

自由度が徐々に上がる初心者向け作曲ソフトウェアの試作

姜 楊禹辰^{1,a)} 名越 崇晃^{1,b)} 北原 鉄朗^{1,c)}

概要：本稿では、作曲未経験者を対象とした作曲支援システムについて述べる。本システムでは、旋律概形の描画によってメロディを作成した後、各音符の音高を編集することができる。その際、使用できる音に制約を掛けることで、音楽的に妥当なもののみが使用できる状態にする。この制約は、将来的に支援なしにメロディ作成ができるように、段階的に小さくする。これにより、音楽知識のない作曲未経験者であっても簡単にメロディの作成ができ、慣れに従って自由度の高い多様なメロディ作成を行えるようになると期待される。

1. はじめに

現在、音楽は主要な娯楽である。これには視聴するのみならず、作曲を試みる者も存在する。しかし、作曲は多くの人にとって難しい作業である。現在流通している作曲ソフトウェアは、自由度が高いため、音楽知識のない人にとって、取るべき選択肢が多く、音楽的に妥当な楽曲を作成することは容易ではない。

既存の初心者向けの楽曲作曲支援を目的とした研究は多い。そのうち、音楽を自動生成するものが多いが、これらはユーザの介入が低く、ユーザが自身で作成した実感が得られにくいことが課題に挙げられる。これに対して、音楽の自動生成を主眼とせず、作曲支援のユーザインタラクションに主眼を置いた研究がいくつか存在する。たとえば、小林ら [1] は、素材を選択して作曲するという作曲の手段において、その素材の選択を補助するシステムを提案した。澁谷ら [2] は、作曲の未経験者が自分なりの法則や指針を見つけることを支援するシステムを提案した。山根ら [3] は、作曲ソフトウェアにロールプレイングゲーム (RPG) の要素を加えることにより、作曲の学習の効果を向上させることを目的としたシステムを提案した。牧野 [4] らはテーブルにカードを並べることで、触覚を取り込み、直感的に作曲ができるシステムを提案した。他にも、畦原らのインタラクティブ作曲支援システム [5]、Granger らの開発した、リアルタイムでコード進行やメロディの要素を提案する Lumanote [6]、Bouche らが開発した、音楽のオブジェクトを動的に変化する構造としてモデル化し、柔軟で創造的な

音楽制作を提供するシステム [7] など、様々なものが提案されている。

支援システムの設計において重要なのは、いずれ支援なしの創作作業を実施できるようになることを目指すかどうかである。我々が目指すのは、ユーザが最終的には支援機能に頼らずに作曲できるようになることである。作曲することを、メロディの音高と音価を決めることであると定義するなら、最初は、たとえば伴奏と不協和になったり音楽的に不自然になったりしないように、コンピュータによって音高や音価などが決められるものの、そのような操作を繰り返すたびにユーザ自身が慣れていき、ユーザが自ら音高と音価を適切に選べるようになる。このような状況を目指していく。

本稿では、このような考えに基づいて実装した作曲支援システムについて述べる。本システムとベースになっているのは、旋律概形に基づく即興演奏システム「JamSketch」[8] である。ユーザは、システム上で旋律概形を描くことで、まずベースとなる旋律を作成する。その後、その旋律を聴き、自分好みに作り替えていく。ここでは特に、音高の変更を取り上げる。この際、最初は、音楽理論に基づいて不協和になりにくい音に制限されているのに対し、段階的にそのような制約を外し、より自由度の高い旋律作成ができるようにする。

2. 提案システム

本稿で提案するのは、徐々に制約がはずれていくメロディ作成システムである。このシステムには、主に次の 2 つの機能がある。

- ピアノロール画面上にマウスなどで旋律概形（旋律のおおまかな形を表す曲線）を描くことで、旋律が自動

¹ 日本大学

^{a)} jan@kthrlab.jp

^{b)} nagoshi@kthrlab.jp

^{c)} kitahara@kthrlab.jp

生成される。

- 編集モードに切り替えることで、自動生成された音符の各音高を手動で変更することができる。複数の制約モードを備え、そのモードに応じて変更先に音高が制限される。

以下、本システムの具体的な内容を述べる。なお、本システムの実装は、前述の JamSketch[8] を拡張することにより行った。

2.1 作成するメロディの制約

現在の実装では、簡単のため、メロディは 12 小節とし、キーとコード進行はあらかじめ与えるものとする。拍子は 4 分の 4 とする。4 分音符をいくつに分割するか（ディビジョンと呼ぶ）をあらかじめ決め、これを音価の最小単位とする。デフォルトでは、キーは C、コード進行は [C, G, C, G, C, G, C, C, G, G, C, C]、ディビジョンは 3 である。

