

非調構成音を含む和音への対応を目的とした TPS の拡張 -ジャズ音楽理論への適用を目指して-

山口 直彦 †

† 工学院大学 工学研究科 情報学専攻 修士課程

管村 昇 ‡

‡ 工学院大学 情報学部 情報デザイン学科

1 はじめに

計算機音楽理論 Tonal Pitch Space(以下 TPS)[1] は、和音間の距離を定量的に求める方法を提唱している。

パークリーメソッドに代表されるジャズ音楽理論は和音(コード)進行を中心に楽曲の作曲、アナリゼ等を行うため、TPS を利用することでジャズ音楽理論を定量的に扱うことができると考えられる。

しかし現在の TPS はクラシック音楽理論を基に考えられているため、使用できる和音の種類が著しく制限されており、多様な和音を扱うジャズ音楽理論に TPS をそのまま適用することはできない。

また、現在発表されている計算機音楽理論の中で、ジャズ音楽理論が考慮されたものは少ない。

そこで本稿では TPS を構成するルールの一部を改訂することで扱える和音の種類を増やし、ジャズ音楽理論を計算機上で取り扱うための手法を提案、検討する。

2 TPS の概要

2.1 和音表現

TPS は 1 つの和音を 12 次元のベクトルで表現し、これを Basicspace(以下 BS) と呼ぶ。各要素は C B までの半音階にそれぞれ対応し、Level a~Level e の重みを持つ。BS は調名と根音の音度のペアで表現でき、以下ローマ数字表記による根音の音度と太字の英語表記による調名(長調は大文字、短調は小文字)をスラッシュではさんで表記する。図 1 に、I/C(コードネームでは Cmaj) を表した BS を示す。

Lv.a	C											
Lv.b	C									G		
Lv.c	C			E						G		
Lv.d	C		D	E	F					G	A	B
Lv.e	C	Db	D	Eb	E	F	Gb	G	Ab	A	Bb	B

図 1: I/C の BS

Level e はすべての音名、Level d には調構成音が含まれている。

Improve TPS to tackle non key constituent note

†Naohiko YAMAGUCHI ‡Noboru SUGAMURA

†Master's course, Major of Informatics, Kogakuin university

‡Faculty of Informatics, Department of Information Design, Kogakuin university

和音構成音のうち、根音は Level a となり、5 度音は Level b、3 度音は Level c となっている。ここで、あるレベルとなっている音名は、それよりも下位のレベルも含まなくてはならないとされている。すなわち、和音構成音(Level a~c) は必ず調構成音(Level d) である必要がある。

2.2 和音間距離の定義

TPS において、和音 x, y 間の距離は式 (1) に示すように、3 つの関数の和で定義され、距離が小さいほど x から y へ進行しやすいことを示す¹。

$$\delta(x \rightarrow y) = \text{region}(x, y) + \text{chord}(x, y) + \text{basicspace}(x, y) \quad (1)$$

$\text{region}(x, y)$ は調の 5 度圏 [3] において 2 つの調がどれだけ離れた弧に存在するかを示す。

$\text{chord}(x, y)$ は図 2 に示す和音の 5 度圏における距離によって region とほぼ同様に求まる。ただし 2 和音の調が異なる場合は、どちらかに調を合わせ、合わせられない場合は十分に大きな値を返すものとする。ここで和音の 5 度圏は、調を持つダイアトニックコード²を、隣同士が代理和音(構成音が似ており、同じ機能を果たす和音の集合)となるように並べたものである。

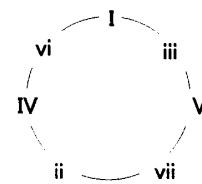


図 2: 和音の 5 度圏

$\text{basicspace}(x, y)$ は x, y それぞれの BS を比較し、異なっている成分の数を表す関数であり、和音の構成音がどれだけ変化したかを表す指標となる。

