

演奏システム INSPIRATION の改良 —— SMF データのインポート機能と音高置換処理の検討

谷井章夫^{†,††} 片寄晴弘^{††,†††}

我々は、ユーザの入力演奏をスケールノートやコードノートへ置換することにより、スケールや和声が逸脱した演奏を音楽的に許容できる演奏に実時間で変換する演奏支援システム INSPIRATION の開発を進めてきた。本稿では、「アボイドノートやクロマティックノートを意図的に用いるための機構」と「音高置換処理に利用する PuriFi (Purificatory File) を SMF (Standard Midi File) 楽曲データから自動作成する機構」を中心に述べる。我々は、アボイドノートやクロマティックノートを意図的に用いるためのインタフェースとして、評価実験においてほとんどの被験者が支持した「黒鍵への打鍵を利用したインタフェース」と「ペダル操作を利用したインタフェース」を採用した。PuriFi を自動作成するには、調性とコード構成音を認識する必要がある。Spiral Array の考え方を基本とした提案手法により、92%の調性認識率と85%のコード構成音認識率が得られた。PuriFi の自動作成機構により、インターネット上で普及している多くの SMF 楽曲データを利用できるようになった。

Improvement of a Performance System “INSPIRATION” —— Consideration of SMF Import Framework and Note Change Function

AKIO YATSUI^{†,††} and HARUHIRO KATAYOSE^{††,†††}

We have developed a piano performance system called INSPIRATION which receives rough keyboard input and translating it into musically correct performance in real time. This system transforms notes input by a user into scale notes or chord notes. In this paper, we focus on mechanisms for using avoid-notes or chromatic-notes intentionally and a framework for data conversion of SMF (Standard Midi File) into PuriFi (Purificatory File: for user's performance translation). We adopted two interfaces: “Touching the black claviers in case of avoid-notes or chromatic-notes,” and “Using pedal in case of avoid-notes or chromatic-notes,” which most subjects preferred in our experiment. Recognition of chords and key boundaries are crucial for automatic generation of PuriFi. The proposed algorithm expanding Spiral Array exhibited recognition accuracy of 92% for key boundaries, and 85% for chord progression, respectively. Automatic generation of PuriFi enables us to use lots of SMF music data widely available on the internet.

1. はじめに

音楽演奏の楽しさとしては、1) 楽器演奏による自己表現、2) 演奏技能習得の達成、3) 合奏におけるインタラクションなどがあげられる。このうち、楽器演奏によって自己表現を行うには、音楽知識と演奏技能を前もって習得しておく必要がある。昨今の集团的音

楽教育では知識や技術の教育に重心が置かれており、自己表現の可能性が必ずしも追及されていなかった。音楽知識と演奏技能を習得していない生徒にとっては、音楽的な表現を試行することへの後ろめたさにもつながってきた¹⁾。近年、これらの事項に注目し、計算機を用いて音楽知識や演奏技能を補う演奏支援システムの開発が行われている。演奏支援システムは、楽器演奏に対する敷居を低くする役割がある。

我々は、鍵盤楽器における即興演奏を対象とした演奏支援システム INSPIRATION の開発を進めてきた。即興演奏を行うには、様々な音楽理論の理解と高度な演奏技能が要求されるため、ピアノ演奏非経験者

† 関西学院大学理学研究科
Graduate School of Science, Kwansei Gakuin University

†† 科学技術振興機構さきかけ研究 21
PRESTO, JST

††† 関西学院大学理工学部
School of Science and Technology, Kwansei Gakuin University

創作と演奏を同時に行う演奏行為。

はもとより、既存楽曲の再現しが経験のないピアノ演奏者にとっても非常に困難な演奏行為である。INSPIRATION は、スケールや和声が逸脱した演奏を音楽的に許容できる演奏に実時間で変換するため、楽器演奏が苦手なユーザでも気軽に演奏表現を楽しむことができる。具体的には、ユーザの表現する演奏意図を、打鍵タイミング、音の強弱、音の高低、同時に入力する音数などの物理的な量としてとらえ、それらをできる限り保持した状態で、スケールノート やコードノートへ置換することにより音楽知識と演奏技能を補う。INSPIRATION は、ピアノ即興演奏非経験者 を対象として開発を進めている。楽器演奏経験のない素人でも扱えるシステムを初期システムとして開発し、先の速報²⁾において、演奏変換処理の概要について報告した。その後、システムの実用化に向け、これまでのINSPIRATION の問題点を検討したうえで、同時に利用者へのアンケートを実施した結果、楽曲データ数の少なさと過剰な演奏補正が課題として浮かび上がった。本稿では、これらの課題への対応に焦点を当てる。以下、2章では、計算機による音楽知識・技能の補完、3章では、インタフェース向上にむけての課題、4章では、意図的に音を外すためのインタフェース、5章では、一般に流通している SMF 楽曲を利用するための機構について述べ、6章でまとめる。

