

隠れマルコフモデルを用いた曲構造を持つ 音符列の生成

耿霽 池田剛 乾伸雄 小谷善行
(東京農工大学 工学部 電子情報工学専攻)

本研究の目的は、自然で多様な音符列を生成することである。そのために、音符の隣接情報と曲の構造に着目する。音符の隣接情報は音高と音価に分けて考える。音高は隣接情報によって構成されたマルコフ過程を使って生成する。音価はマルコフ過程で表現すると不自然な組み合わせが生成されるため、音価終了(開始)時間を隠れ状態とした隠れマルコフモデルを用いる。そして、マルコフモデルでは曲の構造を生成することができないという問題を解決するために、MFSと呼ぶ柔軟な曲構造を提案する。MFSに音符の隣接情報を適用することで自然な音符列を生成できることが実験結果からわかった。

Musical Composition with Musical Structure Using Hidden Markov Model

This research aims to generate natural and various sequences of notes for an automatic composition system. We focus on information between adjacent of notes and musical structure in this paper. Our system generates pitches and durations of notes independently. We assume that a sequence of pitches can be Markov process composed of adjacent pitch information and a sequence of durations can be Hidden Markov process such that note ending (starting) time is a hidden state to avoid generating unnatural sequences of durations. In addition, we propose a flexible musical structure, called MFS, to resolve a weakness of the (hidden) Markov model against the structural expression. Finally our system can generate a sequence of notes by adjacent information of notes with restrictions of MFS acquired from musical database. Preliminary experiments show that our method is effective for our aim.

1. はじめに

自動作曲へマルコフ遷移を最初に利用したのは、1957年 Lefaren Hiller と L. Isacson であり、和声学や対位法の規則を使った「Illiac Suite」という弦楽4重奏曲を生み出した。この研究は最初の自動作曲の研究でもある。自動作曲の研究でマルコフ遷移がこれほど好まれるかということ、マルコフ遷移は楽曲情報を確率統計的に処理しやすく、モデル化が容易のためと考えられる。従来マルコフ遷移を音楽へ適用した研究の多くは、音高列の生成だけを主として扱っている(参考文献の[1]~[5]のいずれもマ

ルコフ遷移を用いた音楽研究であるが、それぞれ音高だけと明記してある)および自動作曲システムは主題のある反復的な構造を生成できないという欠点があるので、中心的な自動作曲の手段として使われなくなった。本研究では、隠れマルコフモデルを用いる音価列の生成方法を提案し、それを曲構造へ適応する。

2. 音符のマルコフ遷移

マルコフ遷移を実現するために、遷移確率情報を獲得することが必要となる。本章では、本

研究における音高の確率情報と音価の確率情報を獲得する方法について述べる。

2.1 音符列の生成方法

同じ要素数の音高列と音価列が独立して生成することで音符列を生成する。生成する際、音高列は確率遷移情報から確率的選択によって生成する。

生成される音高列の確率は式 1 で示すように近似する。

$$P_{(note_1, note_2, \dots, note_n)} \approx P(note_1) \prod_{i=2}^n P(note_i | note_{i-1}) \quad (式 1)$$

音価列は、各音価のひとつ前の音価の終了位置(次の音価の始まる位置)を隠れ状態にすることで、隠れマルコフモデルによって生成することができる。生成される音価列の確率は、各音価が小節内における終了位置(endtime)によって、式 2 で示すように決定される。

$$P_{(time_1, time_2, \dots, time_n)} = P(Time_1) \prod_{i=2}^n P(time_i | endtime_{i-1}) \quad (式 2)$$

2.2 音高確率情報の獲得

音高の確率遷移情報を利用し自動作曲する研究は 60 年代からあった。参考文献の[1] [2] [3] [5]のいずれもそのような研究の例である。本研究では、音高が与えられたときに、確率的選択によって、その次の音高を決める。この方法で音高列を生成することによって、音楽の規則を利用しなくても不自然な音高のつながりを避け、多様な音符列を生成することができる。音高の確率遷移情報は、学習対象曲から得られる音高遷移の頻度によって計算される。たとえば、表 1 のような確率遷移情報を獲得することができる。

