

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

早稲田大学 理工学部

NTT 基礎研究所

後藤 真孝

平田 圭二

1996/07/28 情報処理学会 音楽情報科学研究会

No.1

1. はじめに

□ 音楽知識の表現・操作

- 手法・枠組

演繹オブジェクト指向：DOO

- 実証システム

ジャズピアノ知識ベースシステム：KBS

□ 研究目的

- DOOの枠組と音楽知識処理との

親和性が高いことを実証



- DOOに基づく音楽応用システム構築法を確立



No.3

内容

1. はじめに
2. 演繹オブジェクト指向に基づく
ジャズピアノ知識ベースシステム
3. ハービー君
4. 分散環境での実装
5. 実験結果
6. おわりに

No.2

□ ジャズピアノ知識ベースシステム (KBS)

- ジャズピアノ知識ベースとは？

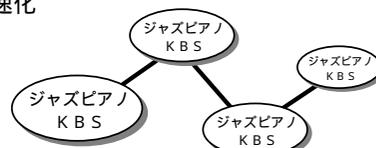
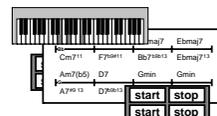
- ジャズピアノに関する様々な音楽知識のデータベース
- 外部からの問合せに柔軟に回答

- 音楽応用システムの構築

- メロディーに対するコード付け
- ジャズらしいコード進行への変換 (リハーモナイズ)
- コードネームから実際のボーシングへの変換
- ソロピアノ譜への編曲

- ジャズピアノKBSの将来構想

- GUIによる対話的な問合せ
- GUIによる知識ベースの容易な更新管理
- ネットワークを介した複数の知識ベースの協調問題解決
- 並列処理による高速化
- 実時間処理の達成



No.4

2. 演繹オブジェクト指向に基づく ジャズピアノ知識ベースシステム

□ D O O の枠組

- 様々な音楽的な概念や実体
 - ・ 合理的かつ自然に表現
 - ・ 見通し良く操作
- ジャズピアノ K B S を核にした音楽応用システム

- ・ 知識ベース部を共有
- ・ 問合せなどのインターフェースのみを新たに付加



- ・ 各々の音楽応用システムが持つ知識を有機的に統合

□ D O O の特長

- ・ オブジェクトによる柔軟な知識表現
- ・ 包摂関係などを用いたインスタンスに基づく検索
- ・ 知識や問い合わせの同一の枠組での記述
- ・ モジュラリティを持った知識の記述

No.5

□ オブジェクト間の包摂関係

• 包摂関係

- ・ 具体的 抽象的 複雑 単純 特殊 一般

note(pitch = C, octave = 5)
note(pitch = C)



chord_name(root = C, name = m7, tension = {11})
chord_name(root = C, name = m)
Cm7¹¹ Cm

• インスタンスに基づく検索

- ・ 知識をルールで表現する手法

不用意な抽象化や重要な情報の捨象により
意図したような推論結果が得られないことがある



- ・ インスタンスをそのまま知識ベースに格納
- ・ システムに与えられた情報を最大限に活用
- ・ 包摂関係： 検索のための強力なオペレータ

あるオブジェクトより具体的なオブジェクトは？

No.7

□ オブジェクトによる柔軟な知識表現

- 音楽的な概念や実体：オブジェクト

• 音符の表現

note(pitch = C)
note(pitch = C, octave = 5)

抽象的な C



具体的な C



- 固有属性：オブジェクトを識別する際に用いる

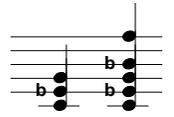
属性名： pitch octave

属性値： C 5

記述したい内容に応じて属性名 - 属性値のペアを
自由に増減できるため制約が少なく表現力が高い

• コードネームの表現

chord_name(root = C, name = m)
/ (notes = {C, Eb, G})



Cm Cm7¹¹

chord_name(root = C, name = m7, tension = {11})
/ (notes = {C, Eb, G, Bb, F})

- 非固有属性：オブジェクト間の関連だけを表す

ポインティングはコード自体を識別するには
影響を与えない付加的な情報

No.6

□ ジャズピアノ K B S の入出力

• 知識ベース構築時 (入力)

- ・ Herbie Hancock が弾いたソロピアノを採譜した譜面
- ・ その演奏に対するコメント
- ・ 原曲のコード進行

• 問合せ時 (入出力)

- ・ 包摂関係などから成る問合せ (入力)
- ・ 答え： 演繹した結果 (出力)

