

パーピープン: 誰でもどこでもインタラクティブに使える 知的ジャズ和音生成システム

平田 圭二

NTT コミュニケーション科学基礎研究所

hirata@bri1.ntt.co.jp

青柳 龍也

津田塾大学

aoyagi@tsuda.ac.jp

本稿では、現在開発中の音楽システム「パーピープン」の設計方針、動作機構、実装等について述べる。パーピープンは、単純な和音進行が与えられると、ケーデンス単位で和声的文脈を考慮しながら誰々風の質の高い和音演奏を自動生成する(リハーモナイズ)システムである。Java アプレットとして実装した GUI により高い利用性も実現できた。

Pa-Pi-Pun: intelligent jazz harmony creation system that can be interactively used by anyone anywhere

Keiji Hirata

NTT Communication Science Laboratories

Tatsuya Aoyagi

Tsuda College

This paper presents the design principle, the system organization and the implementation of a musical system “Pa-Pi-Pun” that we are developing. Pa-Pi-Pun first takes a simple chord progression as an input and then generates a chord performance of high quality a la someone (reharmonize) with taking into account the harmony context on a cadence basis. The GUI implemented as a Java applet achieves high accessibility.

1 はじめに

我々は、音楽知識表現の手法とその応用について研究を進めている。音楽システム「パーピープン」は、ハービー君 [1] 同様、単純な和音進行が与えられ、誰々風の和音演奏を自動生成(リハーモナイズ)するシステムである。ハービー君は、個々の和音単位で演奏を生成していたのに対し、パーピープンはケーデンス単位で和声的文脈を考慮しながら演奏を生成する。これより、パーピープンはハービー君より、質の高い演奏を出力することが可能となった。また、GUI を Java アプレットとして実装したので、利用性が格段に向上した。

本稿では、現在開発中のパーピープンの設計方針、動作機構、実装等について述べる。

2 パーピープンの設計

2.1 目標

我々の研究目標は、質の高い音楽を生成するシステムの構成法を探ることである。

ここで「質の高い音楽」に着目したのは、音楽を対象としたことで初めて探求可能になる題材の 1 つが、まさに音楽的な質の高さだからである。曲の聴取や演奏など、音楽に関連する(特に高次の)振る舞いや概念に関して、普遍的に正しいことは殆んどないと言って良いだろう。音楽的な質の高さも、主観的かつ曖昧な概念である。

例えば、複数のユーザはある 1 つの曲を様々な聴くであろうし、1 人のユーザでも状況に応じて同じ曲の聴き方は変化するであろう。あるユーザにとっては好ましい音楽でも、別のユーザにとってはそうでないという状況は頻繁に起こる。逆に、1 人だけしか音楽的に価値があると認めないような音楽でも、その人にとっては質の高い音楽と言える。

一般に、多くの人を質の高い音楽で満足させるのは、機械ならずとも人間でさえ非常に難しい作業である。従来のシステムの多くは、より多くの人の趣向に沿う出力を生成することを目標としているが、現時点では、100% のユーザが常に満足するような出力を得るのは非常に困難であろう。この立場は、多くの人を平均的に満足

させるアプローチ (A) と換言できる。

一方、限られたユーザの趣向や嗜好に特化し、そのユーザだけより満足させる、即ちそのユーザだけにとってより質の高い音楽を提供するアプローチ (B) もある。

この 2 つのアプローチは、システムの自動化率とユーザの趣向や選好に対する満足度のトレードオフと換言することもできる。つまり、アプローチ A では、多くの人が平均的に満足するような解をシステムが自動的に計算する。この時、ユーザは解の計算過程にあまり関与しない。アプローチ B では、ユーザは解の計算過程を積極的に制御し (customize し)、そのユーザの趣向や選好により合致した解を得ることができる。

一般に、自動化率、操作の簡便性という点では A の方が有利であるが、B は個々のユーザにとってより質の高い音楽を生成することができる。B において操作の簡便性を向上させるためには、システム制御を適切に抽象化してユーザに見せることが重要である。我々は、質の高い音楽生成を重視し、アプローチ B を選択する。