表 1 生成した音高の頻度情報

学習対象となる音高列：

{69, 71, 71, 78, 73, 73, 78, 73}の場合

Note1:	69	Note1TotalNum:	1
Note2:	71	Note2Num:	1
Note1:	71	Note1TotalNum:	2
Note2:	71	Note2Num:	1
Note2:	78	Note2Num:	1
Note1:	78	Note1TotalNum:	2
Note2:	73	Note2Num:	2
Note1:	73	Note1TotalNum:	2
Note2:	78	Note2Num:	2

注：網掛けの部分は、前の音高、網掛けのない部分はその音高につなげる音の種類と頻度である。Note1TotalNum は前の音高が現れる総回数である。

表 1 によって、Note_i が生成される遷移確率は

$$P(note_i) = \frac{Note_i Num}{Note_{i-1} TotalNum}$$

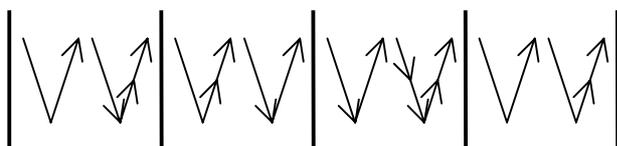
である。

2.3 隠れ状態による音価確率情報の獲得

前述したように、音符の隣接情報を利用する自動作曲の従来研究のほとんどは、音符ではなく、音高だけを操作である。音価については、パターンの遷移を利用することが多い。パターンを利用する場合、音価パターンの長さを決定する問題が存在する。[2]は小節単位の音価パターンを利用する。[4]の音価は、全音符一個から、四分音符 4 個の並びまでのパターンを利用する。音価パターンの遷移によって音価列を生成する方法の欠点は、パターンが長すぎると曲の自由度が限定されてしまう、短すぎると長い音符に対応できないということである。たとえば 1 拍ずつのパターンを取ると、1 拍以上の音価に対応できない。もっともよく使われるのは小節単位であるが、このときはタイの処理が複雑になる。そこで本稿は、これらの問題を解決するために、隠れ状態を利用してマルコフ遷移を用いた生成手法を取り入れることにした。音価も音高と同じような手法で生成できれば、楽曲生成全体のアルゴリズムが簡単になり、パターンを利用する必要もなくなる。

まず音価の遷移情報を利用する問題について考える。たとえばある自然な 4 小節の楽曲があるとすると、図 1 はその楽曲の音価の進行が拍

子であらわしたものである(上下に1回の往復は1拍子とする)。



注：矢印は音価の進行方向と終了位置を示す

図1 曲の中にある4小節の音価進行

図1にある音価を、表1と同様に遷移情報として獲得すると、図2に示した音価列を生成することが可能である。この図では{0.25, 1.0, 0.5, 0.5, 0.25, 0.75, 0.25, 0.5}(1拍は1とする)となる音価の遷移を表している。しかしながら、このような音価列は楽譜から見あたらないような不自然なものである。

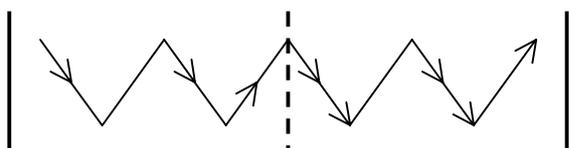


図2 表1で示す音価の確率遷移情報によって生成した音価列

本研究で用いる隠れ状態とは、前の音価と次の音価の遷移確率を利用するのではなく、前の音価のその小節内における終わる位置(次の音価の始まる位置)と次の音価の長さで音価の確率情報を獲得する方法である。この場合、音価 $Time_i$ を生成される確率は次式によって示すことができる。

