



Title	音楽ハイパーテキストシステムに関する研究
Author(s)	有吉, 勇介
Citation	大阪大学, 1998, 博士論文
Version Type	VoR
URL	https://doi.org/10.11501/3144048
rights	
Note	

The University of Osaka Institutional Knowledge Archive : OUKA

<https://ir.library.osaka-u.ac.jp/>

The University of Osaka

音楽ハイパーテキストシステムに 関する研究

有吉 勇介

1998年1月

内容梗概

オーケストラや合唱などの多声部楽曲において一貫性のある演奏を行なうためには、指揮者と演奏者の間での楽曲構造の共有が重要である。しかし、従来の音楽情報処理分野の研究においては、楽曲構造の表現や視覚化では単声部の楽曲しか扱っていなかった。そこで本研究では多声部楽曲の演奏制作支援を目的として、構造化ハイパーテキストのデータモデルを拡張した楽曲構造モデルと視覚化機構の提案を行ない、それにもとづき楽曲構造を視覚化するハイパーテキスト楽譜システム *HypreScore* の試作を行なった。

楽曲の演奏プランは、フレーズ等の音列に対して音楽形式などの楽曲構造が与える音楽的意味によって決まる。しかし、楽曲分析手法の調査を行なったところ、音列の演奏プランを決める音楽的意味は従来から指摘された楽曲構造が直接その音列自身に与える場合に加えて、多段階分析により分解された楽曲構造が音楽的意味を与える場合、楽曲構造により関係づけられた音列や包含関係にある音列の音楽的意味が演奏プランに影響を与える場合があることが分かった。この分析にもとづいて従来の構造化ハイパーテキストモデルを拡張した楽曲構造モデルを提案した。楽曲構造モデルは、音列の階層構造を表現する木構造、音楽形式等の楽曲構造を表現する木構造、両者の間の音楽的意味付けの関係を表す属性付けリンクから構成され、演奏プランを決定する楽曲構造を簡潔に表現することができる。

楽曲構造モデルを従来の構造化ハイパーテキストと比較すると、交響曲のような大規模の楽曲の分析で使われる多段階の分析を表現するために、楽曲構造を分解した楽曲構造で意味付ける多段階の意味付けを表現できるように拡張している点と、音列の粒度により意味を与える楽曲構造が異なることを表現するために、粒度と対応する楽曲構造の関係を表現するノードとリンクを導入している点が特徴となっている。

つぎに、指揮者や演奏者の楽曲構造に対する注目の仕方について分析を行ない、演奏制作作業では指揮者や演奏者の楽曲構造に対する注目の仕方はその時々で異なり、また、演奏プランは注目している楽曲要素の周囲にある音列からも影響を受けるという特徴があることが分かった。そのため視覚化機構には、利用者が注目する粒度と楽曲構造の指定と、注目している音列に加えて周囲の音列が持つ属性の視覚化が求められる。

視覚化機構は選択・粒度・X Y軸・表示ルールの4種類のパラメータにより、利用者による楽曲構造に対する視点の違いに応じたビューを生成する。選択と粒度のパラメータは利用者が注目する音楽的意味と粒度をあらわし、X Y軸のパラメータは音楽的意味とビュー表示のX Y軸との対応をあらわし、表示ルールのパラメータは音楽的意味と表示属性との対応を指定する。これにより生成されたビューは利用者が注目する音列とその周囲の音列が持つ音楽的意味をビュー上の位置と音符の色などの表示属性によって表現する。

視覚化機構を従来のハイパーテキスト構造の視覚化と比較すると、視点指定の点では表示に反映できる論理構造の数、表示の詳細度、表示の煩雑さの抑制の点で優れている。また、注目の周辺の表示の点では、音楽的意味をビュー上の位置に加えて表示属性で表現している

優れている。

提案した楽曲モデルと視覚化機構にもどづきプロトタイプの実装を行ない実装可能性を示すとともに、音楽専門家による試用では良好な評価を得た。

関連発表論文

学術論文誌(査読あり)

- 1) Yusuke Ariyoshi, Shinji Shimojo, Hideo Miyahara, ``Hyperscore: A Design of a hypertext model for musical expression and structure", Journal of New Music Research , Vol.24 No.2, pp.130-147, June 1995 .
- 2) 有吉勇介, 下條真司, 宮原秀夫, ``ハイパーテキスト楽譜のための楽曲構造モデルと視覚化機構", 電子情報通信学会論文誌D-II (条件付採択) .

国際会議(査読あり)

- 3) Yusuke Ariyoshi, Shinji Shimojo, Hideo Miyahara, ``Hyperscore: A Design of a hypertext model for musical expression and structure", Program of the International colloquium New Music Research Festival 30 Years IPFM, pp.10-11, Ghent, October 1994.

学術研究集会報告 (査読なし)

- 4) 藤川和利, 梶本雅人, 有吉勇介, 下條真司, 松浦敏雄, 西尾章治郎, 宮原秀夫, ``マルチメディアプレゼンテーションシステム Harmony", 電子情報通信学会 ヒューマンコミュニケーション研究会, July 1990.
- 5) 上原征樹, 有吉勇介, 下條真司, 西尾章治郎, 宮原秀夫, ``音楽情報の多面的ビューについて", 電子情報通信学会 データ工学研究会, March 1992.
- 6) 藤川和利, 有吉勇介, 下條真司, 松浦敏雄, 西尾章治郎, 宮原秀夫, ``協調型プレゼンテーション作成支援システム HarmonyII", 日本ソフトウェア科学会 ソフトウェア研究会, April 1992.
- 7) 有吉勇介, 島村栄, 下條真司, 宮原秀夫, ``マルチメディアを用いたミュージカル制作支援システム構想", 情報処理学会 情報メディア研究会, November 1992.
- 8) 有吉勇介, 下條真司, 宮原秀夫, ``音楽ハイパーテキストシステム: HyperScore", 情報処理学会 音楽情報科学研究会, August 1994.

目次

1 序論	1
2 楽曲構造モデル	6
2.1 楽曲構造の分析	6
2.2 楽曲構造モデル	12
2.2.1 楽曲要素	12
2.2.2 楽曲論理構造	13
2.2.3 属性付け	14
2.2.4 分析構造	16
2.2.5 要件との対応	18
3 視覚化	19
3.1 視覚化への要求	19
3.2 視覚化機構	20
4 プロトタイプシステムの実装	23
4.1 データモデル	23
4.1.1 ハイパーテキストモデルの適用	23
4.1.2 データモデルの概要	23
4.2 システム概要	25
5 従来研究	29
5.1 演奏制作支援	29
5.2 ハイパーテキストモデル	30
5.2.1 Set-to-set	30
5.2.2 フレーム関係軸モデル	32
5.2.3 ハイパーメディアモデルと楽曲構造モデル	34
5.3 視覚化	35
5.3.1 視点	35
5.3.2 コンテキスト	36
6 評価	37
7 結論	38
謝辞	39
参考文献	40

1 序論

演奏制作の工程は楽譜に基づいて練習し演奏を行なうプロセスと一般には思われていることが多い。しかし、実際は楽譜から音楽形式や和声構造などの楽曲構造を読み取り、その楽曲構造をもとに曲の各々の部分をどのように演奏するかという演奏プランを決める工程が、練習を行う前に存在する。そして練習や演奏はこの演奏プランに基づいて行われ、また、練習や演奏の結果に基づいて演奏プランは洗練されていく(図1)。

本研究で対象としているオーケストラや合唱などの多声部楽曲では、この演奏制作は指揮者と演奏者の分業作業で行われる(図2)。まず指揮者は楽曲構造の分析を行い、その結果に従って楽曲全体の演奏プランを作成する。次に各声部の演奏者は指揮者の楽曲構造と演奏プランに基づいて自声部の詳細な演奏プランを作成し、練習と演奏を行なう。

演奏プランのもととなる楽曲構造は、音楽形式やリズム構造などの複数の構造がからみあつた複雑なものである。また、楽曲構造は楽譜に明示されておらず、楽譜から読み取るときの解釈の違いにより異なるという曖昧性がある。楽曲構造は楽譜への書き込みとして表現されるが、複雑性と曖昧性があるため書き込みで表現することには限界がある。

さらに、多声部楽曲は分業のため指揮者から演奏者へ楽曲構造へ伝える必要がある。楽曲構造がうまく伝わらなかった場合、声部間で表現がばらついたり、演奏者と指揮者で呼吸が合わないなど、一貫性の無い演奏になってしまう。一方、声部が複数あるため楽曲構造はいっそう複雑である。そのため、多声部楽曲では、楽曲構造の楽譜上への書き込みによる表現の限界はより深刻である。

例えば、図3は合唱曲の指揮者の楽譜であり、古典的な楽曲分析手法を用いて分析した結果と演奏プランが書き込まれている。図4はLerdahlらの理論[1]による現代的な楽曲分析の例である。2の部分がもともとも楽曲の楽譜であり、3・4・5は楽譜から抽出したリズム、グループ、和声の構造である。この3つの構造をさらに分析した結果が1の木構造であり、[1]ではこの1の木構造をさらに分析し演奏プランを決定する。

そこで本研究では、演奏制作作業を支援するために、複雑な楽曲構造を表現する楽曲構造モデルと、楽譜上に明示されておらず曖昧性のある楽曲構造を分かりやすく視覚化する視覚化機構を備えた楽譜ハイパーテキストシステム HyperScore を提案する。

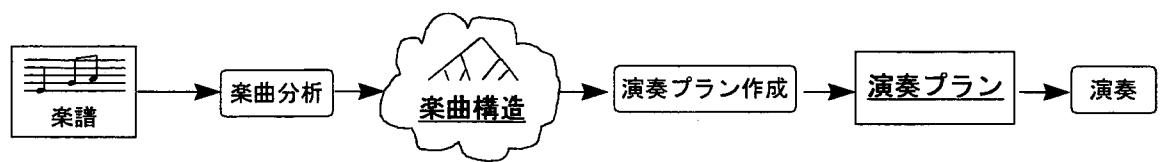


図 1 演奏制作工程

Fig.1 Process of making performance.

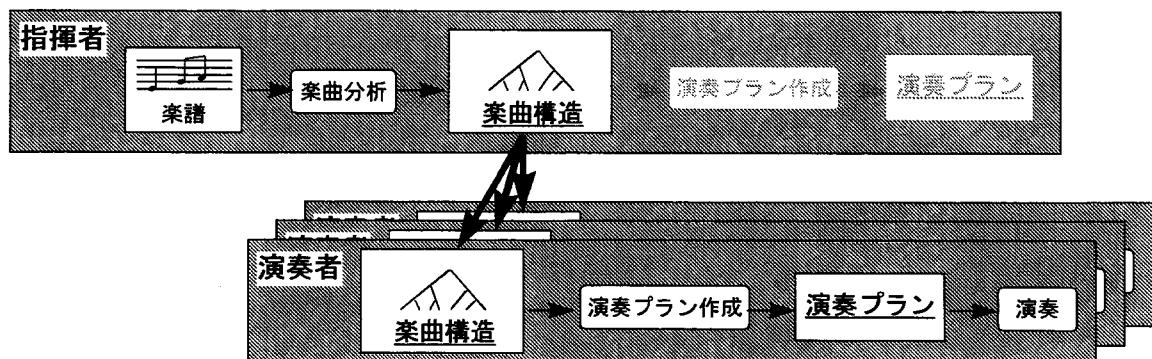


図 2 多声部楽曲の演奏制作工程

Fig.2 Process of making performance for multi-voice score.