No.8

□ インタフェースの問題点

- インタフェースが並列論理型言語 K L 1 のみ
 - システムの内部構造の理解が必要
 - 新たな問合せをするたびに K L 1 でプログラミング



問合せの GUI 化が必要

No.9

□ システム入出力

小節番号, 拍位置 : コードネーム

• ユーザによる入力

- 原曲のコードネームと調
- メロディーの SMF
演奏確認用
(リハーモナイズでは
全く考慮されない)

```

1, 0: key(Bb)
1, 0: C m7
2, 0: F 7
3, 0: Bb maj7
4, 0: Eb maj7
5, 0: key(G)
5, 0: A m7(b5)
6, 0: D 7
7, 0: G min
8, 0: G min
    
```

• リハーモナイズ前後のコードネームを画面表示

- 結果を対比しやすいよう上下に並べる
- 処理途中でも結果の得られた部分から随時表示

Cm7	F7	Bbmaj7	Ebmaj7
Cm7 ¹¹	F7 ^{b9#11}	Bb7 ^{b9b13}	Ebmaj7 ^{9 13}
Am7(b5)	D7	Gmin	Gmin
A7 ^{#9 13}	D7 ^{b9b13}	Gm7 ⁹	G7 ^{#11 13}

• リハーモナイズ前後のコードの響きを

メロディーも合わせて演奏

- コードネームを理解できないユーザでも響きで確認可能
- 処理途中でも結果の得られた部分から随時演奏可能

No.11

3. ハービー君

□ ジャズピアノ K B S の応用システム

- 入力された原曲のコード進行を
ジャズらしいコード進行にリハーモナイズ
- リハーモナイズとは?
 - あるコード進行と似た機能を持つ
別のコード進行を求める操作
(一般にテンションを含んだ
より豊かな響きを持つコードに変える)

Cm7	F7	Bbmaj7	Ebmaj7
-----	----	--------	--------

↓

Cm7 ¹¹	F7 ^{b9#11}	Bb7 ^{b9b13}	Ebmaj7 ^{9 13}
-------------------	---------------------	----------------------	------------------------

• 問合せ用インタフェースの GUI 化

- 将来構想を実現する途中段階においても
実用性のある効果的な応用システム
- ハービー君をベースに将来構想を実現しやすく考慮
拡張性が高く実装が容易なようにシステム設計

No.10

□ リハーモナイズの実現

• 演繹規則による推論

Cm7	F7	Bbmaj7	Ebmaj7
-----	----	--------	--------

↓

Cm7 ¹¹	F7 ^{b9#11}	Bb7 ^{b9b13}	Ebmaj7 ^{9 13}
-------------------	---------------------	----------------------	------------------------

• 包摂関係

リハーモナイズされたコード 原曲のコード

• リハーモナイズは具体化に相当

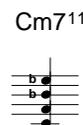
原曲のコードに対し
より具体化したコードを K B S に問合せる操作

• K B S からコードネームとボイスングを取得

- 原曲のコードネームの具体化の問合せ Cm7
→ 知識ベース中にある具体化したコードネーム
(同時に与えられた調も考慮)

• コードネームの非固有属性を検索

→ 対応するボイスング



• 複数の答えの候補からの選択

- 答えの得られる最長一致したコード進行を演繹結果
- 候補が複数あるときには乱数により選択

No.12

4 . 分散環境での実装

□ ネットワーク上に分散実装

- ・各処理やインタフェースの構成要素のモジュラリティを高める
- ・構成要素（プロセス）がお互いに通信して処理を進める
- ・高い拡張性
 - ・必要に応じて複数の計算機を付加してシステム拡張
計算量の多い問題を解く大きな計算能力を得られる
 - ・大規模分散データベースへ拡張する際の整合性が良い
Internet上の様々なKBSを接続
 - ・新たな機能の追加・拡張が容易
知識ベースの更新管理機能を追加する場合
無関係な既存プロセスの変更が不要
- ・容易な実装
 - ・個々のプロセスは小さな機能だけを実装すれば良い
実装や機能単位での再利用が容易
 - ・円滑な共同開発
通信部分の規約を定めプロセス間の独立性を高める
実現すべき機能に適したプログラミング言語を使用

□ R M C P

・ Remote Music Control Protocol

- ・MIDIとLANを融合した分散協調システムにおけるサーバ・クライアント間の通信プロトコル
- ・シンボル化された音楽情報をネットワークを通じて伝送するために設計

・ R M C P パッケージ

- ・コード情報
ジャズピアノKBSのコード入力用
問合せの種類 / 答の種類, 小節番号, 拍位置,
コードネーム, ボイシング
「具体化の問合せ」 「具体化の答え」
- ・演奏情報
MIDIメッセージ 
- ・演奏制御情報
演奏用プロセスに開始 / 停止を指示