$$P(Time_i) = \frac{Time_i Num}{EndTime_{i-1} TotalNum}$$

$EndTime_i TotalNum$: 小節内におけるある位置で終わるすべての音価の数

$Time_i Num$: その位置で始まる $Time_i$ の数

このような隠れマルコフモデルを使うことで、図2のような不自然な音価列の生成を避けることができる。

3. 予備実験

提案した隠れマルコフモデルを用いた音価列の生成結果を確認するために、2章で述べた

隠れマルコフモデルを用いた音価列の生成実験をおこなった。

3.1 実験方法

次の二通りの実験をおこなった。

- 中国少数民族曲一曲より獲得した音高、音価の確率遷移情報を用いて音価列を生成する実験
- 中国少数民族曲10曲より獲得した音高、音価の確率遷移情報を用いて音価列を生成する実験

3.2 実験結果

《阿里山的姑娘》(図3)という中国少数民族曲から獲得した確率遷移情報を用いて生成した音価列を図4に示す。



図3 《阿里山的姑娘》の楽譜



図4 生成した音価列1

中国少数民族曲10曲から獲得した確率遷移情報を用いて生成した音価列は図5に示す。



図5 生成した音価列2

3.2 考察

図3と図4の実行結果を実際に聞いてみると、次のことを感じる。

音高については、音高の隣接情報の拘束によって、変に飛び上がることがない。音価については、

本研究で提案した音価の確率情報の拘束によって、リズム的に非常に自然に聞こえて、実際の楽譜でありえないような音価が生成されない。

また、一曲を教師値にした実験では、教師値とまったく違う音符列を生成されながらも、もとの曲と比較しながら聞くと、元の曲の風味を大きく残していることがわかる。学習に用いた曲の風味が残された曲の生成に適用できると考えられる。

10 曲を教師曲にして生成した音符列においては、非常に多様化をしているため、教師曲のどれも似ていないことがわかる。

以上のことから、音符列に音符列に構造を持たせることで、全体にまとまりのある多様で自然な音符列を生成できることが期待できると考えられる。

4 . 曲の構造 (MFS)

構造を持つマルコフ遷移の音符列を生成するために、自然で多様な曲を生成できる曲構造を定義し、音符列を生成する実験をおこなった。この章では曲構造を定義する。

自然な楽曲はそれぞれ主題を持っている。人間に曲の主題感を持たせることは、音符の反復によって実現できると考える。ある音符の断片は同じ小節のどこかで繰り返されたり、次の小節で繰り返されたり、または、かなり離れているところで繰り返されたりすることもある。その上、この繰り返し操作は、音高に対して行われたり、音価に対して行われたりもする。このようなことから、自然な音符列は、隣接音符の拘束だけではなく、構造を持つ必要がある。本章では柔軟な(多様で複雑な楽曲の構造を表現することができる)曲構造(MFS)を生成する方法について説明する。

4.1 MFS とは

前述したとおり、自然な楽曲フレーズはみんな構造を持っている。ところが曲の構造はまちまちであるため、構造を生成するシステムは次のことが要求される。

- 複雑な曲構造の定義ができる
- 多様な曲構造が生成できる

一つの楽曲フレーズの一部を、違う場所で再現する場合、変奏することが多い。この変奏は、音価的な変奏であったり、音高的な変奏であったり、または部分的な変奏であったりし、非常に複雑である。複雑な反復を定義するために、いろいろな手法が使われてきている。その中

で、本研究では、階層的構造を提案し、曲構造を定義する。階層的な構造を用いることで、少ない反復タイプ(全 6 種類だけ)によって複雑な反復構造を実現することができる。

一つの自動作曲システムで多様な曲構造を生成するために、確率的選択を用いる。これによって、多様な構造が生成されることができる。確率的選択を用いて生成される階層的な曲構造を MFS (Music Phrase Structure) と呼ぶ。MFS についての具体的な説明は次節で述べる。

4.2 MFS の定義

前述したとおり、本稿は階層的構造を用いて MFS を定義する。本節では、MFS の定義について述べる。

一つの音符列の構造は複数の層によって表現される。便宜上 1 層目を FL、2 層目を SL、3 層目を TL、4 層目を OL などと表示する。各層は、次のように、部品 (P) によって構成される。

$$1 \text{ 層} : FL = \{P_1^{FL}\}$$

$$2 \text{ 層} : SL = \{p_1^{SL}, p_2^{SL}, p_3^{SL}, \dots, p_n^{SL}\}$$

$$3 \text{ 層} : TL = \{p_1^{TL}, p_2^{TL}, p_3^{TL}, \dots, p_k^{TL}\}$$

...