図 3 指揮者の楽譜に対する書込

Fig.3 Example of conductor's score.

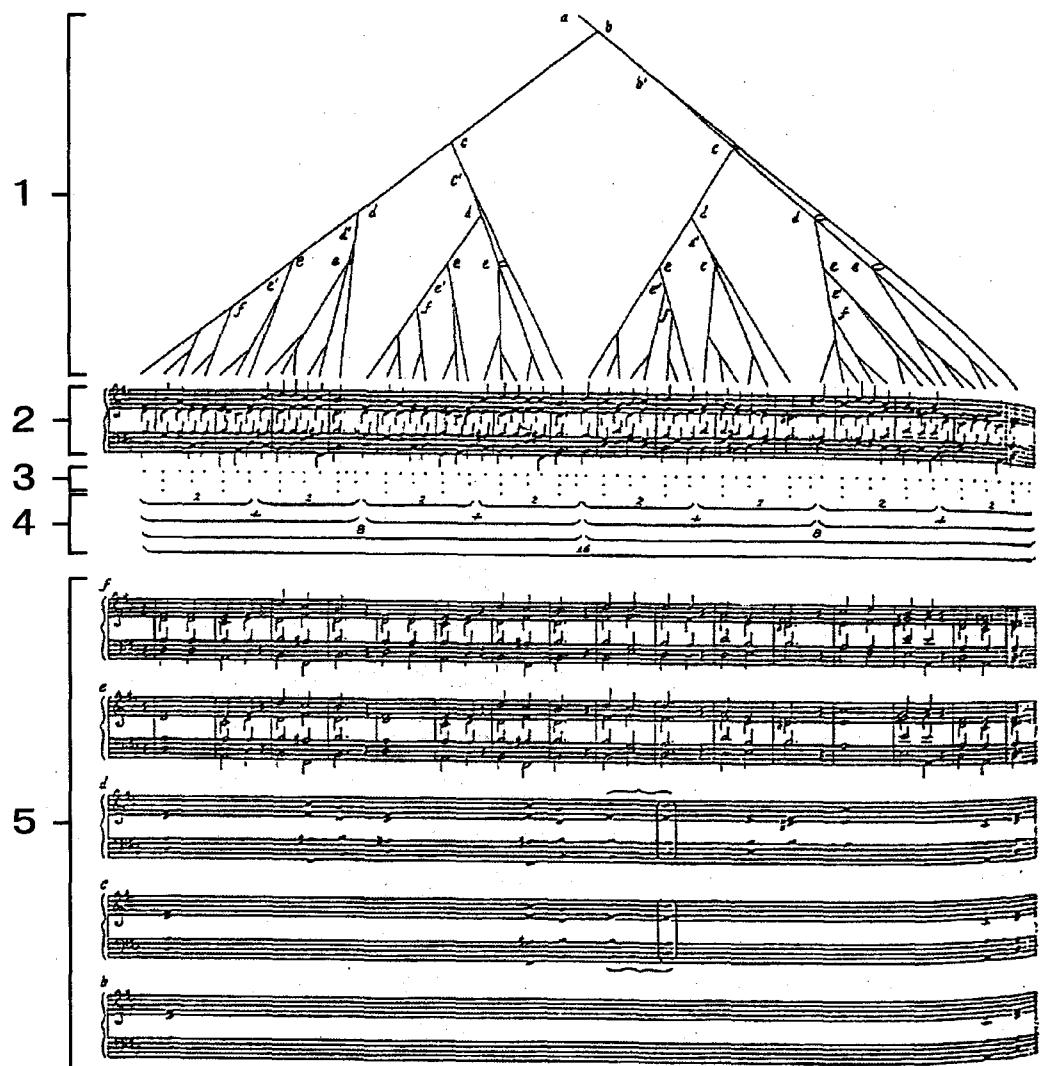


図 4 Lerdahl らの理論[1]による楽曲分析

Fig. 4 Analysis of score from [1].

HyperScore の楽曲構造モデルは楽曲分析の手法の分析に基づいて構造化ハイパーテキストモデルを拡張したものとなっており、楽曲構造の 3 つの要素、楽曲要素・楽曲論理構造・属性付けを表現する。次に、指揮者や演奏者の楽曲構造に対する注目のしかたを分析にもとづいて、楽曲構造の視覚化機構を提案する。視覚化機構は 4 つのパラメータ、選択・粒度・XY 軸・表示ルールの指定により複雑な楽曲構造を様々な視点から見たビューを生成する。

2 楽曲構造モデル

HyperScore の目的である演奏プラン作成の支援のためには、楽曲構造のうち特に演奏プラン作成に関する部分を簡潔に表現する事が重要である。そこで、ここではそのための要件の分析を行い、つぎに楽曲の構造を柔軟に表現するための楽曲構造モデルを提案する。

2.1 楽曲構造の分析

楽曲分析では音楽形式や和声構造などの音楽構造が抽出される。(以下では、音楽構造が組み合わさった楽曲の論理構造全体を楽曲構造と呼び、個々の音楽構造と区別する)。この音楽構造が音やフレーズなどの音列に音楽学的意味を与える。そして、音列の演奏プランは、この音楽学的意味によって決まると考えられている[1]。

我々は指揮者と演奏者へのインタビュと楽曲分析の手法の調査を行なった。それにより音列の演奏プランを決める構造は、従来から指摘されていた音楽構造から音列への音楽的意味付けに加えて、要素音楽構造から音楽構造への意味付け、楽曲要素の包含関係、音楽構造による関係づけの合計 4 種類あることが分かった。

1. 音楽構造から音列への音楽的意味付け

例えば、図 5 の左上端のフレーズに対して、「このソプラノのフレーズは、第一主題の再現部の最初(音楽的意味)なので、はっきり目立つように演奏しよう(演奏プラン)」というような演奏プランの選択が行われる。

このように演奏プランを決めているのが、音楽構造が音列に音楽的意味を与える構造である。例では、音楽形式という音楽構造がフレーズに第一主題の再現部の最初という音楽的意味を直接与えている(図 6)。

この例の演奏プランは「はっきり目立つように」という方針だけで具体的でないプランであったが、図 5 のように 1 つの音列に対して複数の音楽構造から異なる意味が与えられることにより、演奏プランがより具体的なものに絞り込まれていく。

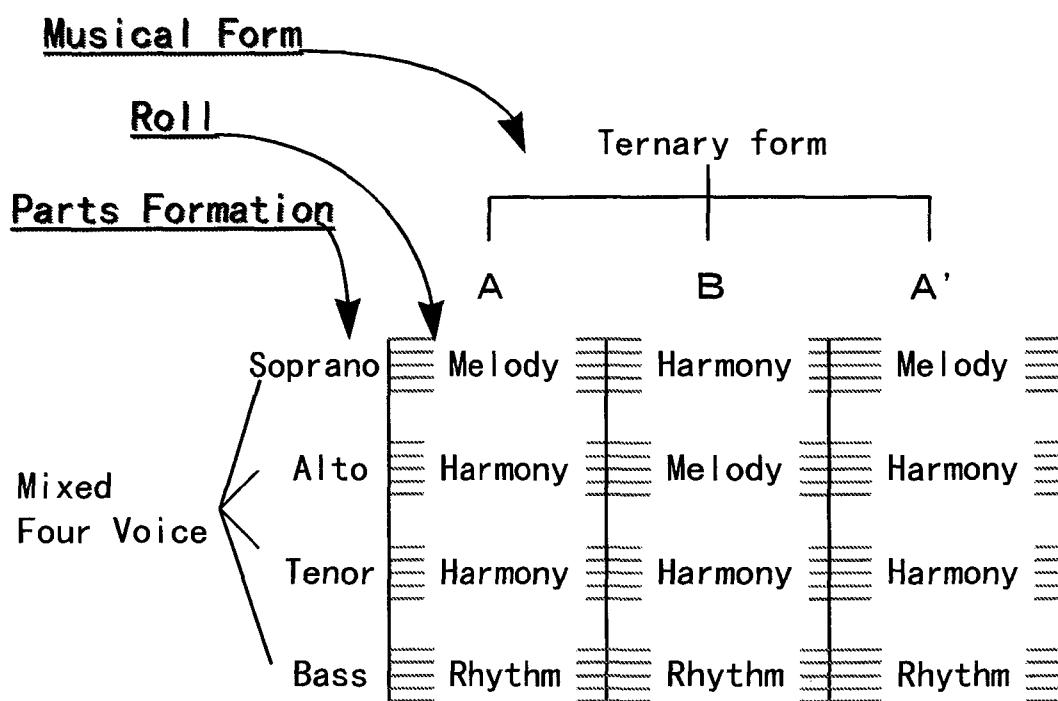


図 5 楽曲分析の例

Fig.5 Analysis of score.

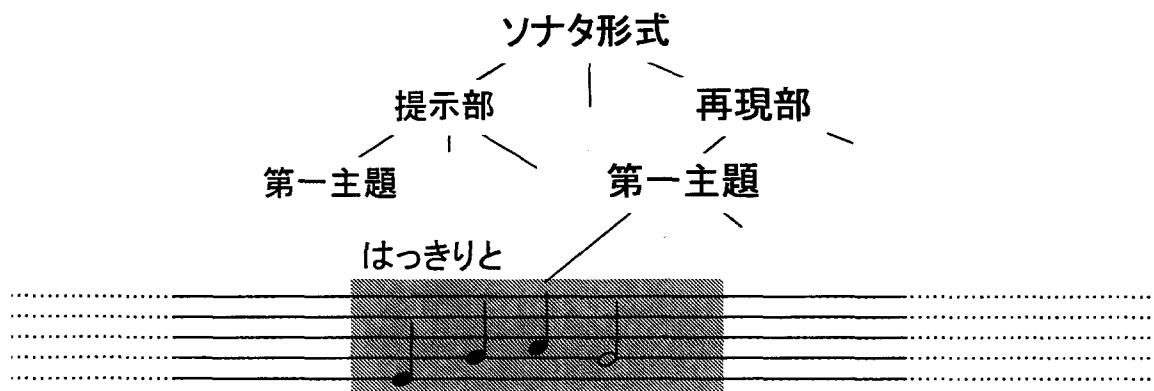


図 6 音楽的意味と演奏プラン

Fig.6 Musical semantics and performance plan.

