

コード進行をベースにして 作曲を支援するシステム

内橋由佳
伊丹 誠

平野健太郎
伊藤紘二

東京理科大学 基礎工学部

曲の構成をメロディ(旋律)、コード(和音)、リズム(正規の拍に乗った他声部の音)の3部分に分けて考えてみる。ここでのメロディは曲の核部分、コードは曲の雰囲気の流れの部分、リズムは曲の性格を決定する部分である。我々が、作曲をする時、通常はまずメロディを生成してからコードやリズムをつけるだろう。しかし、実際はメロディを生成する時、無意識のうちに曲の雰囲気の流れ(コード進行)を考えていると思われる。そこで、本システムでは、コード進行を生成して曲の雰囲気作りをしながらメロディを作っていくという方法に意味があるという、仮説を基に作曲を支援することを試みた。また、本システムでは規範的なコードとコードモーションのみを知識として提供し、利用者が、利用者のレベルに合わせて新しい知識を登録できるようにし、何度も聴きながらコード進行を生成できるようにした。この時音楽的理論からの制約は加えないものとする。こうして、出来上がったコード進行を基に作曲を支援することにより、今まで敬遠されがちであった作曲がより身近になることが期待される。

A support system for composing music based on chord progressions

Yuka Uchihashi, Kentaro Hirano,

Makoto Itami, Kohji Ito,

Department of Applied Electronics,
Science University of Tokyo

A piece of music is composed of three parts: melody, chords, and rhythm. Among which the melody part is its core. The chords determine its atmosphere and the rhythm part gives its character. When we try to compose a piece of music, we usually compose a melody first. However it that seems we feel atmosphere of a chord progression while we are composing the melody. We propose a support system for composing music in which it is recommended and assisted to commpose chord progressions first. Based on this intuition, the system provides users with only basic menus and preference rules on chords and chord motions. The user can add to his/her own menus and preference rules. He/She retrieves and selects from the menus and rules by listening, concatenating the selected motions, thus can compose a chord progression. This system doesn't impose users any compulsory restrictions by musical theory. The system is expected to offer amateur musicians more opportunities for composing music by themselves.

1 はじめに

近年、人々の音楽に対する興味は高まる一方である。多くの子供達は幼い頃から音楽教室に通ってピアノを習い、若者達は競ってアマチュアバンドを結成し、好きなミュージシャンの曲を演奏している。年配の方々においても、音楽観賞を趣味としている方は少なくない。しかしながら、ピアノやギターなどの楽器の演奏に対してかなりの自信がある人でも作曲となると尻込みしてしまう人がほとんどである。作曲といういかにも専門的で、音楽的知識を備えた人たちのもの、という考えが定着しているからである。しかし、現在プロとして活躍している作曲家の多くは、必ずしも専門的な音楽教育を受けてきたわけではない。毎日の生活の中で、いろいろな音楽に親しみ、音感を養ってきたわけで、その意味からいうと我々も作曲の才能が日々養われているのかも知れない。

しかし、いざピアノの前に座り、鍵盤を適当に叩いてみたところで、それらしく聴こえるものはできないであろう。たとえ、何か良いメロディができたとしても、その音はたちまち消えてしまい、二度と同じメロディを聴くことはできない。従って、メロディを作るためのヒントを支援し、なおかつ1度作った音楽を蓄積し、何度でも聴くことができ、さらに自由な修正が可能なシステムが必要になる。

ここで、どのようにして潜在的な音楽能力を引き出すか、ということが問題になる。作曲の方法にもいろいろあり、プロの音楽家でも個人によって様々であるらしい。その中で我々はコード進行を基にした作曲に注目した。コード進行は、曲全体の雰囲気を作る役割を持っている。不愉快なコード進行からは、さわやかなメロディは生まれ

ないのである。本研究においては、先に曲全体の雰囲気を生成することは、核部分であるメロディの生成を容易にし、曲全体の構成も整えることができるという仮説に基づき、アマチュアによる作曲を支援するシステムを試作している。