2.2 ピアノロール画面の表示

本システムを立ち上げると、ピアノロール画面が表示される。現在の実装では、表示される音域は C4 を中心とした 3 オクターブ程度に制限される。

2.3 旋律概形の描画と旋律の生成

JamSketch[8] と同様、ピアノロール画面上で旋律概形を描くことができる。マウスなどのドラッグ操作で旋律概形を描くと、青色の曲線として表示される。

既存の旋律をガイドメロディとして表示することもできる。与えられた MIDI ファイルのチャンネル 1 の MIDI シーケンスを読み取り、その旋律の概形を水色の線で表示することができる。ユーザは、その線をなぞってもよく（指定された既存のメロディに近いメロディが得られる）、自分の意志で部分的にそこから逸脱しても構わない。

旋律概形が描かれると、即座にメロディが生成される。元々のバージョンでは、遺伝的アルゴリズムを用いて旋律が生成されていた [8] が、本システムでは、後から編集する前提であるため、より単純なマルコフモデルを用いている。ただし、後で述べる制約モードによって使用が許可されない音高への遷移確率を 0 にすることで、現在の制約モードの範囲内のメロディが生成されるようにしている。

2.4 編集モードにおける手動での音高の変更

上記の手法によって旋律が生成されたら、編集モードに切り替えて手法で音高を変更することができる。音高の変更は、通常の MIDI シーケンサなどと同様に、音符を表すバーをクリックして選択した後、それを上下にドラッグ＆ドロップすることにより行う。

本システムには、前述の通り、複数の制約モードを備える。ここでいう制約とは、音高の編集の際に選択できる音

高に制限を与えるものである。たとえば、一般に強拍の拍頭にコード構成音やスケールに含まれない音が使われると、不協和の原因となり、強い違和感を生じさせる。そのような音高は選べなくすることで、音楽知識のない人でも手軽に音高の編集をできるようにするのが、この制約モードの趣旨である。本システムでは、次の 4 つの制約モードを備える。

- 強制制約モード
 - － 強拍の拍頭：コード構成音のみ使用可能
 - － それ以外：キーのスケールの構成音のみ使用可能
- 中制約モード
 - － 強拍の拍頭：キーのスケールの構成音のみ使用可能
 - － それ以外：キーのスケールの構成音のみ使用可能
- 弱制約モード
 - － 強拍の拍頭：キーのスケールの構成音のみ使用可能
 - － それ以外：制限なし
- 無制約モード
 - － 強拍の拍頭：なし
 - － それ以外：なし

強拍の拍頭は、それ以外に比べて不協和音程が目立つ傾向にあるため、他より強めの制約とし、コード構成音、スケール構成音、制限なしの 3 段階を割り振る形で定義した。

2.5 制約モードの切り替え

制約モードの切り替えは、現在の実装ではユーザが手動で行う。本来であれば、メロディ編集に対する慣れの度合いをシステムが自動的に判定し、それに基づいて行うのが望ましいが、これは今後の課題とする。

3. 実行例

システムの実行例を示す。システムを起動した直後は図 1 となる。左側に縦に 3 オクターブ分のピアノ鍵盤が表示され、横には 12 小節分のマスが存在する。各小節のマスはそれぞれ 12 分割されており、また、1 小節分は 4 拍と定められている。画面には水色の線が描かれているが、これは先述のように、旋律概形の入力の例として表示されているものであり、入力ではない。マウスカーソルの位置には青色の縁で囲まれている赤色の円型が表示されている。

入力を加えたものが図 2 である。青色の線がマウスで引いた旋律概形である。この旋律概形に沿って生成された音が黄色のブロックが表示されている。このように、水色の線に沿って描いてもよく、これを無視しても構わない。

図 2 は、強制制約を適用した状態である。つまり、強拍の拍頭ではその小節の伴奏のコードの構成音しか使用できず、それ以外ではキーのスケールの構成音しか使用できない。なお、生成された音が 1 拍より長い場合、部分的に制約に合致しない場合がある設定になっている。図 2 での 1 小節目を確認すると、ここの最初の音として「ソ」が生成されて