2. 要素音楽構造から音楽構造への意味付け

HypreScore が対象とする多声部楽曲は大きく複雑な音楽構造を持つものが多い。例えば交響曲などは声部が数十ありフレーズは百以上になる。音楽構造が大きくなるほど演奏プランを導き出すことが難しくなるため、大きな音楽構造は単純な音楽構造の組み合わせに分解して演奏プランを決めやすくする多段階の分析が行なわれる（以下では音楽構造を分解した結果の単純な音楽構造を要素音楽構造と呼ぶ）。

簡単な例をあげると、図 7 は複合三部形式の音楽構造が三部形式と二部形式の要素音楽構造に分解できることを示している。このときフレーズ C 1 の演奏プランは、3 部形式の 3 番目のブロックと 2 部形式の 1 番目のブロックという両方の音楽的意味からくる制約に合ったものが選ばれる。

このように多段階分析を行なうことにより、演奏プランを直接導き出すことが難しい大きな音楽構造であっても、より単純な要素音楽構造が与える音楽的意味から演奏プランを決めることができるようになり、大きな楽曲であっても演奏プランを決めることが簡単になる。

多段階分析が行なわれた場合、要素音楽構造からもとの音楽構造に音楽的意味が伝えられ、その音楽的意味が音楽構造から音列に与えられて演奏プランが決まる。このとき要素音楽構造から音楽構造に音楽的意味が伝えるのが、要素音楽構造から音楽構造への意味付けの構造である。

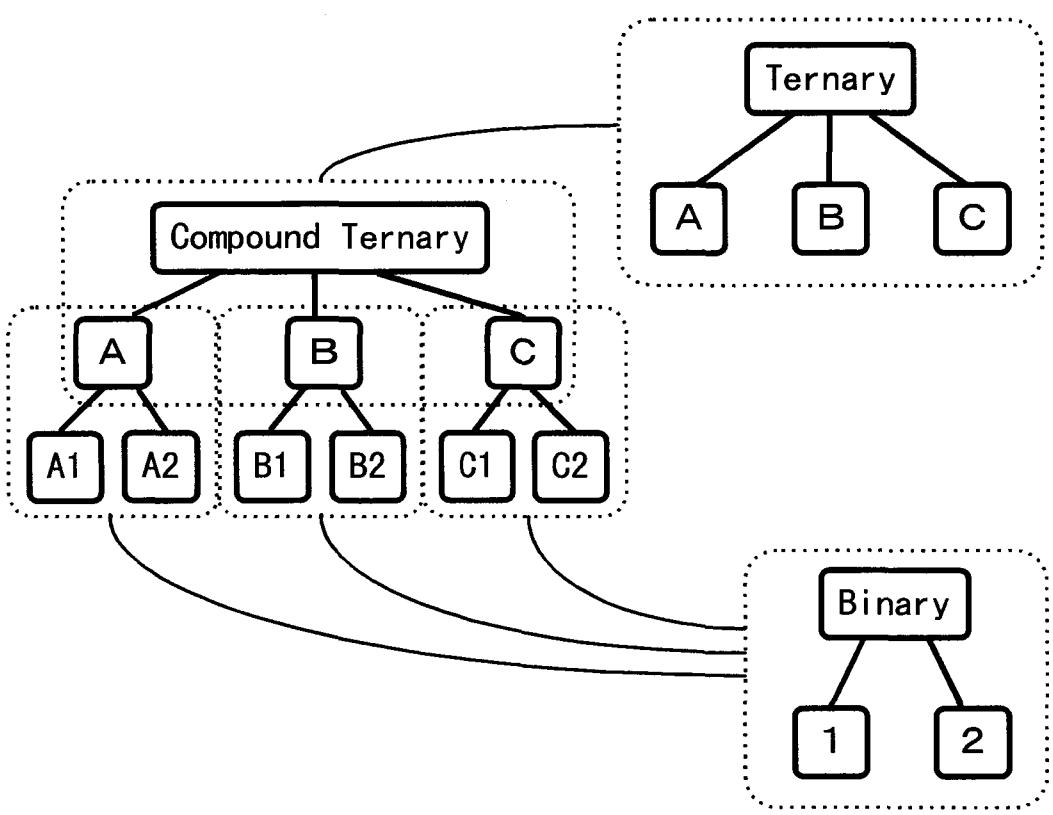


図 7 音楽構造と要素音楽構造

Fig.7 Music Structure and Element Music Structure.

3. 音列の包含関係

音楽的意味を持つ音列には、粒度(Granularity)の違いにより音、モチーフ、フレーズ等の種類がある。粒度の大きな音列と小さな音列は包含関係あり、粒度の大きい音列は粒度の小さな音列から構成される。

粒度の異なる音列に対しては、音楽的意味を与える音楽構造も異なる。図5ではフレーズに対しては声部構成・音楽形式・役割の3種類の音楽構造から音楽的意味が与えられていた。一方、粒度の小さい音レベルに対しては和音中の位置やフレーズ中の位置などの音楽構造から、例えば「Iの和音の第三音」や「フレーズの開始音」といった音楽的意味が与えられる。

そして、包含関係にある音列の音楽的意味や演奏プランによって音列の演奏プラン決定が影響を受ける場合がある。

例えば音レベルの分析で、ある音が「Iの和音の第三音」と意味付けされた場合、和音は第三音の音程によって長調か短調かが決まるため、その音は正確な音程の演奏が要求される。そのため、その音を含むフレーズの演奏プランとしては音程が不安定になりやすいものは除かれる。

つまり、演奏プランは音楽構造が与える音楽的意味だけから決まるのではなく、包含関係にある音列の音楽的意味や演奏プランと相性の良いものが選ばれる。

このように、音列の演奏プランの決定には音列の包含関係という構造も影響を与える。

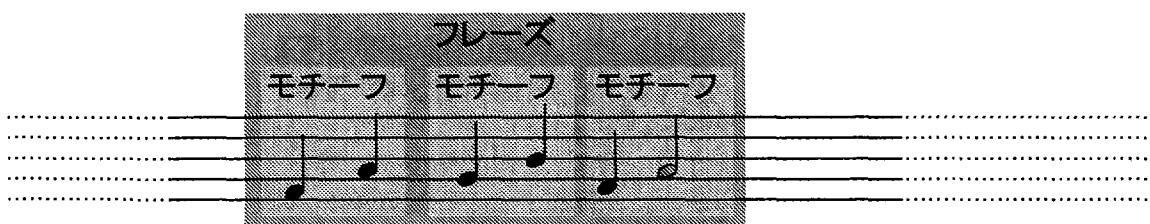


図 8 音列の包含関係

Fig.8 Inclusion of sequence of notes..

4. 音楽構造による関係づけ

例えば図5の一番左上のフレーズは声部構成・音楽形式・役割の音楽構造から、それぞれソプラノ、Aの第1フレーズ、メロディという音楽的意味を与えられ、その下のフレーズはアルト、Aの第1フレーズ、ハーモニという意味を与えられていた。このフレーズの演奏プランは、役割がハーモニであることから、メロディより目立たず、メロディに音量・テンポ等を合わせやすいものが選ばれる。

この例では役割という音楽構造で関係づけられたメロディの音列の演奏プランと相性のよい演奏プランが選ばれている。

つまり、音列の演奏プランは、その音列自身の音楽的意味だけではなく、その音列と音楽構造により関係づけられている音列の音楽的意味や演奏プランの影響を受けて決定される。このように、音列の演奏プランの決定には、音列間の音楽構造による関係づけという構造も影響を与える。

2.2 楽曲構造モデル

楽曲構造モデルの目的は、楽曲構造の特に音列の演奏プランを決める構造を表現することである。そのため前記の4種類の構造を表現することが楽曲構造モデルの要件となる。楽曲モデルは、楽曲要素・楽曲論理構造・属性付け・分析構造の大きく4つの要素から構成され、これらにより音列の演奏プランを決める構造を表現する。

2.2.1 楽曲要素

楽曲構造モデルでは演奏プラン作りの単位となる音列を楽曲要素ノードと包含関係のリンクで表現する。楽曲要素ノードは、フレーズやモチーフなどの音楽的意味を持つ音列を表現する。包含関係のリンクは粒度の異なる音列の包含関係を表現する。これにより、楽曲要素は楽曲要素ノードが包含関係のリンクでつながれた木構造を形成する（図9）。

Granularity

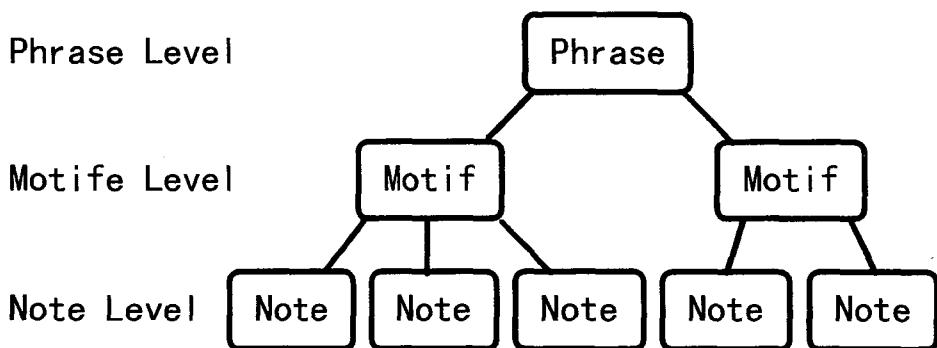


図 9 楽曲要素

Fig.9 Score element.

2.2.2 楽曲論理構造

楽曲構造モデルでは、音楽形式などの音楽構造を音楽属性ノードと属性値間の階層関係をあらわすリンクで表現する。

音楽属性ノードは、音列に与える音楽的意味を表現する。

階層関係リンクは、音楽的意味の間の関係を表現する。

例えば図10は混声4部の声部構成の楽曲論理構造である。

混声4部構成は男声部と女声部から構成され、それぞれ男声はテナーとバス、女声はソプラノとアルトの声部から構成される。

この例では各4声部や男声部・女声部といった音楽的意味を音楽属性ノードで表現し、男声部はテナーとバスで構成されるという関係を音楽属性ノード間を階層関係リンクでつなぐことで表現する。そのため音楽構造は音楽属性ノードが階層関係リンクでつながれた木構造になる。

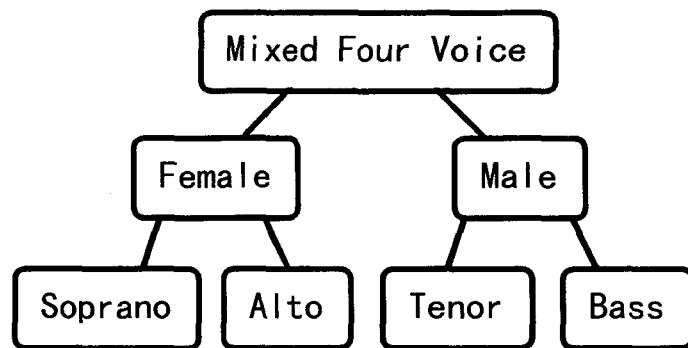


図10 楽曲論理構造（声部構成）

Fig.10 Logical structure(parts formation).