2. これまでの INSPIRATION の研究開発を通じて

2.1 計算機による音楽知識・技能の補完

音楽の制作や演奏によって自己の意図を表現するためには、前もって音楽知識・技能を身につけておく必要がある。何らかの理由でこれらの習得が不十分な場合、計算機によってその部分を補完することにより、表現の範囲を拡張することが可能となる。演奏支援システムの中で音楽知識・技能を補完するシステムとしては以下のようなものがある。「Coloring-in Piano³⁾」は、楽曲の音高情報を計算機に事前に入力しておくことで、打鍵位置にかかわらず正しい音高を出力するシステムである。「sfp⁴⁾」は、指揮的な演奏表現によりテンポや音量などを操作するシステムである。「音機能固定楽器⁵⁾」は、従来の鍵盤楽器の音高配置を音の機能を固定した配置に並べ替えた楽器である。ユーザは、演奏したい音の機能を鍵盤上から瞬時に選択できる。「カシオトーン」は、スケールノートへの自動変換

スケールを構成する音高。

楽器演奏経験のないまったくの素人、ギターなどの楽器演奏経験者、既存楽曲の再現しが経験のないピアノ演奏者も含む。

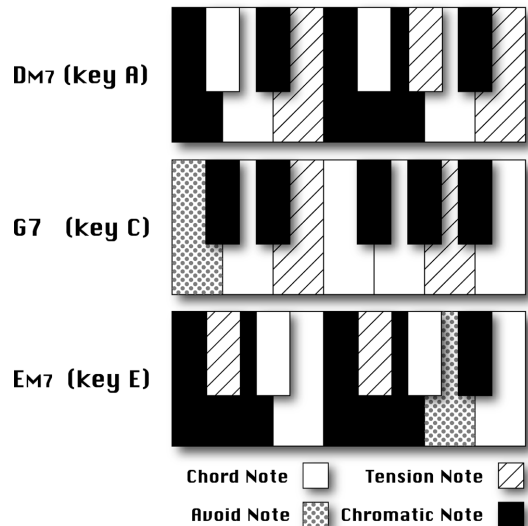


図1 鍵盤上における音の機能の配置

Fig. 1 Assignment of note functions to keyboards.

により誰でもアドリブ演奏を楽しめる機能を持つ。また、簡単な打鍵操作により、意図した minor chord や Major 7th chord を演奏できる機能もある。

演奏支援システムをデザインするには次の事項を考慮する必要がある。

- 演奏意図をどのようにとらえるか。
- どのような音楽知識・技能を補完するか。
- どのようなインタフェースにより実現するか。

たとえば、鍵盤楽器を対象とした演奏支援システムの場合、システムで採用する演奏方法が、鍵盤楽器本来の演奏方法と著しく異なると、周囲の観衆に対して不自然に映るほか、演奏にも違和感を与えるため、ユーザのモチベーションの低下につながる可能性がある。本研究では、これらの事項に十分考慮し、ピアノ即興演奏を対象とした演奏支援システム INSPIRATION の開発に取り組んできた。

2.2 INSPIRATION システムの概要

鍵盤上の各音高は、音の持つ効果と使用条件から4種類の機能に分類され、その配置は調性やコードネームの変化とともに複雑に変化する(図1)。即興演奏を行うには、音の機能を理解し、鍵盤上の配置を事前に暗記しておく必要がある。さらに、意図した機能を持つ音高を鍵盤上から瞬時に選択できる技能も必要である。この音楽知識と演奏技能を習得していなけ

コードノート: ルート音, 3 度音, 5 度音, 7 度音。

テンションノート: 9 度音, 11 度音, 13 度音。

アボイドノート: コードサウンドの構成上, 除外される音高。

クロマティックノート: スケールノート以外の音高。



図 2 システム概要
Fig.2 System overview of INSPIRATION.

れば、音の強弱や音の高低などの演奏表現を行うことができない。このことが、即興演奏を難しくしている要因の 1 つである。

INSPIRATION は、ユーザの入力演奏に対して音高置換処理を施すことにより、音楽理論に基づいた演奏に実時間で変換する。ユーザは、音の機能の配置を意識する必要がなくなり、音の高低や音の強弱などの演奏表現に集中できる。

図 2 にシステムの概要を示す。本システムは、Purifi (Purificatory File) と SMF (Standard Midi File) というデータファイルを使用している。SMF とは、MIDI 形式の音楽データを扱う際の標準的なファイル形式であり、楽曲データのエディタやプレーヤの多くが対応している。本システムで BGM として使用する SMF には、複数パート (ギター、ベース、ストリングス、ドラムなど) の演奏が収録されている。本システム独自のフォーマットである Purifi には、音高置換処理に必要な情報 (コードネームの移行時刻 (tick)、各移

コード移行時刻	調性	ベース	コード構成音
1900	D	C	C E G# B
2100	D	F#	D F# A A#
2500	D	F	C# F G# B
3000	D	G#	C# F G# B
3800	G	G#	D# F# A A#
12000	C	E	D# G A# B
12500	G	E	F# A A#
13000	G	F	D# F# G# A#
13300	F	E	D F G A
14100	F	F	D# F A B
14400	F	G	E G A# B
15200	F	D	E F# A A#

図 3 Purifi 内部データの例
Fig.3 The example of Purifi format.

行時刻における調性、ベース音、コード構成音が記載されている (図 3)。

ユーザは、「打鍵タイミング、同時に入力する音数、大まかな音の高低、音高の推移方向、音の強弱」を意

現在のところ、ユーザの演奏入力から変換後の演奏出力までの処理時間が 44 ms に抑えられており、システムのリアルタイム性が保持できていると考えている。打鍵から音の出力までに 50 ms 以上の遅延が発生すると、演奏が困難になることがヤマハの西堀ら⁶⁾により確認されている。

調性が分かれば、スケールノートを特定できる。SMF 楽曲データ内において和音演奏を構成する音高群であり、コードノートとテンションノートが混在している。

○ ベース音 ◻ コード構成音 ◻ スケールノート

図 4 INSPIRATION を介した演奏の一例

Fig. 4 The performance transformed by INSPIRATION.