2.2 仮定

我々はパーピープンの作成にあたり、以下の仮定を置いた。

- すでに存在している演奏が音楽的に妥当な演奏である。
- 元演奏をケーデンス¹単位まで分解しても、元演奏の雰囲気は保持される。

音楽的に妥当な演奏を生成するためには、実際の演奏に含まれているような音楽的振る舞いを再現しなくてはならない。しかし、そのような高次の音楽的振る舞いや概念は、主観的、暗黙的かつ曖昧であり、普遍的に正しいものは殆んどない。教科書に記述されている事柄や、専門家から聞きとった知識を表現したルールは、確かに、ある限定された状況でのある一面を表現している。しかし、高次の音楽的振る舞いや概念に関する情報すべてを表現している訳ではない。従って我々は、ルールを介さず、すでに存在している演奏そのものから解を生成するアプローチ (事例ベース推論) を採用した。

¹終止感や段落感を与えるような特有の和声構造。ジャズやポップスの場合、通常、数個の和音列から成る。

元の演奏全体には、もちろん、その演奏者の特徴や曲の特徴が反映されている。しかし元の演奏をどんどん分解して行き、1 つの音まで分解したとすると、この 1 つの音には元の演奏の特徴などは反映されていない。つまり、分解する途中で元の演奏にあった特徴が失われて行ったと考えられる。では、どのレベルまでの分解なら元の演奏にある特徴が保持されるのであろうか。音楽的には様々な考え方ができるが、我々はケーデンスを元の演奏全体の特徴を保持した最小単位と仮定した。本稿では個々のケーデンスを signature と呼ぶ。

2.3 方針

我々はパーピープンの内部設計にあたり、以下の方針を立てた。

- 決定的に動作する処理モデルを用いる
- ユーザに高い制御性を与える
- 新しい演奏の生成は、音楽的文脈を考慮した signature の再組合せ (recombinacy) で実現する

順に説明を加える。

音楽学や音楽認知科学等の研究分野では、人間の音楽的な振る舞いや概念を、決定論的因果関係の枠組の中で理解、解釈するということが共通認識となっている。これは、音楽的な振る舞いや概念に確率的要素は少ないと考えることに基づいている。我々もこの考え方に従い、処理モデルは可能な限り決定的に動作し、確率的な振る舞いは極力排除するよう設計する。

実際の音楽ではその処理プロセスが未解明な部分が多いため、音楽システムを作成する際、その未解明部分を確率モデルとして実現したり、適用範囲の狭い ad hoc なアルゴリズムで実現することが多かった。しかし、我々の目標に照らして考えると、確率モデルや ad hoc なアルゴリズムでは、システムの動作をユーザの直観通りに制御することは難しく、ユーザの意図を反映した質の高い演奏の生成には不向きである²。

我々は、音楽的文脈を考慮した signature の再組合せにより、新しい演奏を生成する。

²逆に芸術的な観点から、確率モデルを用いたり ad hoc な制約を課して積極的に予測できない (制御できない) 出力を得る立場もある。

仮定から、signature は妥当な演奏を構成する部品であり、元の演奏の特徴を保持している。従って、signature を再組合せすれば、新しい演奏にも元の演奏の特徴が反映される筈である。そして、和声的文脈やボイスリーディング(隣接した和音の各構成音間の対応関係)を考慮しユーザの意図を反映させつつ signature を再組合せすることで、質の高い音楽生成が期待される。

また、和声的文脈とボイスリーディングを考慮することは、途中解の生成を絞り込むことにつながり、システムはより決定的に動作することができる。

3 音楽的概念や関係の表現

パーピーブンで扱う音楽的概念や関係には、1つの音、和音(1つの音の集合)、和音の列(ケーデンス)、ケーデンスから成る木構造(和声的文脈)、ボイスリーディングなどがある。演繹オブジェクト指向データベース(DOOD)の枠組を用いると、これら抽象度の異なる概念や関係を全てオブジェクト項と包摂関係で統一かつ明示的に表現、操作(計算)することができる[2]。音楽的な抽象/具体関係はオブジェクト項間の包摂関係に対応つけられる。これにより、ユーザの直観に合った和声的文脈を考慮した signature の再組合せや最もスムーズなボイスリーディングの選択が可能となった。