2.2.3 属性付け

楽曲構造モデルでは、音楽的意味を与える関係を属性付けリンクで表現する。この属性付けリンクにより、音楽構造から音列への意味付と、多段階分析における要素音楽構造から音楽構造への意味付けを表現する。

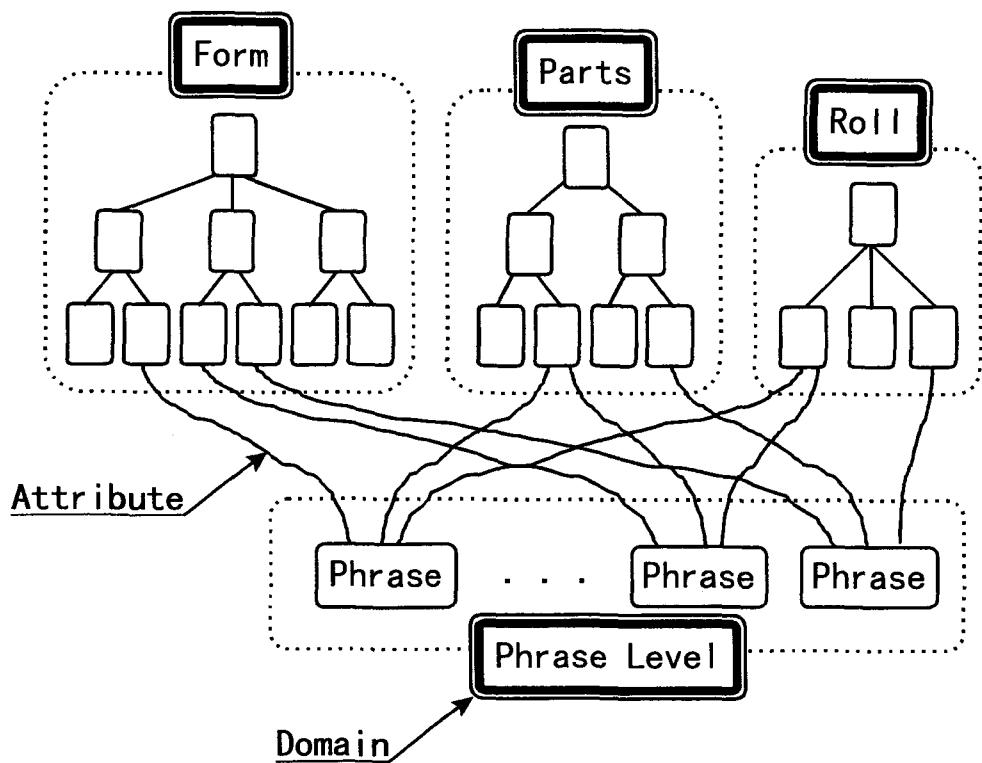


図 1 1 属性付け

Fig.11 Attribute link.

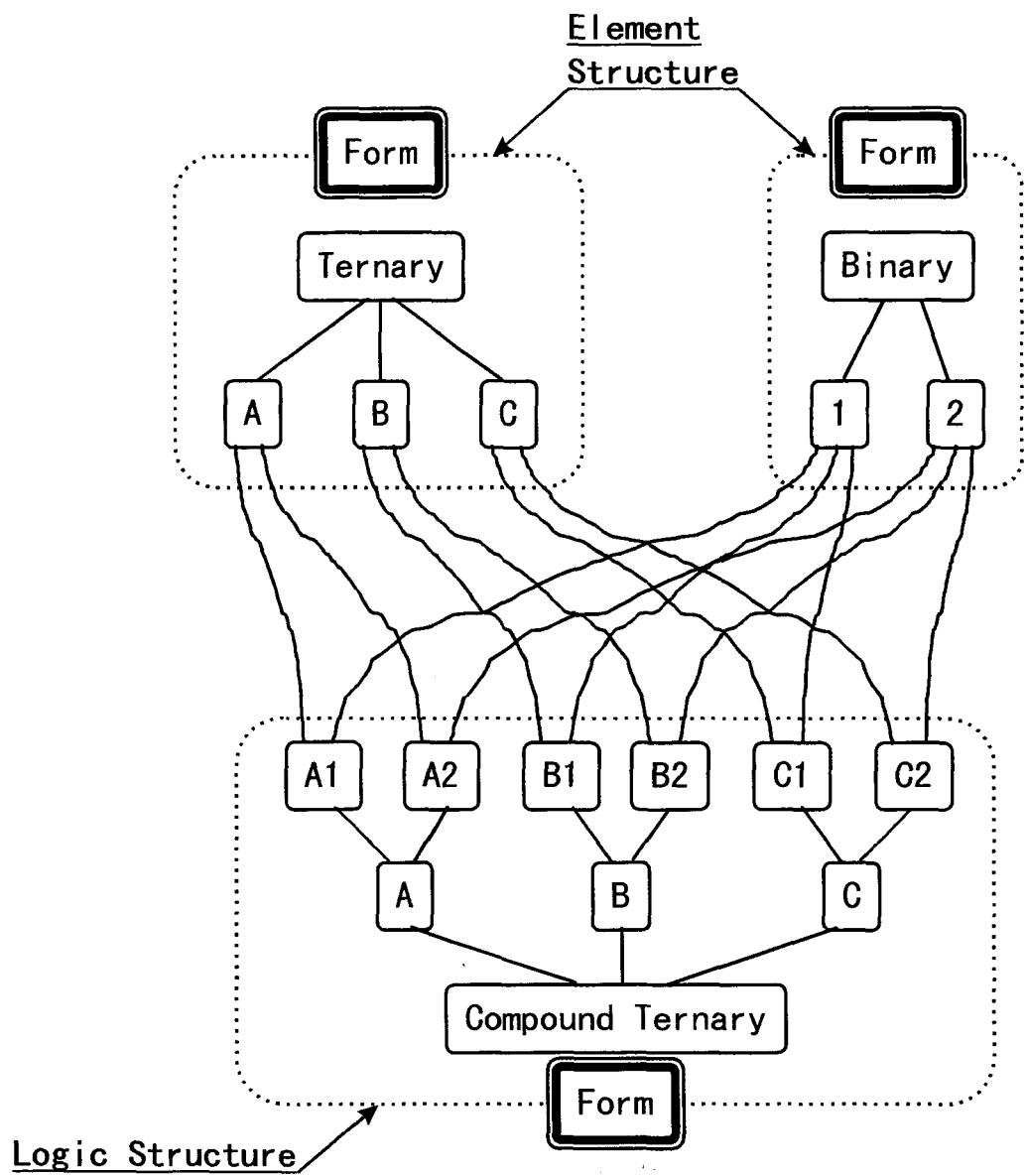


図12 多段階の属性付け

Fig.12 Multi-level attribute linking.

2.2.4分析構造

同じ楽曲であっても指揮者の解釈や楽曲分析手法のちがいによって、楽曲分析結果の楽曲構造が異なることがある。楽曲構造モデルでは分析構造により、指揮者による分析の仕方のちがいを表現する。

分析構造は、指揮者が分析した音列の粒度レベル、各粒度レベルをどの音楽構造で意味付けしたか、多段階分析では音楽構造をどう要素論理構造に分解したかを、粒度レベル、論理構造、要素論理構造を表現するノードと、粒度レベルと意味付けする音楽構造の対応関係、多段階分析による音楽構造と分解した音楽構造との関係を表現するリンクにより表現する。

図13はある楽曲の分析の構造を示したものである。この楽曲分析はフレーズと音の2種類の粒度で分析されており、フレーズは役割(Roll)・音楽形式(Form)・声部構成(Parts)の3つの視点で分析され、音は和音(Chord)とメトリックス(Metrics)の2つの音楽構造で分析されている。いくつかの音楽構造は多段階分析により要素音楽構造に分解されている。

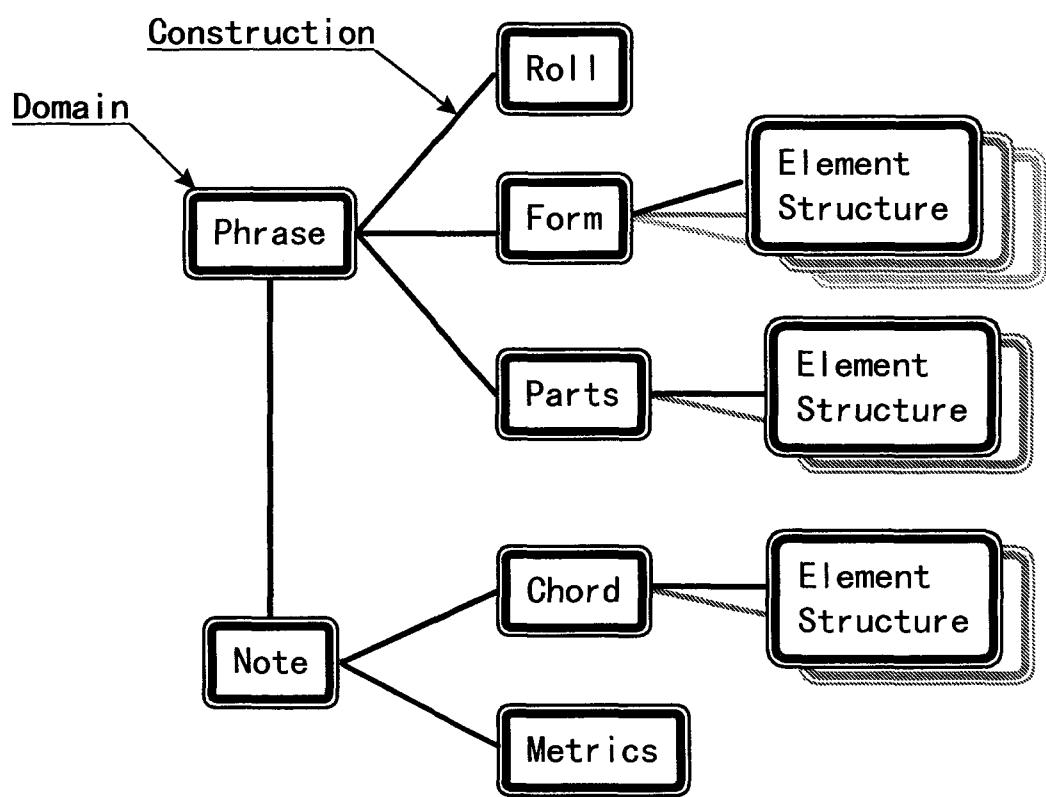


図 13 分析構造

Fig.13 Analysis structure.