識したうえで、音の機能の配置を無視した大ざっぱな演奏を入力する。システムは、同時入力音数を感知し、和音演奏がフレーズ演奏か判断する。フレーズ演奏と判断された音高は、スケールノートへ置換する。和音演奏と判断された音高群は、コード構成音へ置換する。ただし、複数音入力の構成音に C3 以下の音高を含む場合、最も低い音高をベース音へ置換する。音高置換処理は、ユーザの入力演奏と変換後の演奏の音高推移方向を一致させたうえで、置換前後の音高差が最小になるよう音高を置換する。本システムでは、音楽的制約の違いから以下の 3 つの演奏モードを用意している。

- Pops-style mode: 既存のポップス楽曲をベースに、コードポジションやサブフレーズなどを演奏するモード。
- Blues-style mode: ブルースの典型的なコード進行に従って、ブルース風の演奏に変換するモード。
- Free-style mode: 自動生成するコード進行をベースに、調性からの離脱やルートの半音下降進行などを自由に展開するモード。

演奏入力インタフェースには、MIDI キーボードを想定している。ユーザの入力演奏に対して、本システムにより演奏変換した事例を図 4 に示す。音高のみが修正され、打鍵タイミング、音長、同時発音数が保持されていることが分かる。変換前後の演奏を <http://ist.ksc.kwansei.ac.jp/~katayose/members>

/yatsui/inspiration/ から試聴できる。INSPIRATION を利用した演奏状況についても閲覧できる。

2.3 インタフェース向上にむけての課題

一般に、コードノートやテンションノートによって構成されたフレーズは、音楽的には自然であるが、緊張感の乏しいものになる傾向がある。一方、アポイドノートやクロマティックノートを経過音として用いると、緊張感が生まれる。即興演奏において、コードノートとテンションノートのみにより構成されたフレーズを演奏することは稀であり、アポイドノートやクロマティックノートをを用いて意図的に音を外すことも少なくない。初期の INSPIRATION は、意図的に音を外すためのインタフェースを備えていなかった。そこで、アポイドノートとクロマティックノートに関する音楽知識と技能を補完する機能を実装し、意図的に音を外すためのインタフェースの開発に取り組んだ。

INSPIRATION をより多くの人に利用してもらうためには、その人の好みに合う楽曲データ数を充実させる必要がある。初期の INSPIRATION では、手作業により PuriFi を作成していたため、楽曲数の拡充に限界があった。SMF 楽曲データは、MIDI 形式の音楽データを扱う際の標準的なファイル形式であり、広く一般に普及している。これに注目し、SMF 楽曲データから PuriFi を自動作成する機構を実現することにより、楽曲データ数の不足を本質的に解決できると考えた。SMF 楽曲データから PuriFi を作成するには、調性やコードに関する音楽知識が必要であり、これを補完する機能を実装する。

INSPIRATION の初期システムでは、楽器演奏経験のない素人でも気軽に即興演奏を楽しめるシステムの開発を目指した。学会や展示会 において実施したユーザに対するアンケート調査では、この目的は十分に満たされていることが確認された。一方、要望として、即興演奏の対象曲数を増やしてほしいという意見が多数寄せられた。そのほかに、音楽経験者からは、意図的に音を外せるようにしたいとの意見が寄せられた。これらの課題は、我々もその必要性を認識していたものである。

以下、意図的に音を外すためのインタフェースと PuriFi の自動作成機構を述べる。

3. 意図的に音を外すためのインタフェース

フレーズ演奏において意図的に音を外すには、アポ

日本ソフトウェア科学会 WISS2001⁸⁾ において約 50 人、ATC 関西学生卒業作品展「創っ展—TUKUTTEN」Vol.6 において約 150 人、学内で約 200 人の被験者からコメントを得た。

イドノートとクロマティックノートに該当する鍵盤を瞬時に判断する必要があり、素人の演奏者には難しい。本研究では、簡易に音を外すインタフェースとして、3種類のインタフェースを開発した。共通の処理として、同時入力音数を感知し、和音演奏かフレーズ演奏かを判断したうえで、和音演奏と判断された音高群をコード構成音へ置換する。以下、各インタフェースのフレーズ演奏変換処理について述べる。

A) 鍵盤の視覚的特徴を利用したインタフェース

インタフェース A では、鍵盤の視覚的特徴として鍵盤の色の違い（白鍵と黒鍵）を利用した。具体的には、白鍵への打鍵に対しては、コードノートかテンションノートへ置換し、黒鍵への打鍵に対しては、アポイドノートかクロマティックノートへ置換する。ユーザは、黒鍵を打鍵することにより、意図的に音を外す。音高の置換は、ユーザの入力演奏と変換後の演奏の音高推移方向を一致させたうえで、置換前後の音高差が最小になるよう処理する。