音符列

それぞれの P は部品であり、P は次式のように自分の番号 (SelfNum)、自分と関連する部品番号 (RepeatNum)、下の部品の数 (LowerNum)、関係タイプ (RepeatType) との 4 つの成分を持つ。

$$P_i = \begin{bmatrix} SN \\ RN \\ LN \\ RT \end{bmatrix} \quad \begin{matrix} SN = i \\ RT \in \{\text{Re TypeA}, \text{Re TypeF}, \\ \text{Re TypeB}, \text{Re TypeL}, \\ \text{Re TypeS}, \text{Re TypeN}\} \end{matrix}$$

ただし、

ReTypeA: 音符の完全反復
 ReTypeF: 音符の前半相關
 ReTypeB: 音符の後半相關
 ReTypeL: 音符の音価反復
 ReTypeS: 音符のシフト反復
 ReTypeN: 音符の関係なし

FL, SL, TL, OL によって、具体的な音符列の構造を定義する。ただし FL から下の層に移る過程で、定義される反復型の範囲は徐々に狭くなっていく。たとえば、FL では、音符列の一番大まかな反復構造に満たすための構造単位を定義する。SL は大きな構造単位で、音楽の形式、たとえば AA や ABA など定義することを期待している。TL では A や B になるフレーズ内での変奏部分を定義する機能を持つ。

図 6 は生成される MFS の一つの例である。

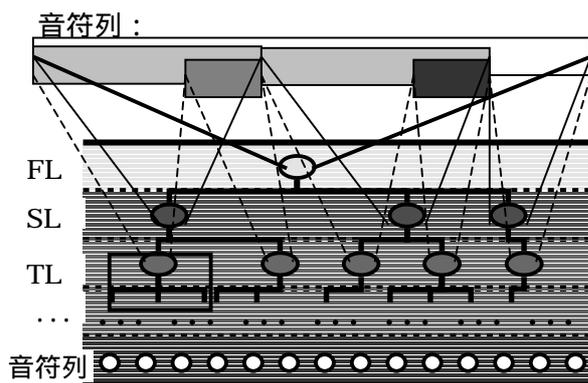


図 6 MFS によって生成する曲構造

注: \square は全音符列である。その中に \square の部分が反復する、反復するときに \square の部分が変奏し、違う音符列 \square になった。図に示す \square で囲んだ部分(丸と丸から出す線)は部品に相当する。部品はそれぞれ SN(自分の番号)、RN(関係部品の番号) LN(下位部品数)と RT(反復タイプ)の四つの成分によって構成される。

部品 P の各成分は必要に応じて、意味を持たないものもありえる。たとえば一番下位層の部品は LN(下の部品の数)が存在しないのでその部品の LN は意味を持たないことになる。

MFS を用いて、曲の構造を記述する。MFS を生成するときに、まず一つの部品しかもっていない FL から生成する。その後、部品の定義によってトップダウンで逐次に下の層まで生成していく操作をおこなう。

5 . MFS を持つ音符列の生成

この章において、4 つの層を持つ MFS に、音符の隣接情報を適用する方法について説明する。

5.1 OL 部品と音符列

前述したように、音符列の一番細かい構造は、MFS の OL 部品列によって表される。MFS の定義によって、次のような OL を生成される。

$$OL = (P_1, P_2, P_3, \dots, P_n)$$

OL 部品の P_i はそれぞれ、自分の部品番号 (SN)、関係部品番号 (RN)、音符の関係タイプ (RT) によって、式 3 のように表記される(意味を持たない LN 成分は表記しないことにする)。

$$P_i = \begin{bmatrix} i \\ RN_i \\ RT_i \end{bmatrix} \quad RT_i \in \{\text{ReTypeA}, \text{ReTypeL}, \text{ReTypeS}, \text{ReTypeN}\} \quad (\text{式 3})$$