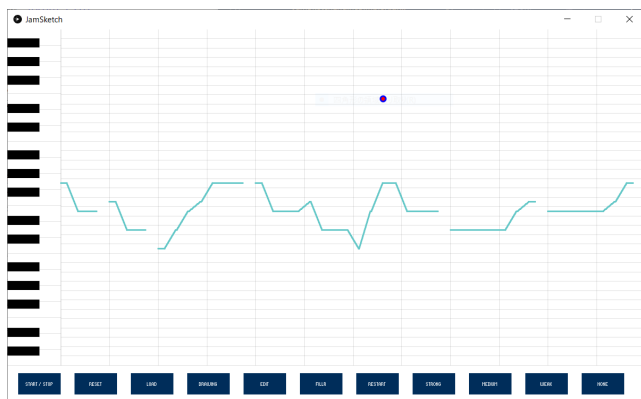


図 1 JamSketch 操作画面

いる。1小節目の伴奏のコードはCであるため、制約に当てはまっている。その次の音はファである。ここは弱拍であり、キーCのスケールの構成音は「ド、レ、ミ、ファ、ソ、ラ、シ」なので、制約内に当てはまっている。このように、旋律概形に基づくメロディ生成において、制約内に当てはまった音のみが生成されている。

図3では、編集モードを使用して、1小節目の拍頭の音を「ミ」に移動させた。1小節目の伴奏のコードはCであるため、制約内としてこの音を使用することができた。その次の音を「ミ」に移動させた。ここは弱拍であるため、キーCのスケールの構成音であればよく、使用することができた。このように、編集機能でも、制約に当てはまっている音を使用することができた。

また、他の制約モードでの実行例も表示する。図4は中制約モードの場合である。1小節目と2小節目にまだがって描かれている旋律概形は「ド♯」の位置にあるが、その位置には音は生成されず、代わりに「ミ」の音が生成されている。キーCのスケールの構成音は「ド、レ、ミ、ファ、ソ、ラ、シ」に含まれているため、制約内に当てはまった音のみが生成されている。図5は弱制約モードの場合である。1小節目の音は「ド」であり、キーCのスケールの構成音は「ド、レ、ミ、ファ、ソ、ラ、シ」に含まれている。また、2小節目の旋律概形は「ド♯」であるが、強拍の音は「レ」であり、キーCのスケールの構成音は「ド、レ、ミ、ファ、ソ、ラ、シ」に含まれている。一方、弱拍の音は順に「ド♯」、「レ」、「レ♯」である。キーCのスケールの構成音は「ド、レ、ミ、ファ、ソ、ラ、シ」に含まれていない音であり、制約が掛かっていないのがわかる。図6は無制約モードの場合である。強拍弱拍関わらず、12音が満遍なく生成されており、その上旋律概形に忠実に沿っている。

4. おわりに

本稿では、使用可能な音の制約を段階的に変えることができる作曲支援システムを提案した。当初は強めの制約があることで、作曲未経験者でも音楽的に妥当な音を選べるようにし、慣れに従って制約モードを変えることで、メロ