B) ペダル操作を利用したインタフェース

インタフェース B では、ペダルのオン・オフ操作により、鍵盤上の音機能配置の切替えを行う。具体的には、ペダル・オフ時の打鍵に対しては、コードノートかテンションノートへ置換し、ペダル・オン時の打鍵に対しては、アポイドノートかクロマティックノートへ置換する。ユーザは、ペダルを踏みながら打鍵することにより、意図的に音を外す。音高の置換は、ユーザの入力演奏と変換後の演奏の音高推移方向を一致させたうえで、置換前後の音高差が最小になるよう処理する。

C) 音高の使用頻度を考慮したインタフェース

過度の演奏補正は、演奏表現の幅を狭めることにつながる可能性がある。この問題を指摘し、対処を行ったものとして、東京理科大学の柳川らによる「即興演奏における演奏補正システム⁷⁾」がある。柳川らのシステムは、旋律のデータベースに基づいて、様々な条件下でのアポイドノートやクロマティックノートの使用頻度を算出し、使用頻度の低いアポイドノートやクロマティックノートのみを補正することで、表現の幅を狭めず演奏補正を実現した。この手法をインタフェース C として実装した。処理で用いる特徴ベクトルを以下に示す。

- 対象音の直前の音がアポイドノートやクロマティ

たとえば、キーが A でコード名が D メジャー 7th の場合に「E F F# G」という順で単音入力された場合、変換後の演奏は「E F# G G#」となる（図 1）。ユーザは、黒鍵である F# の打鍵により「音を外したい」という意図を示す。

表 1 意図的に音を外すためのインタフェースの評価結果
Table 1 Estimation for three interfaces.

	ピアノ演奏 経験者	楽器演奏 経験者	楽器演奏 非経験者
インタフェース A	6 人	5 人	2 人
インタフェース B	2 人	2 人	3 人
インタフェース C	0 人	0 人	0 人
必要性アリ	8 人	6 人	1 人
必要性ナシ	0 人	1 人	4 人

クノートか。

- 対象音の発音のタイミングは、8 分音符レベルで表か裏か。
- 対象音の発音のタイミングは、コードチェンジと同時か。
- 対象音とその直前の音との音高差。
- 対象音とその直前の音との間に休符があるか。

音高の置換は、ユーザの入力演奏と変換後の演奏の音高推移方向を一致させたうえで、置換前後の音高差が最小になるよう処理する。

3.1 3種類のインタフェースの評価

前述した意図的に音を外すための 3 種類のインタフェースについて、即興演奏経験のない被験者 20 人に対して「音を外す」演奏表現が最もやすかったインタフェースの選択を求めた。また、音を外す演奏表現の必要性についても意見を求めた。被験者の詳細は以下のとおりである。実験結果を表 1 に示す。

- ピアノ演奏経験者 8 人
- 楽器演奏経験者（ピアノ非経験者） 7 人
- 楽器演奏非経験者 5 人

インタフェース A に対する評価が、全体的に最も高かった。音を外す演奏表現と素人が持つ黒鍵に対するイメージが類似している点をうまく利用できていて、直感的に操作しやすいという意見が得られた。インタフェース B に対しては、ペダル本来の機能の印象が強いため、スイッチとしての使用に違和感を感じるという意見があった。しかし、どちらのインタフェースも、素人の演奏者でも意図的に音を外せる点が評価された。使いやすさに関しては、結局は慣れ次第ではないかという意見もあった。インタフェース C は、鍵盤位置を特別意識しなくても音を外す演奏表現が可能である。しかし、演奏者が意図的に音を外すにはアポイドノートやクロマティックノートの配置を暗記しておく必要がある。INSPIRATION は、即興演奏非経験者を対象としているため、インタフェース C は適当でない。インタフェース C が参考にした柳川らの手法は、ある程度の音楽知識と演奏技能を習得したピアノ演奏経験者を対象としていると考えられる。インタ

フェース A および B と比較して、インタフェース C に対する評価が低かったことから、被験者が“外れた音の自然さ”よりも“音を外したいときに外せる”ことを重視していることが分かる。また、我々は、ある条件下で使用された外れた音を、自然と感じるか不自然と感じるかを判断することは、本質的に難しいと考えている。好みの音楽などが要因となって、心地良いと感じる聴取者もいれば、違和感を覚える聴取者もいる。このように、聴取者の主観が影響するため、外れた音の自然さは万人に共通するものではない。以上の評価から、INSPIRATION では、意図的に音を外すためのインタフェースとして、インタフェース A および B を採用した。

音を外す演奏表現については、ピアノ演奏経験者と楽器演奏経験者のほとんどが、その必要性を感じていることが確認された。一方、楽器演奏非経験者は、あまり必要性を感じておらず、コードノートやテンションノートへの置換処理で満足していることが分かった。

4. PuriFi の自動作成機構

INSPIRATION は、PuriFi (Purificatory File) を参照して音高置換処理を行っている。PuriFi を自動作成するためには、SMF 楽曲データから調性とベース音とコード構成音を抽出する必要がある。図 5 に、PuriFi の自動作成機構を示す。以下、処理の詳細について述べる。

4.1 調性・転調認識

調性・転調認識の手法としては、調性ごとに各音高の出現率を設定し、楽曲データ内の各音高の出現率との相関から調性を判断する単純な手法が考えられるが、精緻に行うには事前に大量の楽曲データから調性ごとの各音高の出現率を求めておく必要がある。そこで本研究では、座標演算により調性を特定する Elaine Chew の Spiral Array⁹⁾ を元にした手法を用いる。Spiral Array は、各音高を 5 度圏の順で 3 次元座標空間内に配置したものである(図 6)。調性的に重要な音高が密集しており、各調性ごとに 1 つの座標値に集約できる。「調性ごとに集約された座標値」と「対象楽曲の音高群を集約した座標値: center of effect (以下 c.e.)」の距離が最小となる調性を対象楽曲の調性として判断する。Spiral Array は、次元の異なる音楽的な実体(音高と調性)を、同じ次元の情報(座標値)として同じ座標空間内に表現することを可能にした。

Spiral Array は、楽譜データのように、基本となる調性が既知なデータを対象としているため、C と B# や F と E# などをあえて区別している。しかし、



図 5 PuriFi の自動作成機構
Fig. 5 Extraction of PuriFi from SMF.