3.1 一つの音と和音の表現

今、C5, C, 単なる1音という3種類の音を考える。これらを DOOD では以下のようなオブジェクト項で表現する。

| | |
|-------|-------------------------|
| 単なる1音 | note |
| C | note(pitch=C) |
| C5 | note(pitch=C, octave=5) |

ここで、note は基本オブジェクト項、pitch = C は属性、さらに pitch は属性ラベル、C は属性値と呼ばれる。

この時、C5, C, 単なる1音というオブジェクトは、種類は同じだがわずかに異なっており、C5, C, 単なる1音という順に抽象的になって行くと考えるのは自然である。

演繹規則によって、オブジェクト項間に包摂関係(\sqsubseteq)を定義することができる。包摂関係と

は直観的には、「具体的なオブジェクト \sqsubseteq 抽象的なオブジェクト」、「特殊 \sqsubseteq 一般」を意味する。この例で $\text{note}(\text{pitch} = C, \text{octave} = 5)$ と $\text{note}(\text{pitch} = C)$ を比較すると、 $\text{note}(\text{pitch} = C, \text{octave} = 5)$ の方が $\text{octave} = 5$ という属性の分だけ記述が具体的なので、 $\text{note}(\text{pitch} = C, \text{octave} = 5) \sqsubseteq \text{note}(\text{pitch} = C)$ となる。同様に $\text{note}(\text{pitch} = C) \sqsubseteq \text{note}$ となる。

和音には root (根音) と和音名という2つの属性を考える。例えば G_7 という和音は、 $\text{chord}(\text{root} = G, \text{name} = 7)$ と表現される。

3.2 ケーデンスの表現

例えば、 $\text{II}_{m7} - \text{V}_7 - \text{I}_{M7}$ という和音進行(ケーデンス)の作る和声的文脈は以下のようなオブジェクト項で表現され、これが前述の signature に対応する。

```
ctxt(prom = IM7,
      seq = ctxt(prom = IIm7)
          → ctxt(prom = V7)
          → ctxt(prom = IM7),
      d = 0)
```

ここで、ctxt は和声文脈を表現するための基本オブジェクト項である。seq という属性は、ある和声文脈を構成する実際の和音進行を表す。prom という属性は、この和音進行に含まれる和音の中で最も重要で代表的 (prominent) な和音を表す。通常、1曲の中には複数のケーデンスが出現し、そのケーデンスどうしの間にも関係が見いだせる。ケーデンス木とはケーデンスを節や葉とする木構造であり、曲全体の流れを表す。seq の属性値中に現れる和音が ctxt の形になっているのは、ケーデンス木を構成するためである。個々のケーデンスは固有の key を持ち、ここでは、ケーデンス木中の直上のケーデンスとの key の差を d という属性によって相対的に表している。これにより、ケーデンス木の部分木だけ取り出しても音楽的に意味を持つことができる。

パーピーブンでは、 I_{M7} , II_{m7} , V_7 は前節で述べたようなオブジェクト項として、 $A \rightarrow B$ は $\text{arrow}(\text{src} = A, \text{dst} = B)$ というオブジェクト項として表現されているが、紙面の都合上略記している。

Generative Theory of Tonal Music[4]の Time-Span Reduction (TSR) では、曲のある一部分

をグルーピングし、その中の prominent な音でそのグループを代表させ、階層構造を作り曲を解釈する。ケーデンス木は、この TSR の考え方を和音進行に応用したものと見なすこともできる。

例として、 $E^b_{M7}-B^b_{m7}-E^b_7-A^b_{M7}$ という和音進行のケーデンス木を図 1 に示す。この和音進

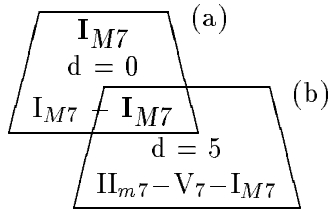


図 1: ケーデンス木の例

行は、key E^b で $E^b_{M7}-A^b_{M7}$ という上位のケーデンス (a) と key A^b で $B^b_{m7}-E^b_7-A^b_{M7}$ という下位のケーデンス (b) に分解される。(a) の prominent 和音は E^b_{M7} (I_{M7}) であり、(b) のそれは A^b_{M7} (I_{M7}) である。(b) 中の $d = 5$ は、(a) の key E^b に対する (b) の key A^b の音程差が半音 5 個分 (4 度) であることを表わしている。