ただし、

- ReTypeA: 音符の完全反復
- ReTypeL: 音符の音価反復
- ReTypeS: 音符のシフト反復
- ReTypeN: 音符の関係なし

ここで生成される OL 部品は MFS から音符列へつながる重要な情報列である。一方 OL 部品は MFS の一部であり、他の部品と同様な構造を持っているので、曲構造全体の情報を OL 部品を用いて表現することが可能である。OL 部品の末端記号は単音音符であるので、部品と音符は一对一の関係であり、部品列は音符列の音符関係列とみなされることになる。末端音符を生成するときは、OL 部品で示す MFS 規則にしたがって生成されなければならない。

5.2 音高列の生成

基本的には音高は第 2 章で説明した確率遷移情報を用いる方法で生成する。MFS の規則と矛盾が生じるときに、曲構造が優先される。生成する目的音高列を $H = \{h_1, h_2, \dots, h_n\}$ とした場合、 h_i は TR_i^{Pitch} の値によって次のどちらかの方法で生成される。

- $RT_i^{\text{Pitch}} = \text{ReTypeA}$ または $TR_i^{\text{Pitch}} = \text{ReTypeS}$ のときに、MFS の規則にしたがって生成される
- $RT_i^{\text{Pitch}} = \text{ReTypeN}$ のとき、確率遷移情報を用いる方法で生成される

5.3 音価列の生成

MFS の規則を守りながら、隠れマルコフモデルを用いる音価列の生成をおこなうのは、音高列の生成よりも複雑であるが、再帰によって実現される(図 7)。

```
if(音価の数 = OL 部品数) return(1)
if(RN=音価の反復)
  MFS によって音価(Time1)を生成する
  音価の遷移情報から Time1 の情報を探索
  ある場合：
    if( 音符列生成成功(再帰値=1))
      return(1)
if(RN=音価関係なし)
  遷移情報による音価(Time2)を生成する
  if(音符列生成成功(再帰値=1)) return(1)
  else Time2 を削除する
return(0)
```

図 7 音価列を生成するアルゴリズム

6 . 音符列の生成実験

6.1 実験目的

曲構造を持つ遷移情報を適用する音符列の生成実験をおこなった。生成した音符列について評価する。

6.2 生成された曲の例

図 8 と図 9 は生成した音符列である。



図 8 生成した音符列 3



図 9 生成した音符列 4

6.3 考察

図 8 で示す音符列において、1 小節目は 2、4、5 小節 3 回も再現されるが、それぞれ違う変奏の仕方でも再現されることがわかる。図 9 で示す音符列は、3 小節、4 小節は 3 回も変奏をしながら反復して進行していく曲が生成されたことがわかる。どちらも人が作ったフレーズに近く、自然である。

7 . 終わりに

本稿は隠れマルコフモデルによる音価列の生成方法を提案し、構造を持つ自然な音符列を生成することに成功した。

将来の課題として、もっと長い音符列の生成と、自然に終了する方法について研究を進めていきたい。

参考文献

- [1] 音楽分析学会：「音と思索」，「統計的音楽分析()」-ドイツの子供の歌-，(株)音楽の友社，1969 年 12 月 25 日，P89-107
- [2] 大槻説乎：「数理科学」電子計算機による作曲 -メロディー編-，1963 年 7 月，P56-59
- [3] 水谷一郎：音楽におけるマイクロ・コンピュータ導入の可能性について，「大阪音楽大学研究紀要第 19 号」，1980 年，P65-187
- [4] 関根智明、林大雅：電子計算機による作曲，「数理科学」，1963 年 8 月，P58-60
- [5] 徳丸吉彦：日本民謡の統計的分析，「国立音楽大学大学院研究年報」，1975 年，P25-56