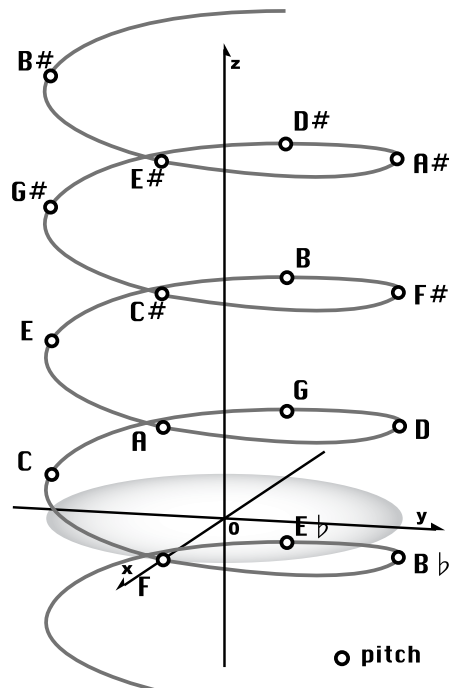


図 6 Spiral Array
Fig. 6 Spiral Array.

SMF 楽曲データのように調性の分からない状況下では、これらを区別することができない。その結果、座標空間内に同一音高を意味する座標値が複数存在してしまう。そこで、本研究ではこの問題を解決するために、Spiral Array を改良したデータ表現として Spiral in Torus を提案する。以下、データ表現 Spiral in Torus と調性・転調認識手法について述べる（図 7）。

Spiral in Torus のデータ表現

step1 ~ step3 は、Spiral Array のアイデアを採用している。step4 は、上記の問題点を解決するために拡張した処理である。

step1 各音高を示す座標値 $P(k)$ の算出

$$P(k) \stackrel{\text{def}}{=} \begin{bmatrix} x_k \\ y_k \\ z_k \end{bmatrix} = \begin{bmatrix} \sin \frac{k\pi}{2} \\ \cos \frac{k\pi}{2} \\ k \cdot \sqrt{2/15} \end{bmatrix} \quad (1)$$

step2 各コードを示す座標値 C の算出

ルート音 ≥ 5 度音 ≥ 3 度音の順で重み付けを行い、各音高を示す座標値 $P(k)$ から各コードを示す座標値 C へ集約する。重み係数については、文献 9) における設定値を採用した。

$$C_M(k) = 0.516 \cdot P(k) + 0.315 \cdot P(k+1) + 0.168 \cdot P(k+4) \quad (2)$$

$$C_m(k) = 0.516 \cdot P(k) + 0.315 \cdot P(k+1) + 0.168 \cdot P(k-3) \quad (3)$$

$$C = \{C_M(k) \text{ for all } k\} \cup \{C_m(k) \text{ for all } k\} \quad (4)$$

step3 各調性を示す座標値 T の算出

主和音 \geq 属和音 \geq 下屬和音の順で重み付けを行い、各コードを示す座標値 C から各調性を示す座標値 T へ集約する。重み係数は、文献 9) における設定値を採用した。

$$T_M(k) = 0.516 \cdot C_M(k) + 0.315 \cdot C_M(k+1) + 0.168 \cdot C_M(k-1) \quad (5)$$

$$T_m(k) = 0.516 \cdot C_m(k) + 0.315 \cdot C_m(k+1) + 0.168 \cdot C_m(k-1) \quad (6)$$

$$T = \{T_M(k) \text{ for all } k\} \cup \{T_m(k) \text{ for all } k\} \quad (7)$$

step4 z 成分に対する処理

$P(k)$, T の z 成分のみオクターブで畳み込み、 θ_z として独立させる。 z 成分を角度として表すことにより、 $P(k)$ や T を 1 つの座標値に集約できる。

$$\theta_z = \frac{z}{\sqrt{2/15}} \cdot 30^\circ \quad (8)$$

k は、Spiral Array 上の各ノートに便宜的に振られた番号を示す。たとえば、図 7 の場合、 $z = 0$ である F を $k = 0$ とすると、B は $k = -1$, C は $k = 1$, G は $k = 2$, D は $k = 3 \dots$ となる。

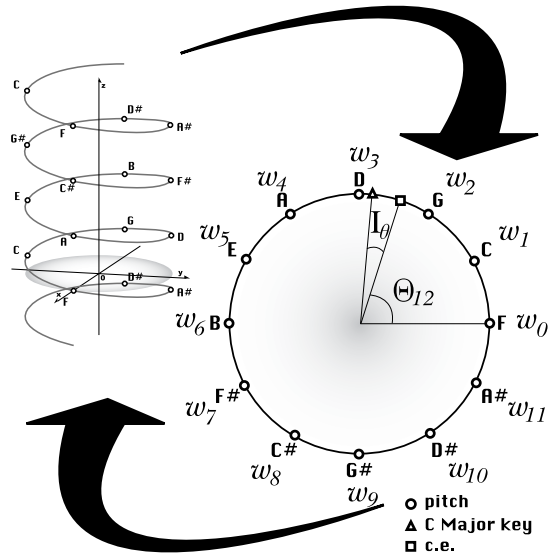


図 7 Spiral in Torus
Fig. 7 Spiral in Torus.