2 システムにとり入れる音楽の原理と構想

2.1 調 (Key) と転調

調とは、例えばハ長調、ト長調、イ短調などのことを指す。現在では、12音階で最低音が主音となり、12種の長調と、12種の短調が確立している。1つの曲を歌手や楽器の音域に合わせて調を変えることを移調という。しかし、聴いている方にとっては高さが変わるだけで、全く同じ曲である。したがって、本システムでは混乱を避けるため、ハ長調を基準としたものを扱う。

曲の途中で調が変わることを転調といい、一時的に他の調に転調するものと、完全に他の調に転調してしまうものに分けられる。本システムでは、前者に対しては、ハ長調、イ短調のいずれかに含まれるものとして扱い、後者に対してハ長調、イ短調で先に作曲を行ない、全体を演奏する段階で、転調できるような機能を付け加える。

2.2 コード (和音)

和音とは、高さの異なる2つ以上の音が同時に響いた時に合成される音をいう。一般に、音楽で使われる和音は、規則的に選ばれた音の集まりである。実際に使われているコードは、3つか4つの音からなる。現在使用されているコードの種類は1つの調について20種類ほどあるが、働きを大別すると、Tonic, Subdominant, Dominantの

3つの機能に集約される。その中でも代表的なものは、それぞれI度和音（主和音）、V度和音（属和音）、IV度和音（下屬和音）である。即ち、とりあえずこの3つの和音さえ知っていれば、曲を作ることができるので、本システムでは、最初この3つのコードのみを扱い、利用者のレベルに合わせてコードを増やすことができるようになってい

2.3 コード進行

先にも述べたように、どんな単純な曲でも2つか3つのコードが使われる。そして、それらのコードを連結していくことをコード進行という。コード進行は、Tonic, Dominant, Subdominantの3つの機能の組合せからなる。それらは、すべて1つ前のコードにのみ依存する。その組合せを次に示す。しかし、実際の曲の中には例外も多くあるため、本システムではこれらの音楽理論を用いた拘束は行なわない。

- Tonic \Leftrightarrow Dominant
- Tonic \Leftrightarrow Subdominant
- Subdominant \Rightarrow Dominant

2.4 展開形

コードにおいて普通は根音が最低音となるが、根音以外が最低音となることもある。そのような形を展開形という。どのような展開形をとっても、その働き、機能は変わらないが、前後のコードにより展開形を変えることによって、その響きが大きく変化する。したがって、本システムでは、利用者が自由に展開形を選択できるようにする。

3 システムの実現

3.1 ハードウェアの構成

本システムのハードウェアの構成では、ワークステーションとシンセサイザーはMIDI (Musical Instrument Digital Interface) ケーブルにより接続されている。MIDIとは、音楽情報をデジタル信号によって伝達するためのインターフェイスである。これにより、コンピュータから鍵盤を押す、離す、音色を変えるといった演奏情報を送ってシンセサイザーをならしたり、シンセサイザーから鍵盤入力した演奏情報をコンピュータで扱うことができる。

3.2 MIP と MMO

MIPとは(Media Interface Prolog)の略であり、我々の研究室で学習支援システムを作成する場合において、1つのシステムでC言語とProlog言語の2つの言語を使用するためのPrologインタプリタである。つまりシステム内部での記号処理(知識処理)にはPrologを、数値計算や利用者とのインターフェイスについてはC言語とXviewを用いる。そこでProlog言語とC言語の間でデータをやりとりするためのインターフェイスが必要となり、設計されたものである。MIPの特徴は、C言語側がオブジェクト指向で設計されていることである。Prologはインタプリタ言語であるため1つの流れがある。その合間でCを呼び出すことになる。