¹式 (1) の表記は原著 [1] ではなく先行研究 [2] による。

²各調構成音を根音とし、2 つおきの調構成音のみで構成される 3, もしくは 4 和音の集合

3 提案手法

3.1 BS の定義変更

TPS をジャズ音楽理論に援用する際の最大の問題は、非調構成音を含む和音、例えば (IV_7/c) といった和音を BS が表せないことにある。そこで提案手法では、非調構成音が和音構成音に含まれている場合、その音は調構成音となっていると見なして Level e を埋める。

また、TPS では 7 度音と 3 度音は等価値として扱われていたが、3 度音は和音の機能を決定する音として重要と言える。そこで、BS を構成する Level を 1 つ増やし、根音、5 度音、3 度音以外の調構成音 (具体的には 7 度音や 9 度音など) を Level d とする。それに伴い、調構成音と音名はそれぞれ Level e と Level f となる。

また短調における和音を表記する場合、短音階には自然短音階、和声的短音階、旋律的短音階の 3 種類がある。これらの短音階は楽曲の中で自由に使い分けられ、ジャズにおいてはマイナーにおけるスケール理論の根幹をなしている。そこで、短調における和音を表す場合、調構成音はこれら 3 つの短音階を複合したトニックマイナースケール [4] を利用する (図 3)。



図 3: トニックマイナースケール (ハ短調の例)

以上を踏まえ、提案手法の例として bVI_7/c (コードネームでは A_7^b) の BS を図 4 に示す。下線部は非調構成音が和音構成音となっているため、和音を BS で表すにあたって level e を埋めた部分である。

Lv.a										Ab
Lv.b			Eb							Ab
Lv.c	C		Eb							Ab
Lv.d	C		Eb		Gb					Ab A
Lv.e	C	D	Eb	F	<u>Gb</u>	G				Ab A Bb B
Lv.f	C	Db	D	Eb	E	F	Gb	G		Ab A Bb B

図 4: bVI_7/c の BS

3.2 chord(x,y) 関数

図 2 に示した和音の 5 度圏がダイアトニックコードしか含んでいないことから明らかなように、従来の $chord(x,y)$ 関数はダイアトニックコードしか対応しておらず、BS の定義を変更しただけではダイアトニックコード以外の和音について、 $chord(x,y)$ 関数の値を定めることができない。

ダイアトニックコードでない和音はダイアトニックコードの代理和音として機能している。そこで提案

手法では、与えられた和音がダイアトニックコードでない場合、与えられた和音とダイアトニックコードの $basicspace(x,y)$ の値を求め、最も $basicspace(x,y)$ の値が小さい和音と与えられた和音が連結されているとみなして chord 関数の値を求める。

例として I_7 が与えられたときの chord 関数の働きを図 5 に示す。chord 関数はまず各ダイアトニックコードとの $basicspace$ 関数が返す値を比較し、最も値が小さいダイアトニックコード (例では I) と与えられた I_7 が連結されているものとみなし、太い矢印で示されたルートをとって chord 関数の値として 4 を返す。

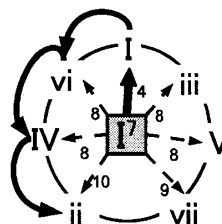


図 5: 提案手法による chord 関数の働き

4 まとめと今後の課題

TPS を非調構成音を含む和音に対応させるため、BS 及び $chord(x,y)$ 関数の定義を修正することで非調構成音を含む和音が含まれたコード列に対しても、和音間距離を求めることができるようになった。

現在 Ruby 言語を用いて提案手法の実装を進めている。今後、提案手法が算出する和音間距離の妥当性や、楽曲解析などでの有用性について、詳細な検証を進める予定である。

謝辞

本研究を進めるにあたり、プロベーシストである山口康広氏に音楽理論について多くのご教示を頂きました。感謝いたします。

参考文献

[1] Lerdahl, F.: Tonal Pitch Space (2004).
 [2] 坂本鐘期他: Tonal Pitch Space を用いた楽曲の和声解析 (2009).
 [3] 小方厚: 音律と音階の科学 (2007).
 [4] 大山大宣: JAZZ THEORY WORKSHOP 初級編 (1983).