Spiral in Torus による調性・転調認識

step1 c.e. の x 成分と y 成分の算出

$$c = \sum_{i=1}^N \frac{d_i}{D} \cdot p_i \quad \text{where } D = \sum_{i=1}^N d_i \quad (9)$$

c は c.e. を示す座標値、 p_i は各音高を示す座標値、 d_i は各音高の音長、 N は認識対象区間内のノート数を示す。

step2 c.e. の z 成分の算出

$$\Theta_{i+1} = \begin{cases} \Theta_i + \frac{w_i}{W_i} \cdot (30 \cdot i - \Theta_i) & (w_i \neq 0 \text{ かつ } 30 \cdot i - \Theta_i \leq 180^\circ \text{ のとき}) \\ \Theta_i + \frac{w_i}{W_i} \cdot (30 \cdot i - \Theta_i - 360) & (w_i \neq 0 \text{ かつ } 30 \cdot i - \Theta_i > 180^\circ \text{ のとき}) \\ \Theta_i & (w_i = 0 \text{ のとき}) \end{cases}$$

$$\text{where } W_i = \sum_{k=0}^i w_k \quad (10)$$

w_i は、各音高の音長の総和値が全ノートの音長の総和値に占める割合を示す。 $i = 0$ から順に計算し、 $i = 11$ で終了する。c.e. を示す z 成分に対応する角として Θ_{12} を採用する。

「 $w_2 = 1/4, w_3 = 1/4, w_{10} = 1/2$ 」の場合を例にあげ、計算過程を以下に示す。

$$i = 0 \rightarrow w_0 = 0 \text{ より, } \Theta_1 = \Theta_0 = 0$$

$$i = 1 \rightarrow w_1 = 0 \text{ より, } \Theta_2 = \Theta_1 = 0$$

$$i = 2 \rightarrow w_2 = 1/4, W_2 = 1/4 \quad \Theta_3 = 0 + 1 \times (30 \times 2 - 0) =$$

60

$i = 3 \rightarrow w_3 = 1/4, W_3 = 1/2 \quad \Theta_4 = 60 + 1/2 \times (30 \times 3 - 60) = 75$

$i = 4 \sim 9 \rightarrow w_i = 0$ より, $\Theta_{i+1} = \Theta_i = 75$

$i = 10 \rightarrow w_{10} = 1/2, W_{10} = 1 \quad \Theta_{11} = 75 + 1/2 \times (30 \times 10 - 75 - 360) = 7.5$

$i = 11 \rightarrow w_{11} = 0$ より, $\Theta_{12} = \Theta_{11} = 7.5$

よって, この場合, c.e. を示す z 成分に対応する角は, 7.5° となる. なお, たとえば, w_3 は, D が認識対象区間に占める割合を示し, W_3 は, F, C, G, D が認識対象区間に占める割合を示す.

step3 T と c.e. との距離 I の算出

$$I = \sqrt{(x_T - x_c)^2 + (y_T - y_c)^2 + (Z_I)^2} \quad (11)$$

$$\text{where } Z_I = \frac{I_\theta}{30} \cdot \sqrt{2/15}$$

$$\text{and } I_\theta = |\Theta - \Theta_{12}| \quad (0 \leq I_\theta \leq 180)$$

なお, T の x 成分を x_T , y 成分を y_T , c.e. の x 成分を x_c , y 成分を y_c , T の z 成分に対応する角を T_θ とする.

step4 c.e. の z 成分の算出

I が最小となる調性を調性候補として採用する.

認識対象区間を 4 小節に設定し, それを 16 分音符分シフトし, 以上の処理を繰り返す. 得られた調性が変化した時刻を転調時刻と判断する.

4.2 ベース音・コード構成音認識

調性音楽理論に基づいた楽曲のコード進行認識手法として, Winograd の手法¹⁰⁾ をはじめとして, 数多くの研究事例がある. 現代のポップス楽曲は, 一時的転調を多く含んだ楽曲や, コード進行上の連続するコードの音楽的なつながりを無視した楽曲も少なくない. このような楽曲に対しては, 調性音楽理論に基づいた従来の手法では解析が困難である. 本研究の「ベース・コード構成音認識」の目的は, INSPIRATION の和音演奏変換に用いる音高群を決定することである. ユーザの入力演奏に対して変換処理を施した和音演奏と BGM の伴奏パートの行う和音演奏との整合性を高める必要がある. そこで本研究では, 伴奏パートの和音演奏の構成音をできる限り正確に抽出することに重点を置いた以下のようなオリジナルの手法を採用した.

4.2.1 コード移行時刻の抽出

SMF 楽曲データ 10 曲に対して事前に調査したところ, コード移行時刻上で発音される音高群に以下の特徴がみられた.