MMOとは、MIPの中で定義されたC言語の中のクラスオブジェクトのことである。

3.3 クラスの構築

本システムのMMOクラスは、図1のようになっている。

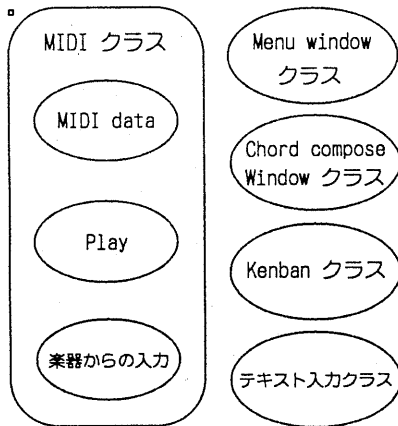


図 1: MMO クラス

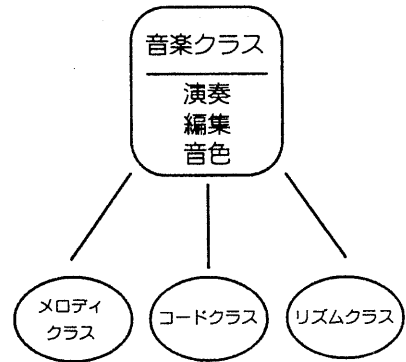


図 3: Prolog におけるクラス

本システムの Prolog 側のクラスは、図 3 に示すような構造になっている。

● 音楽クラス

メロディクラス、コードクラス、リズムクラスを含み、メロディ、コード、リズムの合成、演奏、音色を管理する。

ー メロディクラス

メロディの生成とメロディの登録、メロディの配列（音の並び）を管理する。

ー リズムクラス

リズムの生成、リズムの登録とリズムの配列（音の並び）を管理する。

ー コードクラス

コードクラスは、図 2 のような構造になっている。

3.4 Prolog による音楽知識の記述

Prolog は、物事にあらかじめ成立する事実、何らかの条件を満たすものに共通の性

質や関係を記述した規則を与え、解きたい問題を質問として入力して結果を得る言語である。本システムでは利用者とのインターフェイス部分は C 言語、システム管理部分は MIP、そして知識と知識処理部は Prolog を使用した。特に利用者が入力したコードや、コードモーションの知識を、以前に登録されたものと比較し、同じものや紛らわしいものがあつた時は注意を促し、なければ新しく追加するといった支援を行なえるようにした。更に、利用者がコード名を自由につけることができるので、同じ要素（ピッチ）のコードでも使い方によって分類することができる。

● コードの音（ピッチ）に関する知識

コード名とそれに含まれる音の集まりを記述したもの

（例）コード C は、ドとミとソで成り立つ。

chord('C',[do,mi,so]).

コード Am は、ラとドとミで成り立つ。

chord('Am',[ra,do,mi]).