3.3 ボイシングの表現

ある和音やケーデンスに対して、実際に演奏される音をボイシングと呼ぶ。ボイシングは、以下のように非固有属性 (オブジェクトの識別に影響を及ぼさない属性) を用いて表現される。

```
ctxt(prom = chord(...), seq = ..., d = N)
  /(rehar={play(voicing = ...),
        play(voicing = ...)})
```

ここで、‘/’ の左側は $ctxt(\dots)$ というケーデンスを表すオブジェクト項であり、オブジェクトの識別子として機能する。‘/’ の右側に非固有属性を記述する。異なるケーデンスに対して同じボイシングが現われる場合があるので、ボイシングでケーデンスを識別することはできない。上のオブジェクト項では、rehar 属性の属性値が様々なボイシング (play オブジェクト) の集合になっている。さらに、voicing 属性の属性値として実際に演奏された音が現われる。

リハーモナイズの問い合わせを行う時は、まず、あるケーデンス木を文脈として与え、リハーモナ

イズしたい和音進行に対応するその中の部分木を指定する。例として、図 1 の (a), (b) がケーデンス木全体つまり文脈とすると、例えば、部分木 (b) がリハーモナイズの対象となる。

4 システム構成と動作

4.1 全体構成

パーピープンのシステム構成は図 2 のようになっている。図中、Context Editor (4.2 節)

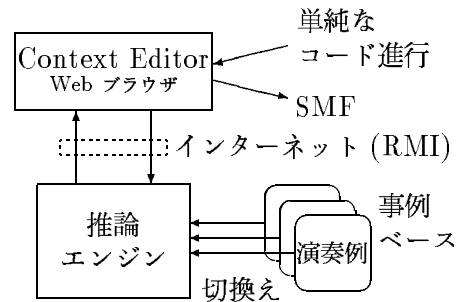


図 2: パーピープンのシステム構成

は、Web ブラウザから起動される Java アプレットとして実装した GUI であり、問い合わせを作成、発行したり、Standard MIDI File (SMF) を演奏したりする。推論エンジン部は KLIC[3] で実装されており、Context Editor で作成した問い合わせを受けて事例ベース推論を行い、その結果を SMF に変換する。

Context Editor は、推論エンジン部を Java の remote method invocation で呼び出す。

次に、パーピープンの大まかな解生成手順を説明する。まず、オフラインで事例ベースの作成を行う。

1. 誰々風の演奏を生成したい場合、その人の実際の演奏を集める。
2. その実際の演奏を分析し、ケーデンス単位 (signature) まで分解する。
3. 実際に弾かれた音符や和音の発音タイミングの近いものどうしをグルーピングし和音の列にまとめる (クオンタイズ)。
4. 上の 3. で作成した和音列を、和音進行の各ケーデンスに対する voicing 属性として付与する

事例ベースには、実際の演奏に現われる全ての signature を格納する。

オンラインで事例ベースのロード、問い合わせの作成、解の生成を行う。

1. ユーザが指定した事例ベース (一般には複数) を推論エンジンにロードする。
2. ユーザは Context Editor を用いて入力の未知曲の単純な和音進行をケーデンス木で表現する。
3. Context Editor で作成したケーデンス木に対し、リハーモナイズしたい部分木を指定する。問い合わせの一般形は、ケーデンス木とその部分木指定のペアの列から成る。
4. 問い合わせのケーデンス木に最も類似したケーデンス木を事例ベース中から検索する。
5. 検索結果のケーデンス木 (一般には複数) から、問い合わせで指定された部分木を抜き出し、その voicing 属性を取り出す。
6. 最も滑らかに接続できるボイスングを選んで順次つなげて行く (ボイスリーディング)。
7. 最終的な解を SMF として出力し、その URL を Context Editor に送る。