・ブロードキャストによる情報共有

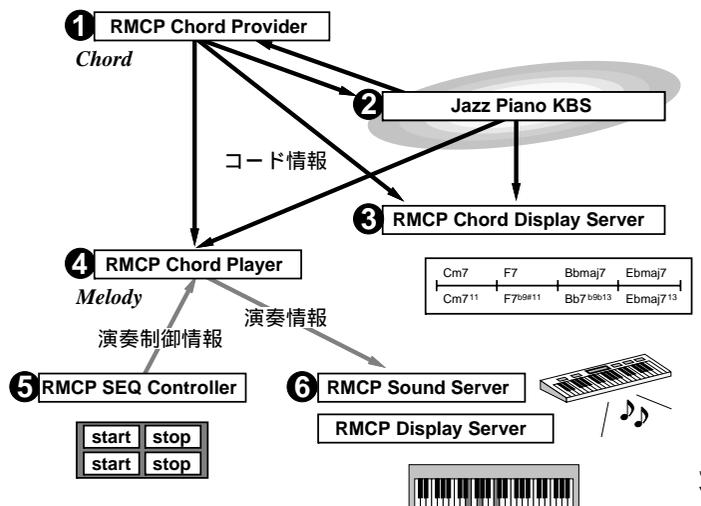
- ・クライアントがRMCPパッケージをブロードキャスト
- ・コード情報などを様々なサーバが共有して同時に活用

□ 実装環境

- ・ネットワーク: Ethernet
- ・ワークステーション: SGI Indigo2, Indy
- ・処理の構成要素: UNIXのプロセス
サーバ・クライアント・モデル
- ・プロセス間の通信: RMCP
- ・演奏の入出力: MIDI

□ ハービー君システム構成

- ① 問合せの発行（原曲のコードネームを送信）
- ② リハーモナイズ（ジャズピアノKBS）
- ③ 問合せ / 答えのコードネームを画面表示
- ④ 問合せ / 答えのコードをメロディーを含めて試聴
- ⑤ 演奏の開始 / 停止の指示
- ⑥ 演奏情報の聴覚化 / 視覚化



5 . 実験結果

□ 実験条件

- ・ K L I C 上に実装されたジャズピアノ K B S を運用
- ・ 入力知識 Autumn Leaves (1 コーラス 3 2 小節)
Herbie Hancock が弾いたソロピアノを採譜した譜面
その演奏に対するコメント
原曲のコード進行
- ・ リハーモナイズ対象曲
Autumn Leaves (Joseph Kosma)
Summertime (George Gershwin)

□ 実験結果

- ・ 合計約 3 8 0 個のオブジェクトを知識ベースに格納
- ・ Autumn Leaves
一旦知識ベースに知識として格納されたものから演繹
元の演奏とは異なるリハーモナイズもなされている
- ・ Summertime
異なる曲に対してもリハーモナイズできることを確認

No.17

□ 考察

- ・ 二つの研究目的
 - ・ 単にシステムの出力するコード進行の
音楽的善し悪しだけでは評価できない
- ・ D O O の枠組と音楽知識処理との親和性
 - ・ リハーモナイズ前後のコード進行の関係を
D O O の包摂関係を用いて簡潔に表現
 - ・ 音楽的に意味のある計算をしていることを確認
 - ・ ユーザの意図を合理的かつ自然に記述するのに
D O O の枠組が十分な表現力を提供できた



- ・ D O O に基づく音楽応用システム構築法
 - ・ 各処理や G U I の構成要素を異なるプロセスとして実装
 - ・ 負荷分散効果 / 高い拡張性と再利用性
 - ・ 今後のシステム拡張にも十分対応可能
- 問合せと答えの通信プロトコルは汎用に設計



No.18

6 . おわりに

□ まとめ

- ・ 原曲のコード進行をジャズらしいコード進行に
リハーモナイズするシステム「ハービー君」
- ・ D O O に基づくジャズピアノ K B S に
G U I を付加する際の一応用例
具体化と非固有属性の問合せによってリハーモナイズ
高い拡張性を考慮しながら分散環境で実装
- ・ D O O が音楽知識処理に適していることを
応用の観点からも一つ支持できた

□ 今後の課題

- ・ ハービー君の改良
前後のボイスのの流れの考慮
知識の効率的な入力法の開発
- ・ ジャズピアノ K B S の将来構想へ向けた拡張
ジャズピアノ K B S の問合せプロトコルの拡張
知識ベース更新管理の G U I 化

No.19