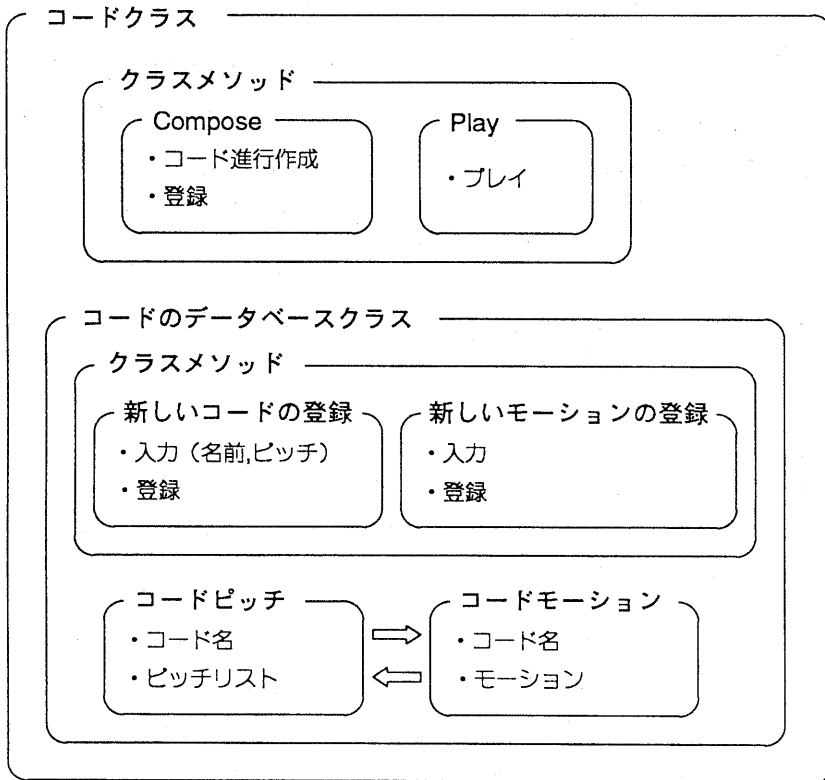


図 2: コードに関して Prolog 内で定義するクラス

- コードのモーションに関する知識
あるコードからの進行可能なコードの集まりを記述したもの
(例) コード C は、コード F とコード G7 に進行可能である。
`chord_motion('C',['F'],'G7').`
コード G7 は、コード C に進行可能である。
`chord_motion('G7',['C']).`

3.5 支援の実際

3.5.1 新しいコードの登録

新しいコードの登録は、利用者が Xview 上のフレームから名前を、鍵盤からそのコ

ードのピッチを入力する。それを以前に入力したものと比較し、コード名とピッチがマッチするものがあれば、登録を中止する。コード名、ピッチのどちらかがマッチした時には警告を出し、利用者の意志で登録の有無を決定する。コード名、ピッチのどちらもマッチしない場合には、そのまま登録を行なう。

3.5.2 新しいモーションの登録

最初に今まで登録したすべてのコードを表示する。利用者が 1 番目のコードと 2 番目のコードを選択する。コードの登録と同じようなチェック、登録を行なう。

3.5.3 コード進行の生成

最初に今まで登録したコードすべてを表示する。利用者が1番目のコードを選択したところで、そのコードに進行可能なコードを表示する。更に利用者が2番目のコードを選択したところで、このコードに進行可能なコードを表示する。この繰り返しによって、コード進行を生成する。この時、展開形も選択できるようにする。

3.5.4 演奏

演奏は、Xview上の鍵盤の色を反転させることによって視覚的にも訴えられるようにした。コードやコードモーションの登録時でも、随時それらの音を聴かせることができ、音の位置は鍵盤上に表示させる。

4 おわりに

本システムでは現在のところ、コードの支援しか取り組んでいない。また一つの調しか対応できない欠点がある。今後の課題としては、メロディ、リズムを含めた支援が必要である。特にメロディにおいては、楽器からのリアルタイム入力および、高さと時間軸を持った2次元表示が必要になってくると思われる。さらにリズムを組み合わせたコード進行も必要になってくるだろう。それはコード進行にリズムをのせることによって、一つのイメージにこだわらず、いろいろな雰囲気曲が作れるようになると考えられるからである。なぜなら、リズムもまた曲の雰囲気を作り出す大切な要素の一つであるからである。また、いくつかの調を利用者に意識させながら扱えるようにする必要があるのであろう。

参考文献

- [1] 竹内剛, 菅野真子: “新ハーモニー入門”, 財団法人ヤマハ音楽振興会, (1982).
- [2] 藤田進: “作詞・作曲入門ゼミ”, 自由現代社, (1993).
- [3] 若松正司: “若松正司の音楽セミナー／コードのしくみ”, 音楽之友社, (1988).
- [4] 若松正司: “若松正司の音楽セミナー／コードの使い方1”, 音楽之友社, (1989).
- [5] 小林一夫: “ちょっと便利なコードの知識”, 中央アート出版社, (1990).
- [6] 平野健太郎: “MIP 入門”, (1993).
- [7] 清洲美紀, 伊丹誠, 伊藤紘二: “音楽教育における作編曲を支援するシステム”, 電子通信学会技術研究報告 ET91-34, (1991).
- [8] 穴田啓樹: “音楽教育における作曲を支援するシステム”, 東京理科大学基礎工学部卒業論文, (1992).