4.2 ユーザインタフェース

Context Editor のウィンドウを図3に示す。図

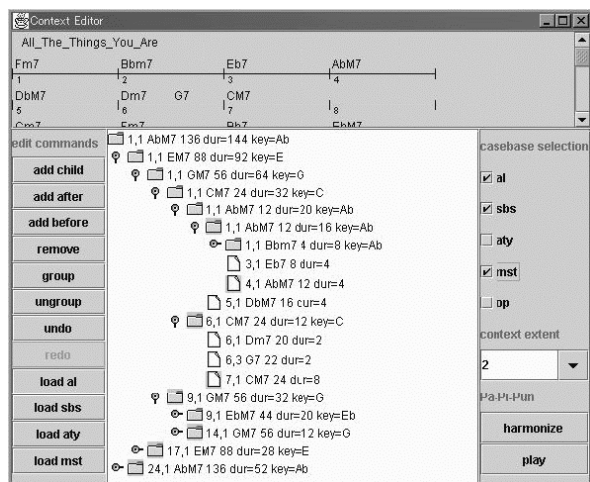


図3: Context Editor ウィンドウ

中央の白い部分には、ケーデンス木がその部分木に対応した段下げとともに pretty print される。ユーザの指示で、部分木の表示を畳んだり展開したりでき、各行先頭の鍵アイコンでその表示状態が分かる。

ウィンドウ左側には、ケーデンス木の操作コマンド (add child, undo など) と事例ベース

のロードボタン (load al, load sbs など) がある。この操作コマンドを用いて問い合わせのケーデンス木を自由に編集できる。ウィンドウ右側には、事例ベースの選択ボタン (al, sbs など)、和声文脈の深さ指定 (context extent, 後述)、harmonize ボタン、play ボタンがある。

4.3 推論エンジン

文脈の検索 例を用いて説明する。事例ベースに $C_{m7}-F_7-B_{M7}^b-E_{M7}^b-A_{m7}-D_7-G_7$ という和音進行の模範演奏があり、推論エンジン中に図4のようなケーデンス木としてロードされたとする。

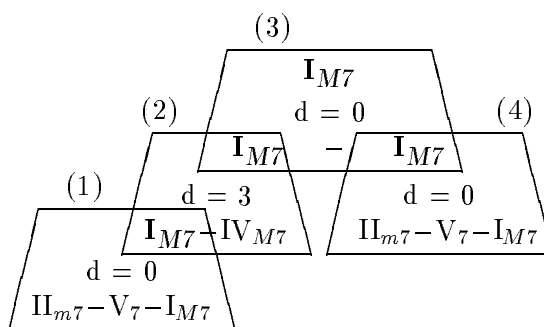


図4: 推論エンジンにロードされたケーデンス木

ここで、図5に示す2つの問い合わせ Q_1 , Q_2 を考える。

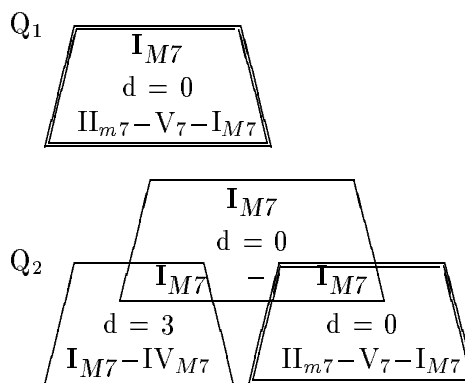


図5: 2種類の問い合わせ

図中、ケーデンスを囲む2重線は「このケーデンスをリハーモナイズせよ」という指示を意味する。図5の Q_1 は図4の(1)と(4)にマッチし、 Q_2 は(4)のみにマッチする。つまり、周囲の文脈も含めた検索を行うと解を絞りこむことが可能となる。

実際に Context Editor 上で組み立てられる問合わせは、可読性を考慮して、ケーデンス木とその部分木の指定という形ではなく、リハーモナイズしたい部分木と 0 以上の整数 N のペアという形をしている (図 3 中の context extent という欄). これは、リハーモナイズしたい部分木を中心に考えて、その部分木より N 段上に上ったケーデンスを根とするようなケーデンス木を和声文脈とする、という意味である.