2.2.5要件との対応

楽曲構造モデルでは音列の演奏プランを決める4種類の構造を次のように表現する。

- 音楽構造から音列への音楽的意味付けは、楽曲要素へ論理構造中の音楽属性から属性付けのリンクを張ることで表現する。
- 要素音楽構造から音楽構造への意味付けは、論理構造中の音楽属性へ要素論理構造中の音楽属性から属性付けのリンクを張ることで表現する。
- 音列の包含関係は楽曲要素のパートオブリンクが表現する。
- 音楽構造による関係づけは、論理構造と属性付けリンクで表現する。

3 視覚化

3.1 視覚化への要求

HyperScore の目的である演奏プラン作成の支援のためには、楽曲構造の視覚化機構には以下に挙げる 2 つの項目を実現することが望まれる。

第一の要件は、人によって異なる注目に対応した視覚化である。楽曲構造のうち注目する部分は指揮者や演奏者という人の役割によって異なる。

例えば、指揮者はいつどの声部にどんな合図を送るかを決めるために、主題が声部間で受け継がれ変奏されていく楽曲の構成をつかもうとする。一方、演奏者は自分が主旋律の時は演奏をリードするが、そうでない時はテンポや音量などを調節して主旋律に合わせる必要がある。そのため、その時々の自声部の役割と、どの声部が主旋律なのかに注目する。このように人の役割が異なれば楽曲構造のうち注目する部分も異なる。

また同じ演奏者でも、楽曲構造の概要をつかみたい場合は粒度の大きい楽曲要素のレベルで属性に注目するが、詳細を知りたい場合は粒度の細かいレベルの楽曲要素の属性に注目する。このように、楽曲構造のうちで注目する部分は人や場合によって異なる。

この注目の違いは、楽譜上での注目する部分が違うだけでなく、どの粒度で注目するか、どの楽曲論理構造から与えられる属性に注目するのかが異なると考えられる。そのため、視覚化機構は人や場合によって異なる見方に合わせて視覚化するために、見方に合わせて注目する粒度と楽曲論理構造を指定できることが必要である。

二番の要件は、注目している楽曲要素自身の属性だけでなく、周囲との関係を視覚化することである。楽曲要素の演奏プランを決めるのは、その楽曲要素自身が持っている属性だけではなく、周囲の楽曲要素との関係、言い換えると注目している楽曲要素がどのような楽曲要素に囲まれているかという一種のコンテキスト情報が重要である。

この要件は 2.1 節で述べた音列の演奏プランを決める 3 番目と 4 番目の構造の視覚化にあたる。3 番目の包含関係にある楽曲要素の場合、演奏プランは注目している楽曲要素自身が持つ属性だけではなく、包含関係にある周囲の楽曲要素が持つ属性や演奏プランの影響を受けていた。また、4 番目の音楽構造による関係づけ例では、ハーモニのフレーズの演奏プランはメロディのフレーズとテンポや音量などが合わせやすいものが選ばれ、この場合も演奏プランの決定が楽曲要素自身の属性だけでなく周囲の楽曲要素の属性や演奏プランに影響を受けている例である。

このように楽曲要素の演奏プランを決定するにはコンテキスト情報も重要である。そのため、視覚化機構は、注目している楽曲要素が持つ属性だけでなく周囲の

コンテキスト情報も視覚化する必要がある。

HyperScore では上記の 2 つの要件に加えて、対象と想定している利用者である音楽家が計算機利用に習熟していないことが考えられるので、楽譜のメタファからかけ離れたものは用いないで、利用者に分かりやすい（親しみやすい）楽譜システムをもとにして、その拡張になるように設計を行なった。

3.2 視覚化機構

データベースでは、その対象をモデル化した論理的なデータ構造（概念スキーマ）の一部を特定の利用者に対して特定の見方で切り出したものをビューと呼ぶ。本研究でもこれにならい、楽曲構造の一部を利用者の見方に応じて切り出し、視覚化した表示をビューと呼ぶ。

HyperScore の視覚化機構は 4 つのパラメータ、XY 軸、表示ルール、選択、粒度を指定することにより、利用者の楽曲構造に対する見方に応じたビューを生成することで先の二つの要件を満たす。

- 選択： ビュー表示上にどの楽曲要素を表示するかを属性で指定する。例えば旋律(melody)という属性を持つ楽曲要素だけを表示するといった指定ができる。
- 粒度： 表示の詳しさを指定する。例えば、曲全体の概要を知りたいときにはフレーズを表示の最小単位とし、一部分を詳しく見たいときは音符を最小単位として表示することができる。
- XY 軸： ビュー表示の X 軸・Y 軸に対応させる楽曲論理構造を指定する。これによりビュー表示上での楽曲要素の配置が決まる。例えば、普通の楽譜では Y 軸は声部構成が対応しているが、役割を対応させると、メロディ・ハーモニ・リズムと役割ごとに段組したビュー表示が生成される。
- 表示ルール： 楽曲要素が持つ属性と表示属性との対応を指定する。これにより XY 軸に対応されなかった楽曲論理構造を視覚化することができる。例えば、フレーズの役割と音符の色を対応付けることで、フレーズの役割を視覚化することができる。

例えば、図 1.4 は粒度にフレーズレベルを指定し、X 軸に音楽形式、Y 軸に声部構成を指定した場合のビューであり、通常の楽譜と同じ段組になっている。図 1.5 は Y 軸の指定を役割に変更した場合のビューであり、1 段目はメロディライン、2 段目がハーモニ、3 段目がリズムと段組が役割ごとになっている。また、表示ルールでソプラノに灰色の背景色を指定しているため、ソプラノの役割が分かりやすいように強調されている。

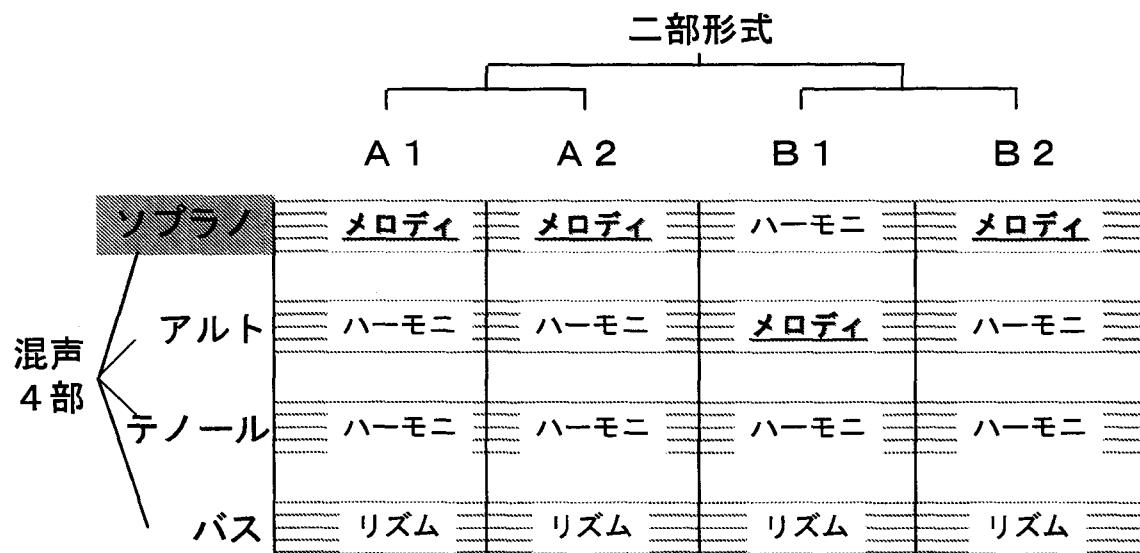


図 1 4 X Y 軸と表示ルール(1)

Fig. 14 XY-axis and display rule(1).

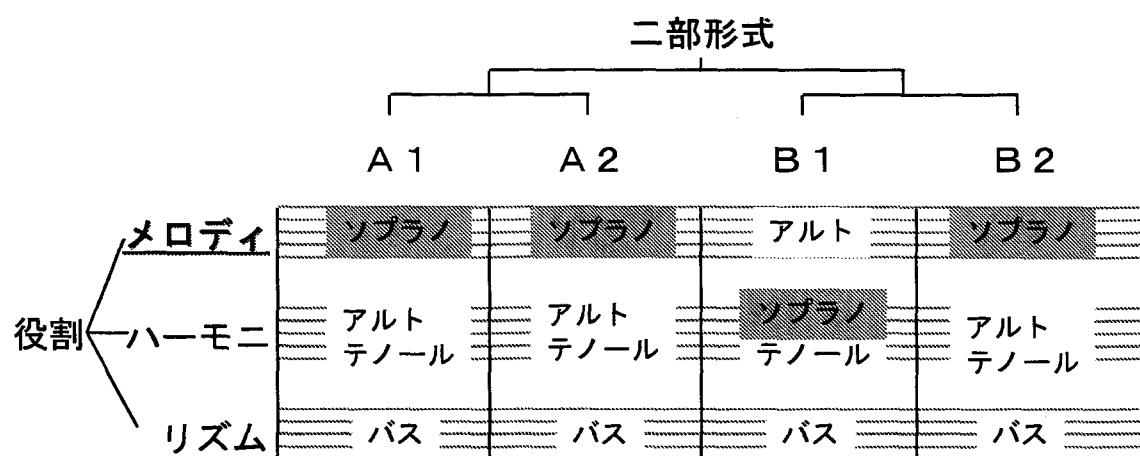


図 1 5 X Y 軸と表示ルール(1)

Fig. 15 XY-axis and display rule(1).

視覚化機構は、注目している楽曲要素を選択と粒度によって指定し、XY軸で注目している属性をビュー表示上の楽曲要素の位置で視覚化し、表示ルールではXY軸に割り当てられなかった楽曲論理構造を表示属性を使って視覚化する。このようにして利用者の注目に合わせたビューを生成し、一番目の要件に対応する。

二番目の要件であるコンテキスト情報の提示に対しては、XY軸の指定によってビュー表示上の楽曲要素の配置は楽曲要素の持つ属性を表しており、位置関係は楽曲論理構造上での属性の関係を反映している。さらに、表示ルールによって音符の色などの表示属性によっても楽曲要素の音楽的属性を示すことができる。つまり、生成されたビュー表示は、ビュー表示上の空間配置と表示属性でコンテキスト情報を視覚化し利用者に伝える。

4 プロトタイプシステムの実装

ここではこれまで述べた楽曲構造モデルと視覚化機構に基づいて実装した HyperScore のプロトタイプについて述べる。

4.1 データモデル

4.1.1 ハイパーテキストモデルの適用

楽曲構造モデルを構造化ハイパーテキストモデルをもとに設計した。構造化ハイパーテキストモデルは柔軟な表現能力を持ち、従来から文書やマルチメディア・コンテンツの論理構造を表現するのに用いられている。

ハイパーテキストモデルの基本構成要素は、情報の実体を表現するノードと情報間の関係を表現するリンクである。構造化ハイパーテキストでは、リンクやノードが拡張されてリストや木などの構造を持つ。

4.1.2 データモデルの概要

プロトタイプシステムのデータモデルでは2種類のノードと4種類のリンクにより楽曲構造を表現している。2種類のノードは次のリージョンとドメインである。

- リージョンは情報の実体と分類属性値を表現するノードであり、楽曲要素と音楽的属性を表現する。
- ドメインは分析の視点を表現するノードであり、分析構造の楽曲要素の粒度レベルや楽曲論理構造、要素論理構造を表現する。

従来の構造化ハイパーテキストでは、情報の実体を表わすノードと分類属性値を表わすノードは別のオブジェクトであった。しかし、音楽属性のノードは、音楽構造から音列への音楽的意味付けでは属性付けリンクの始点になるが、要素音楽構造から音楽構造への意味付けでは属性付けリンクの宛先になる。そのため、プロトタイプシステムのデータモデルでは属性付けリンクの始点と宛先のオブジェクトは同じクラスで表現している。

4種類のリンクは次のパートオブ、属性付け、メンバーオブ、コンストラクションである。

- パートオブのリンクは、楽曲構造における楽曲要素ノード間の包含関係と

楽曲論理構造における音楽属性間の階層関係を表現する.

- 属性付けのリンクは情報の実体と属性値の関係を表現するリンクであり, 意味付けの関係を表現する.
- コンストラクションのリンクは分析構造を表現するドメイン間のリンクである.
- メンバーオブのリンクはドメインとそのドメインに属するリージョンの関係を表わす. 例えば, 声部構成のドメインと各声部のリージョンの間を結び付ける.

このプロトタイプシステムのデータモデルをクラス図[12]で表現すると図16のようになる.