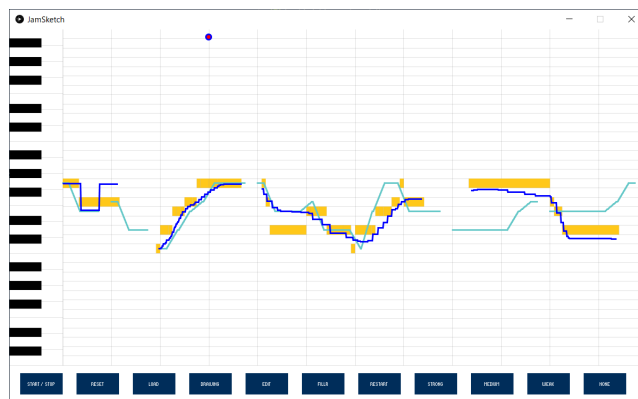


図 2 旋律概形の入力：強制モード

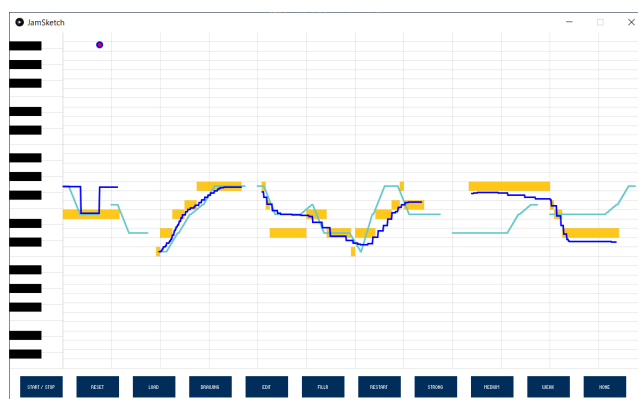


図 3 生成後の音の編集

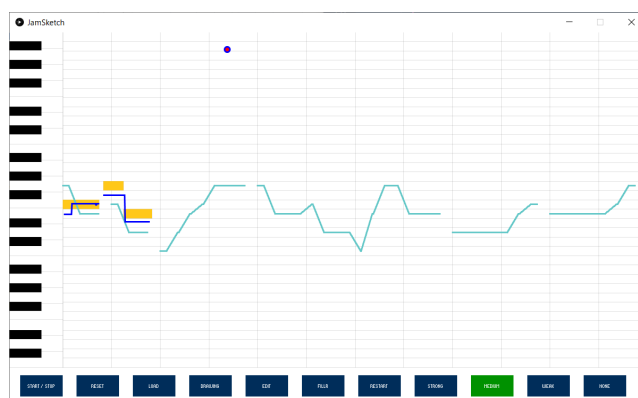


図 4 旋律概形の入力：中制約モード

ディ作成の幅を広げられるようにした。しかし、本システムによって本当にメロディ作成の幅を広げられるのかは確認できていない。今後は、実際に作曲未経験者に本システムを使ってもらうことで、検証していきたい。

謝辞

本研究は、JSPS 科研費 23K24966, 24H00748 の助成を受けた。

参考文献

- [1] 小林篤矢, 佐藤哲朗, 舘野啓, “楽曲制作初心者向けアプリケーションの開発と評価: ダブルダイヤモンドモデルに沿ったインタフェースデザイン”, インタラクシ

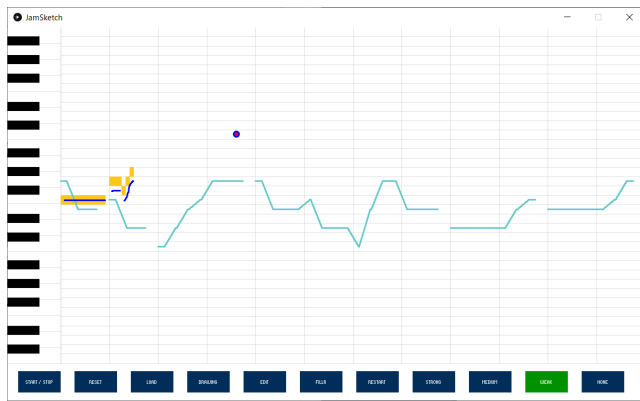


図 5 旋律概形の入力：弱制約モード

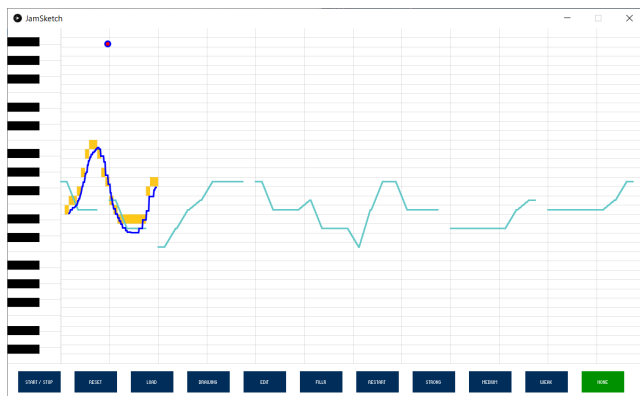


図 6 旋律概形の入力：無制約モード

2024, 情報処理学会, 2B-22, 724-729, 2024.

- [2] 澁谷 智志, 木村 健一, “作曲の熟達過程を支援するシステムの提案”, 日本デザイン学会 第 52 回研究発表大会, 2005
- [3] 山根 諒, 田中 春菜, 米澤 朋子, “初心者の作曲動機を向上する RPG 要素によるステップ入力型作曲システム”, 2016 年度 情報処理学会関西支部 支部大会 講演論文集, 2016
- [4] 牧野 真緒, Rodney Berry, 樋川 直人, 鈴木 雅実, “作曲・演奏支援システム The Music Table”, インタラクション 2004
- [5] 畦原 宗之, 鬼沢 武久, “インタラクティブ作曲支援システム – ユーザの負担の軽減 –, The 17th Annual Conference of Japanese Society for Artificial Intelligence, 2003
- [6] James Granger, Mateo Aviles, Joshua Kirby, Austin Griffin, Rebel Yoon, Raniero Lara-Garduno, Tracy Hammond, “Lumanote: A Real-Time Interactive Music Composition Assistant”, MILC '18, 2018
- [7] Dimitri Bouche, Jérôme Nika, Alex Chechile, Jean Bresson, “Computer-Aided Composition of Musical Processes”, Journal of New Music Research, 2017
- [8] T. Kitahara, S. Giraldo, and R. Ramírez, “JamSketch: A Drawing-based Real-time Evolutionary Improvisation Support System”, Proceedings of the International Conference on New Interfaces for Music Expression (NIME 2017), 2017.