- ベース音の発音がある.
- ドラムスの発音がある.

- 同時発音数が多い.
- ベロシティ (音の強さ) の総和が大きい.
- デュレーション (音の長さ) の総和が大きい.

我々は, これらの特徴をヒューリスティクスとして利用し, コード移行時刻の抽出を行うことにした. 各項目の特徴量は, 小節単位で極端に変化する可能性があるため, 小節単位で以下の処理を行う. なお, 特徴量の抽出は, 16 分音符間隔で行っている.

step1 ベース系楽器の発音の有無を調べる. ベース系楽器の発音がない場合は, 持続音系楽器と減衰音系楽器の演奏内容でベース音域の発音の有無を調べる. 存在した時刻をコード移行時刻の候補として絞る.

step2 ドラムスの発音の有無を調べ, さらに候補を絞る. ドラムスの発音がない場合は, step1 で得られた候補を保持する.

step3 各候補 i ごとに, 同時発音数 α_i , ベロシティ総合値 β_i , デュレーション総合値 γ_i を求めたうえで, H を求める.

$$H = \left(\frac{A}{\|A\|} + \frac{B}{\|B\|} + \frac{C}{\|C\|} \right) / 3 \quad (12)$$

ここで, A, B, C は, i 番目の要素が, それぞれ $\alpha_i, \beta_i, \gamma_i$ であるベクトルである. H の要素 h_i のうち閾値を上回った要素に対応する地点を, コード移行時刻として認定する.

4.2.2 ベース音・コード構成音の抽出

ベース系楽器は, 一般的にコード移行時刻においてルート音 (ベース音) を演奏することが多い. そこで, コード移行時刻上の最低音をルート音 (ベース音) として採用することにした.

コード構成音の抽出の目的は, コード移行時刻の各区間内において, コード構成音としてふさわしい音高を抽出することにある. コード構成音としての確信度を音高ごとに算出し, 確信度の高い音高をコード構成音として認定する. 減衰音系楽器は, コード構成音の使用回数が他の音高よりも多いため, 音高の使用回数を各音高のコード構成音としての確信度とする. ベース系楽器と持続音系楽器については, コード構成音の持続時間の総和が他の音高よりも長い場合, 各音高のデュレーション総合値が 16 分音符の何倍に相当するかを算出し, その値をコード構成音としての確信度とする. なお, 和音を構成する音高については, コード構成音としての可能性が高いため, 大きな重みを与えている.

楽曲 10 曲を用いて行った前実験において, 0.2 を閾値として設定したところ, 最もコード変化地点認識率を上げることができたため, これをデフォルト値として利用した.



調性・転調認識

正解例	C Major	F Major
認識結果	C Major	F Major

コード構成音認識

正解例	C E G	G B D	F A C	C E G
認識結果	C E G	G B D	F G C	C E G

図 8 認識結果例

Fig. 8 Example of recognition result.

4.2.3 コード構成音の補充

調性を特定することにより、使用されうるコードを絞ることができ、和音構成音数の少ない楽曲データに対してコード構成音を補うことができる。ここでは、コード構成音の抽出が2音以下の場合に限り、コード構成音の補充を行う。処理の流れを以下に示す。

step1 抽出されたルート音（ベース音）と同じルート音（ベース音）を持つコード候補に絞る。

step2 抽出されたコード構成音を含むコード候補を抽出する。

step3 決定したコード候補のコード構成音により補充する。

4.3 認識結果

インターネット上で配布されている SMF 形式のポップス楽曲データ 10 曲に対して調性・転調認識とコード構成音の認識率を調査した。この結果、92%の調性の認識率、85%のコード構成音の認識率が得られた。認識率の算出方法を以下に示す。

$$\text{調性認識率 (\%)} = (\alpha/N) \cdot 100$$

$$\text{コード構成音認識率 (\%)} = (\beta/N) \cdot 100$$

N: 全小節数 × 16

α : 正解例と調性が一致した地点の総数

β : 正解例とコード構成音が一致した地点の総数

なお、正解例としては、音楽経験者複数名が話し合いうえ、選定したものをを用い、1小節につき16カ所で認識結果との照合を行った。調性・転調認識に関して正解例と一致しなかったケースとしては、近親調間の転調時刻の不一致がほとんどであった。コード構成音認識においては、アルペジオ演奏されたコードノートとフレーズの構成音を同等に扱ってしまい、コード構成音が正解例と一致しないケースがあった。上記の理由で正解例と一致しなかった認識結果例を図8に示す。近親調間では、構成するスケールノートが類似し

ている。FメジャーとCメジャー間では、「C, D, E, F, G, A」のスケールノートが共通している。3小節目のフレーズは、CメジャーとFメジャーの共通のスケールノートにより構成されており、INSPIRATIONにおけるフレーズ生成においてどちらのスケールノートを用いても、特に大きな問題とはならない。なお、式(2)~(7)の係数の設定値を対象曲の特徴に応じて決定すれば、認識精度が上がる可能性があり、今後の課題としたい。また、3小節目のコード構成音が正解例と一致していない。正解例の3小節目のコード構成音は、Fメジャーコードのコードノートにより構成されている。認識結果のコード構成音Gは、Fメジャーコードにおける9thのテンションノートである。このように、誤認識についても、システムの不適切な動作の要因となるものは少なく、INSPIRATIONにおける和音生成に用いるうでは、大きな問題にはならないと考えている。