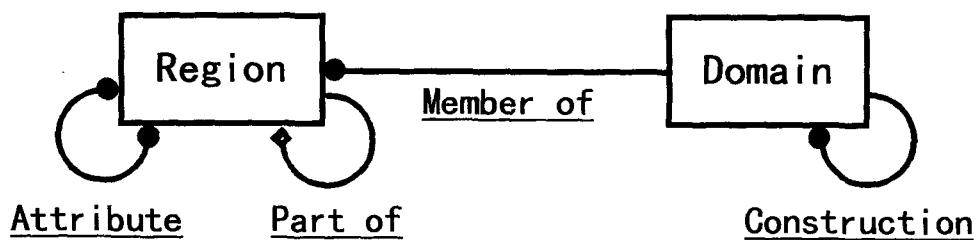


図 16 スキーマ

Fig.16 Scheme of musical structure model.

4.2 システム概要

プロトタイプシステムの実装は NEXTSTEP 上に Objective-C 言語によって行った。また、楽譜の描画には NEXTSTEP 上の楽譜システム Nutation[2]を使用した。

プロトタイプシステムは図 17 のように GUI 部・ビュー管理部・データ管理部の 3 つのモジュールから構成されている。

GUI 部ではユーザからビュー生成のためのパラメータを受け、ビュー表示の描画を行う。

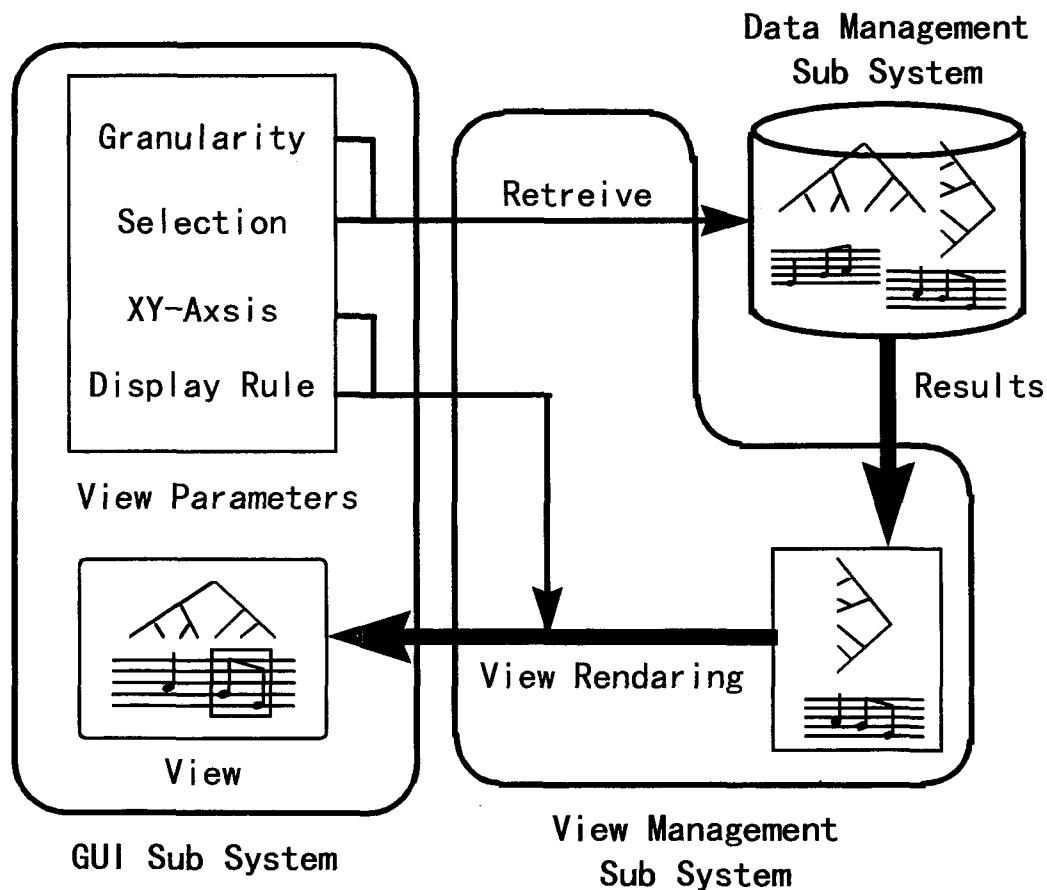


図 17 モジュール構成

Fig.17 System configuration.

図18はGUI部の画面例である。1は楽曲論理構造を表示するウィンドウである。ウィンドウ毎に各楽曲論理構造の木構造が表示されており、選択と粒度の指定ができる。ここで注目している属性を示すノードを選ぶことで、その属性を持つ楽曲要素がビュー表示に表示される。また、楽曲論理構造の根に近いノードを選ぶか、葉に近いノードを選ぶかで粒度が制御される。例えば、「役割」の楽曲論理構造で、ノード「旋律」を選ばずにその子ノードの主旋律・副旋律・対旋律を選べば、旋律の役割を詳細に表示することができる。

2はXY軸と表示ルールを指定するウィンドウである。1で選ばれた属性を、楽曲論理構造毎にまとめて表示している。その中から属性を選び表示属性を指定すると、属性と表示属性の間に表示ルールが設定される。プロトタイプでは表1の表示属性が指定できる。

3と4は生成されたビュー表示であり、3はフレーズの粒度で、4は音の粒度で表示したものである。

図19は図18右下のビューウィンドウを広げた時の画面であり、フレーズの役割によって枠線の太さと音符の色が異なって表示されている。

ビュー管理部は、GUI部で入力されたパラメータのうち選択と粒度から対応する楽曲要素を取出す質問を組立て、データ管理部に検索をかける。そして、XY軸・表示ルールと検索結果から描画命令を組立てGUI部に渡す。

データ管理部は提案モデルに基づいて楽譜情報と楽曲構造の管理を行うモジュールである。楽曲構造のファイルへの入出力とビュー管理部からの検索要求に答える簡易データベースとなっている。

表1 表示属性

Table1 Display attributes.

Display element	display attribute
note	color
character strings	color, style, size, position
region frame	color, style, width
region background	color

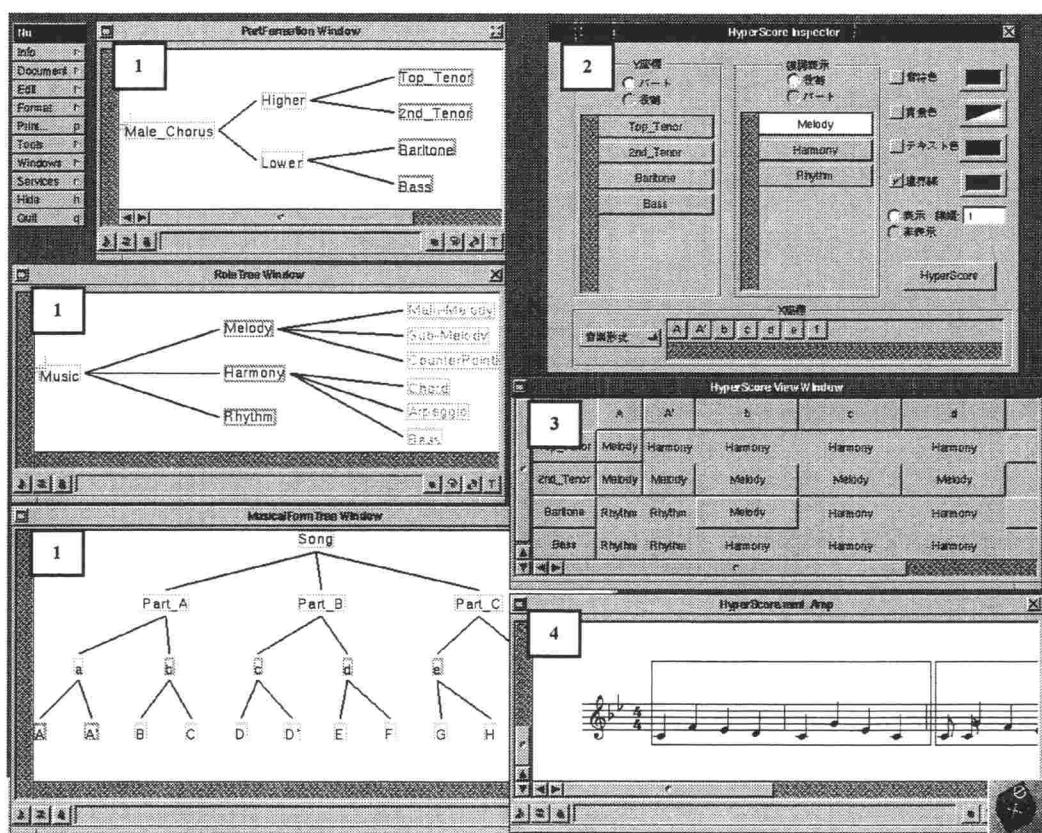


図 18 画面例(1)

Fig. 18 Example of a view(1).

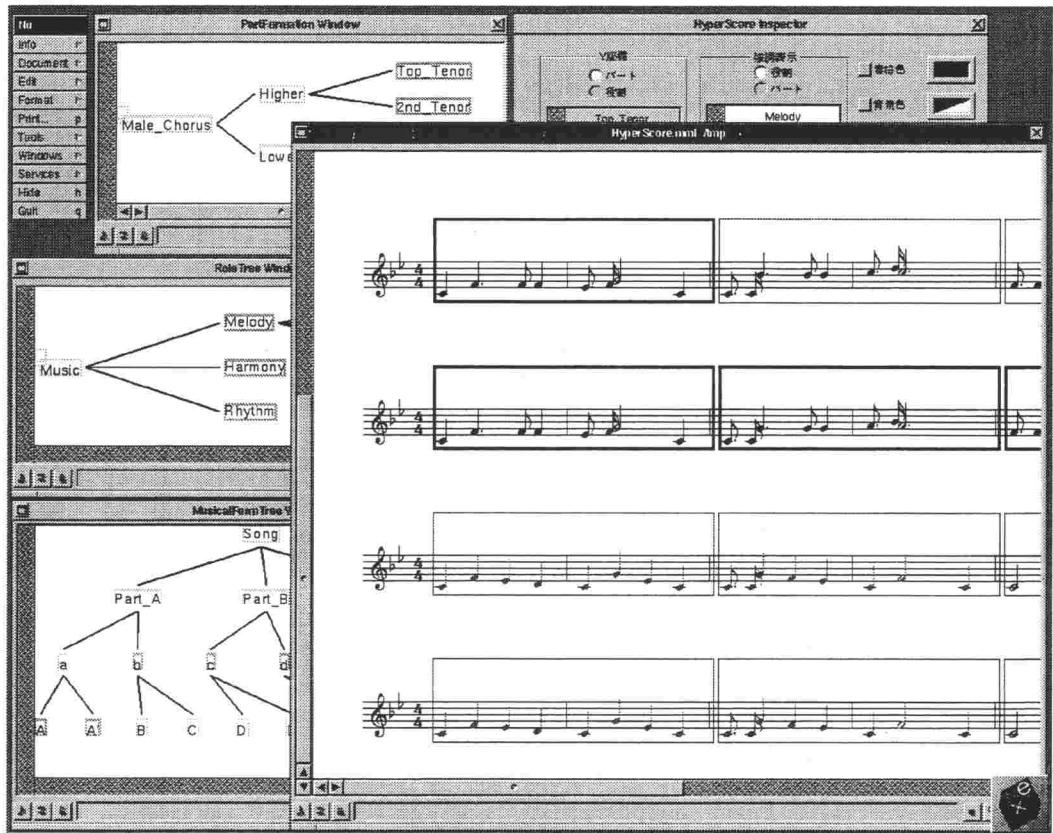


図 1 9 画面例(2)

Fig.19 Example of View(2).

