordana Composer (コードナコンポーザー)」 - カシオ計算機”. <http://web.casio.com/app/ja/composer/>, (参照 2017-12-20).
- [4] 山下峻, 藍圭介, エバンズベンジャミン, 棟方渚, 小野哲雄. 予測変換のアイデアを用いた作曲支援システムの提案. 情報処理学会研究報告, 2017-MUS-114, no.20, pp.1-6, 2017.
- [5] Toshiyuki Masui. An Efficient Text Input Method for Pen-based Computers. In Proceedings of the ACM Conference on Human Factors In Computing Systems CHI'98, Addison-Wesley, pp.328-335, 1998.
- [6] 深山覚, 西本卓也, 小野順貴, 嵯峨山茂樹. 非和声音規則に基づく経路制約を用いた旋律自動生成. 情報処理学会研究報告, 2009-MUS-81, no.15, pp.1-6, 2009.
- [7] 深山覚, 齋藤大輔, 嵯峨山茂樹. 日本語歌詞からの自動作曲における DP 経路制約による旋律制御. 情報処理学会第 75 回全国大会講演論文集, pp.65-66, 2013.
- [8] 北川祐. ポピュラー音楽理論. リットーミュージック, 2008.