相対的な類似度 厳密にマッチするケーデンス木が存在しない場合は、最も類似したケーデンス木を検索しなければならない. そのため、ここで相対的な類似度という概念を提案する.

相対的な類似度の定義は以下のようである. 今 3 つのオブジェクト項 A, B, C があるとす. もし条件「 $\text{lub}(A,B) \sqsubseteq \text{lub}(A,C)$ かつ $\text{glb}(A,C) \sqsubseteq \text{glb}(A,B)$ 」が成立すれば、 A と B の方が A と C より相対的に類似していると言い、 $\text{rs}(A,B) \sqsubseteq \text{rs}(A,C)$ と書く (図 6). ここで lub は least upper bound (最小上界), glb は greatest lower bound (最大下界) の意味である (lub と glb も包摂関係を用いて定義されているが紙面の都合で省略する). もしこの条件が成立しない場合は、

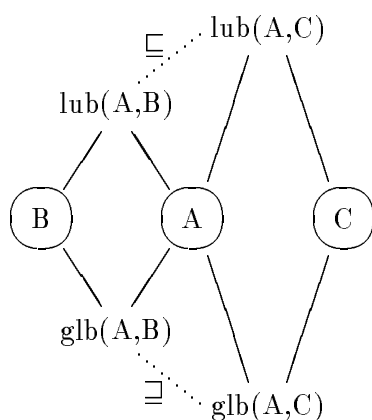


図 6: 相対的な類似度

A と B の類似度と A と C の類似度は比べられないと言い $\text{rs}(A,B) \not\sqsubseteq \text{rs}(A,C)$ と書く. 例えば、 $G_{7(9)}$ と $G_{7(b9)}$ の類似度と $G_{7(9)}$ と $G_{7(9,b13)}$ の類似度を比較する. ここで $b9 \sqsubseteq 9$, $\{9, b13\} \sqsubseteq \{9\}$ を仮定すると、 $\text{rs}(G_{7(9)}, G_{7(b9)}) \not\sqsubseteq \text{rs}(G_{7(9)}, G_{7(9,b13)})$ である. 実際には、ケーデンス木や和音のペアが二つ提示された時、どちらがより類似しているか比較できない (判定できない) 場合は

存在する. 相対的な類似度は、この比較できない場合を形式化している.

この相対的な類似度を用いて、問合わせのケーデンス木 Q と事例ベース中のケーデンス木 C_i ($i = 1, 2, \dots$) を比較し、 $\text{rs}(Q, C_i)$ が \sqsubseteq に関して最小 (極小) になるような C_i を検索する (比較できない場合が存在するので、検索結果は一般に複数).

ボイスリーディング 各声部 (ボイス) どののつながりを表現するオブジェクト項を導入し、そのオブジェクト項間の包摂関係に関して最小 (極小) のもの、つまり、最も滑らかに接続するボイスリーディングを選ぶ.

5 おわりに

我々は、パーピーブンの設計試作にあたり、以下の要件に留意した.

- 音楽的な直観に合った内部モデル
- その内部モデルを忠実にユーザに示すインタフェース
- その内部モデルをインタラクティブに思い通りに操作できるインタフェース
- 様々なユーザ層に対応するために、どのようなプラットフォームでも利用できる

今後は上記の点について評価、改良を加えるとともに、対象を和音進行から、一般の多声曲 (ソロピアノ) に広げたいと考えている.

謝辞: 本稿の一部を執筆するにあたり、(財) 日本情報処理開発協会 先端情報技術研究所 (内田俊一所長) の環境を利用させて頂きました.

参考文献

- [1] 後藤, 平田, ハービー君: 演繹オブジェクト指向に基づいてジャズらしいコードにリハーモナイズするシステム, 情報処理学会研究報告 96-MUS-16 (1996).
- [2] Hirata, K.: Representation of Jazz Piano Knowledge Using a Deductive Object-Oriented Approach, *Proc. of ICMC'96*, pp. 244-247 (1996).
- [3] KLIC 協会ホームページ, <http://www.klic.org/index.ja.html>.
- [4] Lerdahl, F. and Jackendoff, R.: *A Generative Theory of Tonal Music*, The MIT Press (1983).