5 従来研究

5.1 演奏制作支援

演奏制作支援に関連した従来研究として INTERPRET[3]と Muse[4]がある。INTERPRET は初心者に楽曲の演奏法を対話的に指導するチュータ・システムである。INTERPRET が指導する演奏法、つまり演奏プランは楽曲構造に基づいている。しかし、その楽曲構造は旋律のグルーピング構造 1[1]であり、グルーピング構造を決めるために一部和声構造も扱っているものの、声部は単声であり、本研究の多声部楽曲の複雑な楽曲構造と比べ単純な楽曲構造しか扱っていない。また楽曲構造や演奏プランを利用者に伝える機能では、表面的な楽譜情報は五線譜やピアノロール形式のグラフィカルな表示を補助的に使用しているが、楽曲構造の視覚化は特に行っていない。

Muse はオーケストラの指揮者と演奏者のための電子譜面台であり、譜面台部分に組込まれた液晶ディスプレイで楽譜を表示するようになっている。Muse は楽譜を総譜かパート譜に切り替えられるようになっているため、内部に声部構成の構造を保持していると思われるが、演奏制作を支援するために楽曲構造を積極的に扱うこととはしていない。

演奏制作支援に限らず、楽曲構造の表現に関連した研究をみると、[5]で音楽形式や和声構造などの個々の論理構造を扱った 17 件のシステムが紹介されている。しかし、これらのなかに複数の論理構造を扱えるものは無い。複数の論理構造を扱う機能があるものとしては SMDL[6]、MODE[7]などの音楽言語がある。SMDL は HyTime[8]に基づいた音楽用マークアップ言語の規格である。SMDL は規格上は複数の論理構造を扱えるように設計されているが、実際に複数の論理構造が扱える実装は報告されていない。MODE は Smalltalk の音楽用クラスライブラリとツール群であり、このクラスライブラリを使い複数の論理構造を扱うツールを作成することは可能であるが、現在のところ一つの論理構造を扱うツールしか報告されていない。

5.2 ハイパーテキストモデル

ここでは従来のハイパーテディアモデルを紹介し、楽曲構造モデルとの対比を行なう。ハイパーテキストモデルの基本は2つのノードを1対1リンクで直接結び付けるノードリンクモデルである。そして、本研究の楽曲構造モデルやここで取りあげるフレーム関係軸モデル[9,10]と Set-to-set モデル[11]は、ノードリンクモデルを拡張し、ノードに属性を付与しノード間の関係を属性値間の関係で表現するようにした構造化ハイパーテキストモデルを採用している。構造化ハイパーテキストモデルは属性値によってリンクを一段抽象化することで、1対1リンクモデルにおける複数のリンクを、1つの属性値間の関係で表現できることが特徴である。例えば、フレーム関係軸モデルではノード間の関係を関係軸と呼ばれる属性値の順序関係で表現している。また、set-to-set モデルではファセットと呼ばれる属性値の木構造で属性値間の関係を表現している。

5.2.1 Set-to-set

・モデル

Set-to-Set モデルは絵画や画家を様々な視点からブラウズできる電子美術辞典のために開発されたモデルである。

モデルは情報の実体を表わすドキュメントとある視点からの体系をあらわすファセットの2種類のオブジェクトから成り立つ。ドキュメントは属性値を持っており、それによってファセットと関係づけられる。ファセットは木構造に体系化した値(属性値)である視点からの体系を表わす。

また、ドキュメントは is-a や part-of リンクでサブドキュメントと関係づけることができる。

・トラバース操作

set-to-set は従来のハイパーテキストと同じリンクをたどるブラウズ操作に加えて、ドキュメント集合に対して繰り返しフィルタリングすることでブラウズを代用するトラバースという操作ができる。トラバース操作の基本は、ドキュメント集合から指定した属性値を持つものだけを取出すフィルタリング操作である。そして、フィルタリング条件を一部変更して次の条件にすることで、取出されるドキュメント集合を少しづつ変更していくのがトラバース操作である。そして、set-to-set には次の4種類のトラバース操作がある。

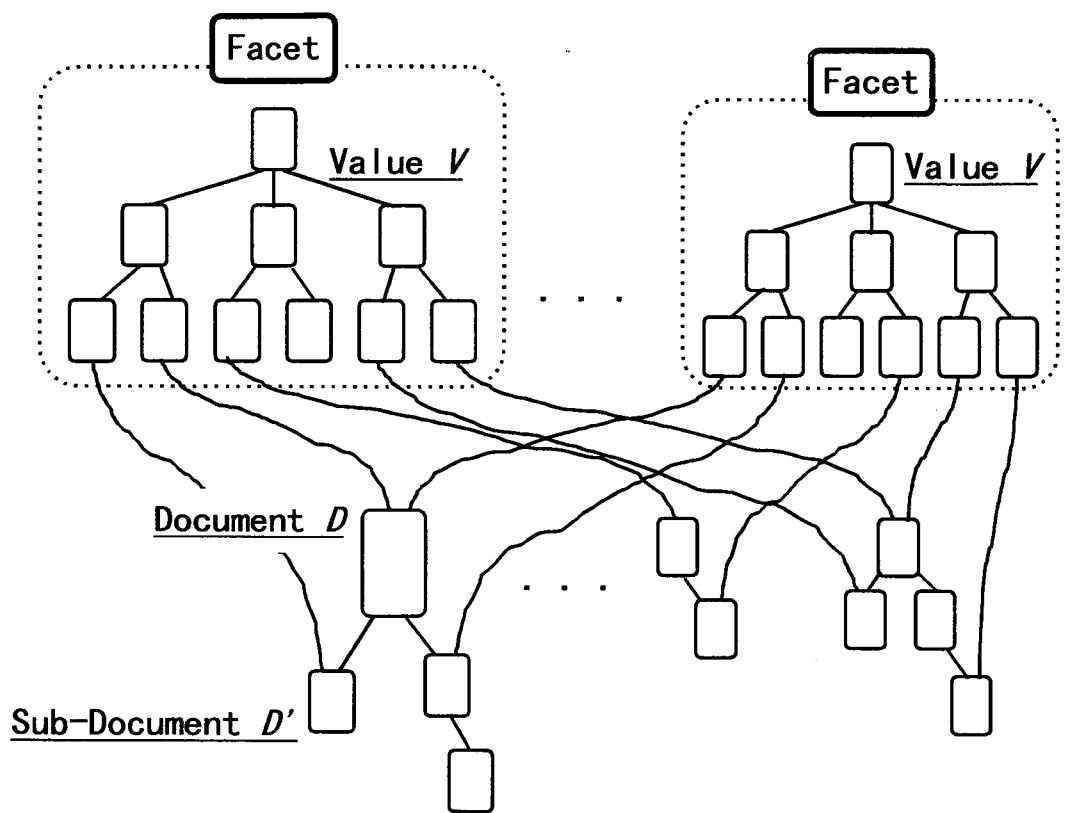


図 20 Set-to-set モデル

Fig. 20 Set-to-set model.

- **Hierarchy in/out**

Hierarchy in は条件中の属性値を、 値の階層構造の中でより下位レベルの詳細な値に変更してドキュメント集合を絞りこむ操作であり, **Hierarchy out** は上位レベルの値に変更してドキュメント集合を広げる操作である.

- **Zoom in/out**

zoom in は条件のファセットを増やしてドキュメント集合を絞りこむ操作であり, **zoom out** は条件となっているファセットを減らしてドキュメント集合を広げる操作である.

- **Constraint**

Constraint は **zoom in** と **zoom out** を同時に行なう操作である. 現在の条件からファセットの1つはずすと同時に別のファセットを条件に加える.

- **Aggregation**

Aggregation は複数のファセットを1つのファセットとして扱う操作である.

5.2.2 フレーム関係軸モデル

- **モデル**

フレーム関係軸モデルは情報の体系化作業を支援する CastingNet システムのために開発された.

フレーム関係軸モデルは情報の実体を表わすフレームとフレーム間の順序関係を表わす関係軸の2種類のオブジェクトで構成されている. フレームはオスロットと呼ぶ属性を持ち、 関係軸はフレーム間の関係を軸上の順序組として表わすため、 写像ルール集合を持っている. 写像ルールは、 ルールに適合したフレームを関係軸に写像する.

- **操作**

CastingNetにはハイパーチャート生成とブラウズの2種類の視覚化操作がある.

1) ハイパーチャート生成

ハイパーチャートは関係軸が持つ順序情報を利用して、 フレームをグラフやテーブル等のチャート上に配置するものであり、 ハイパーメディアのオーバービューを発展させたものといえる. 基本的にはチャートのX Y軸などの各軸に、 どの関係軸を対応させるかを指定することによりハイパーチャートは生成される. また、 スロット値を統計処理して視覚化するアプリケーションチャートと呼ばれる

チャートもある。例えば、クラスタリング処理したフレームをテンドログラムで表示することができる。

チャート生成の操作は利用者が注目する関係軸を複数選ぶと、チャート候補のメニューが表示され、その中から選べば良いようになっている。

2) ブラウズ

CastingNet でのブラウズ操作はフレームとハイパーチャートを交互参照する。そのため Overview と Focus の 2 つの操作がある。Overview はフレームからハイパーチャートへ移動する操作で、利用者がフレームのスロットを選択すると、そのスロット値を使っているハイパーチャートが表示され、そのフレームを表わす領域がハイライトされる。

逆に Focus はハイパーチャートからフレームへ移動する操作で、利用者がハイパーチャート上の領域を選択すると、その領域に対応するフレームが表示され、そのハイパーチャートで使っているスロットがハイライトされる。

各操作で複数の候補がある場合には候補がメニューで表示されるようになっている。

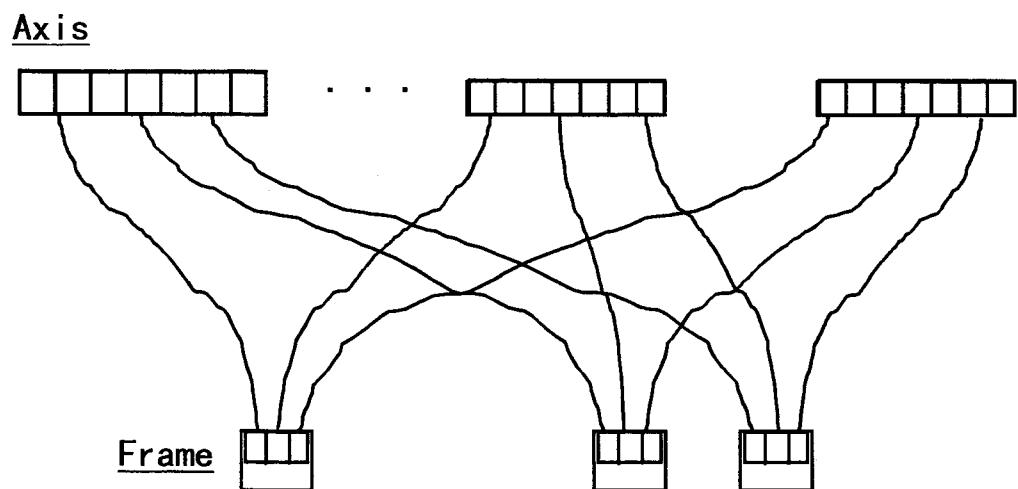


図 21 フレーム関係軸モデル

Fig.21 Frame-axis model.