5. おわりに

即興演奏を行うためには、様々な音楽知識と高度な演奏技能が必要である。我々は、打鍵タイミングや音の強弱、音の高低、同時に入力する音数などのユーザの大まかな演奏意図を保持し、スケールノートやコードノートへの置換処理によって、音楽理論に基づいた演奏に実時間で変換する演奏システム INSPIRATION の開発を進めてきた。これにより、音楽知識と演奏技能を習得していない素人の演奏者でも、即興演奏による様々な演奏表現を行うことが可能となった。

本稿では、アポイドノートとクロマティックノートの扱いに関する3種類のインタフェースとその評価、一般に流通している SMF 楽曲を対象とするための機構を中心に述べた。通常の鍵盤楽器において、「音を外す」演奏表現を行うためには、アポイドノートやクロマティックノートに関する知識と鍵盤上の配置の理解が必要になり、初期の INSPIRATION では対応していなかった。意図的に音を外すためのインタフェースの追加により、音の機能の配置を理解していなくても、視覚的な判断により音を外す演奏表現が可能となり、演奏表現の幅を広げることができた。また、演奏変換するために必要な情報を書き込んだ PuriFi (Purificatory File) を SMF 楽曲データから自動作成する機能の追加により、楽曲データ不足に対処する目処がついた。

楽曲制作やライブ演奏において INSPIRATION を利用すれば、素人でも能動的に音楽へ参加することができる。また、即興演奏における演奏表現のトレーニ

ングシステムとしての利用形態も考えられる。今後は、ギターやバイオリンなどの弦楽器への展開を予定している。

謝辞 本研究の一部は、日本学術振興会未来開拓学術研究推進事業 (JSPS-RFTF 99P01404)、科学技術振興機構さきがけ研究 21「協調と制御」領域の支援を受け実施されました。NTT コミュニケーション科学基礎研究所の平田圭二氏には、熱心かつ有益なご討論ならびにご協力をいただきました。ここに記して感謝いたします。

参 考 文 献

- 1) 安西祐一郎, 草原真知子, 片寄晴弘, 笹田剛史, 中津良平, 黒川隆夫: 自己の表現, pp.67-113, 岩波書店 (2000).
- 2) 谷井章夫, 片寄晴弘: 音楽知識と技能を補うピアノ演奏システム INSPIRATION, 情報処理学会論文誌, Vol.43, No.2, pp.256-259 (2002).
- 3) 大島千佳, 宮川洋平, 西本一志: Coloring-in Piano: 表情付けに専念できるピアノの提案, 情報処理学会研究報告, Vol.2001, No.42, pp.69-74 (2001).
- 4) 奥平啓太, 片寄晴弘: 指一本によるピアノ演奏システム: sfp, 情報処理学会ヒューマンインターフェース研究会報告, 2003-HI-102, No.10 (2003).
- 5) 西本一志, 渡邊 洋, 馬田一郎, 間瀬健二, 中津良平: 創造的音楽表現を可能とする音楽演奏支援手法の検討—音機能固定マッピング楽器の提案, 情報処理学会論文誌, Vol.39, No.5, pp.1556-1567 (1998).
- 6) 西堀 佑, 多田幸生, 曾根卓朗: 遅延のある演奏系での遅延の認知に関する実験とその考察, 情報処理学会研究報告, Vol.2003, No.127, pp.37-42 (2003).
- 7) 柳川貴央, 北原鉄朗, 武田正之: 即興演奏における演奏補正システム, 情報処理学会第 64 回全国大会, pp.2.9-2.10 (2002).
- 8) 谷井章夫, 片寄晴弘: 音楽知識と技能を補うピアノ演奏システム INSPIRATION, インタラクティブシステムとソフトウェア IX: 日本ソフトウェア科学会 WISS2001, pp.199-200, 近代科学社 (2001).
- 9) Chew, E.: The Spiral Array: An Algorithm for Determining Key Boundaries, *Music and Artificial Intelligence (ICMAI 2002)*, pp.18-31 (2002).
- 10) Winograd, T.: Linguistics and the computer analysis of tonal harmony, *Journal of Music Theory*, Vol.12, No.1, pp.2-49 (1968).
- 11) <http://ist.ksc.kwansei.ac.jp/~katayose/members/yatsui/inspiration001.html>

(平成 16 年 4 月 13 日受付)

(平成 17 年 1 月 7 日採録)



谷井 章夫 (学生会員)

1979 年生。2003 年和歌山大学大学院システム工学研究科修士課程修了。現在、関西学院大学理学研究科博士後期課程在学中。2003 年より、科学技術振興機構さきがけ研究 21「協調と制御」領域片寄研究グループリサーチスタッフ。音楽情報処理、インタラクティブシステムの研究に従事。2003 年度情報科学技術フォーラム FIT2003 論文賞受賞。



片寄 晴弘 (正会員)

1991 年大阪大学大学院基礎工学研究科博士課程修了。工学博士。和歌山大学システム工学部助教授、関西学院大学助教授を経て、現在、同教授。2004 年度より、同大学文学研究科心理学専攻兼任。音楽情報処理、感性情報処理、HCI の研究に従事。2002 年度人工知能学会ベストプレゼンテーション賞、2003 年度 FIT 論文賞受賞。科学技術振興機構さきがけ研究 21「協調と制御」領域研究者。