5.2.3ハイパーメディアモデルと楽曲構造モデル

楽曲構造モデルが表現する3種類の属性付けの関係のうち、第一の楽曲論理構造による楽曲要素に対する属性付けは、楽曲要素をノード、属性を属性、楽曲論理構造をファセットに対応させることでSet-to-Setモデルにより表現することができる。

しかし、第二の要素論理構造による楽曲論理構造に対する属性付けを従来のモデルで表現することは難しい。従来のモデルでは属性を与える側と属性を与えられる側を別々の種類のノードで表現していた。しかし、楽曲論理構造中の属性を表わすノードは、楽曲要素に対しては属性を与えるが、要素論理構造からは逆に属性を与えられる。そのため、本研究では属性を与えることも、与えられることもできるノードとしてリージョンを導入し、リージョンで属性を表現することで第二の属性付けの関係の表現を可能にしている。

また、第三の楽曲要素間の包含関係は楽曲構造モデルでは楽曲要素間のパートオブリンクで表現しているが、Set-to-Setモデルでもis-aとpart-ofリンクがありノード間の包含関係を表現することができる。しかし、Set-to-Setモデルではドキュメントもサブドキュメントも、同じファセットから属性付けられており、属性付けの点からは区別はない。一方、楽曲構造では、楽曲要素は粒度により属性付けする楽曲論理構造は異なるという区別がある。この区別を表現するために、粒度と楽曲論理構造・要素論理構造毎にドメインが存在し、メンバオブのリンクでリージョンの所属関係を、コンストラクションのリンクでどの粒度の楽曲要素をどの楽曲論理構造が属性付けるかを表現している。

楽曲分析の分野では、名前は同じであるが属する楽曲論理構造が異なる属性が少なからず存在するが、ドメインがリージョンの楽曲論理構造への所属関係を表現していることは、この同じ名前の属性が指揮者から演奏者へ間違って伝わることを防ぐという役割も兼ねている。例えば、同じテノールという声部であるが、男声合唱でのテノールは高音域を担当するが、混声四部合唱では中音域を担当し、これらは演奏プランの選択に与える影響が異なる。しかし、それが同じ名前であるため間違って伝わるということがしばしば起こる。楽曲モデルではドメインがリージョンの楽曲論理構造への所属関係を明確に表現しているため、このような混乱が起こりにくくなっている。

5.3 視覚化

ここではハイパーテキストの視覚化について利用者の視点の指定とコンテキスト情報の2点から既存のモデルと HypreScore のモデルを比較する。

5.3.1 視点

Set-to-Set モデルには視覚化機能は無いが、ノード集合から特定の属性値を持つものだけを取り出すことができ、その取り出す条件によって利用者の視点を表現している。また、取り出す条件の一部を変更して次の条件として取り出されるノード集合を少しづつ変更していくトランザクション操作が提供されている。例えば、条件の属性値をファセット中の上位・下位レベルの値に変更することで視点の詳細度を変える **Hierarchy in/out** や、条件にファセットを追加・削除することで視点を狭めたり広げたりする **Zoom in/out** など、視点を意識した4種類のトランザクション操作がある。

フレーム関係軸モデルでは関係軸が持つ順序情報をを利用して、ノードをグラフやテーブル等のチャート上に配置するハイパーテキストチャートと呼ばれる視覚化機構が提案されている。利用者の視点は注目する関係軸を選択することで指定する。それによって、選択された関係軸がXY軸に対応したハイパーテキストチャートが生成される 2.

本研究の視覚化機構は、XY 軸・表示ルール・選択・粒度の4つのパラメータで視点を指定することでビュー表示を生成する。

三つのモデルを利用者の視点の指定という点で比較すると次の3項目の違いがある。

軸の数： ハイパーテキストチャートでは3次元表示のチャートを使っても3本の関係軸しか視覚化できない。しかし、Set-to-Set モデルでは任意の数のファセットを条件に使用できる。本研究の視覚化機構は XY 軸で視覚化できない楽曲論理構造は表示ルールにより表示属性に割り当てることで視覚化することができる、より多くの楽曲論理構造を視覚化することができる。

詳細度： ハイパーテキストチャートでは詳細度を変える機能は無く、ハイパーテキスト構造全体を俯瞰したり、一部分を拡大して注目することができない。しかし、Set-to-Set モデルでは **Hierarchy in/out** 操作により視点の詳細度が指定できる。本研究の視覚化機構も粒度パラメータにより詳細度を指定できる。

表示の煩雑さ： ハイパーテキストチャートは属性値が指定した関係軸に乗っている全てのノードが表示されてしまう。そのため画面が繁雑になりやすく、関心ある部分の弁別性が低い。しかし、Set-to-Set モデルでは取り出したいノードの属性値を指定するので、指定した属性値を持っていないノードは表示されない。本研究の視覚化機構も表示したい楽曲要素の属性を指定するので、指定した属性を持たない楽曲要

素は表示されず、画面が煩雑になりにくい。

5.3.2 コンテキスト

次に注目ノードが他のどのようなノードに囲まれて存在しているかというコンテキスト情報の提示という点について述べる。

Set-to-Set では、コンテキスト情報の重要性は指摘されているが、条件により取り出したノード集合に対して特に視覚化は行っていない。

ハイパーテチャートではノードの属性値やノード間の関係をチャート上での配置で表現しているため、コンテキスト情報が着目ノードとその周辺のノードとの位置関係や距離関係で表現され、視覚的に把握できるようになっている。

本研究の視覚化機構もハイパーテチャートと同じく、XY 軸の指定によりコンテキスト情報が着目ノードとその周辺のノードとの位置関係や距離関係で表現され、視覚的に把握できるようになっている。加えて、表示ルールの指定により、ノードの持つ属性は位置だけでなく表示属性で表現することができる。

6 評価

プロトタイプシステムの評価をするために、音楽教諭で指揮者でもある音楽専門家に実際に試用してもらった。その結果、

- 利用者が音楽学を勉強して楽曲構造に関する知識のある場合には、楽曲構造を理解し、演奏プランを立てる作業の良い支援になる
- 音楽学の知識のない人に対しても、楽曲分析や演奏プランの立て方などを教えるときの支援ツールとしても有効である。
- 視覚化だけでなくフレーズ等を実際に鳴らして確認できる機能が付け加わればさらに有効性であろう。

という評価であった。

また、指揮者だけでなく作曲家や演奏者に対してデモを行なったところ、好意的な感想が多く、特に視覚化機構の表示ルール指定、つまり、楽譜上で楽曲構造が強調表示される機能に対する評価が高かった。

そのため、今後次の項目の改良と評価を行なうことを考えている。

- 視覚化機構に加えて、属性を音色や音量に割り当てて聴覚化する機構を加え、楽曲構造を提示する機能を強化する。
- 現在のプロトタイプシステムの視覚化機構は、2.1節で述べた音列の演奏プランを決める4種類の構造のうち1, 3, 4番目にしか対応していないため、2番目の多段階分析による要素音楽構造から音楽構造への意味付けにも対応する。

また、楽曲構造モデルでは楽曲要素と楽曲論理構造のトポロジとして木構造しか扱っていない。しかし、音列の最後の音と次の音列の最初の音の音程が同じ時に、音楽の流れの継続感を出すために両音を1つにしてしまう、省略やオーバラップと呼ばれる技法がある。現状の楽曲モデルではオーバラップがある場合は表現できない。しかし、オーバラップを適用する前の音が分離している楽譜を、オーバラップを適用した後の楽譜の背後に関連づけておくことで対処できることが分かっている[1]。そのため、楽曲構造モデルをオーバラップに対応できるように拡張する予定である。

視覚化機構では対象と想定している利用者である音楽家が計算機利用に習熟していないことが考えられるので、楽譜のメタファからかけ離れたものは用いないで、利用者に分かりやすい（親しみやすい）楽譜システムをもとにして、その拡張になるように設計した。しかし、分かりやすい（親しまれやすい）楽譜システムの拡張という範囲にこだわらず、五線譜等の伝統的な音楽表現から離れて、例えばGraphical fisheye view[13]等のより高度な視覚化手法の適用も行なっていく予定である。

7 結論

本研究では多声部楽曲の演奏制作支援を目的として、楽曲構造を視覚化する HypreScore について述べた。まず、楽曲の演奏プランを決定する楽曲構造による音楽的意味づけの分析を行ない、従来のハイパーテキストモデルを拡張した楽曲構造モデルを提案した。次に、指揮者や演奏者の楽曲構造に対する見方の調査にもとづいて楽曲構造の視覚化機構の提案し、プロトタイプシステムの実装について述べた。

楽曲構造モデルは、フレーズ・モチーフ・音という楽曲要素の階層構造を表現する木構造と、音楽形式や声部構成等の楽曲論理構造を表現する木構造、両者の関係を表す属性付けリンクから構成され、楽曲構造の演奏プラン作成に関係する部分を簡潔に表現する。視覚化機構は X Y 軸・表示ルール・選択・粒度の 4 種類のパラメータにより、利用者による楽曲構造に対する視点の違いに合わせたビュー表示を生成し、ビューではフレーズ等の持つ音楽学的属性をビュー表示上の位置と表示属性によって表現することで、複雑な楽曲構造を分かりやすく表示する。

本研究では計算機に習熟していない音楽家による使用を考慮し、楽譜システムをもとに楽曲構造を視覚化する拡張を行ったが、Graphical fisheye view[13]等のより高度な視覚化手法を適用することが今後の課題である。

本研究で扱った多声部楽曲の論理構造とマルチメディアコンテンツには、時間軸があることと、複数の声部や複数のメディアが同時に存在するという類似性がある。計算機をマルチメディアを扱う道具にするためには、本研究や文章におけるアウトラインエディタのように、マルチメディアも論理構造レベルで扱えることが重要である。そのため、今後、本研究で得られた多声部楽曲の論理構造レベルの扱いに関する成果を、マルチメディアの論理構造レベルでの扱いに応用する研究を行なっていく予定である。

謝辞

本研究をまとめるにあたり、本研究の機会を与えていただき、終始懇切な御指導をいただいた宮原秀夫教授に心から感謝致します。

研究の方針や内容について直接御指導いただいた下條真司助教授に感謝致します。

学部、大学院を通じて御指導、御教授いただいた大阪大学基礎工学部情報工学科の首藤勝教授、菊野亨教授、萩原兼一教授、嵩忠雄教授(現在、奈良先端科学技術大学院大学教授)、谷口健一教授、都倉信樹教授、鳥居宏次教授(現在、奈良先端科学技術大学院大学教授)、柏原敏伸教授に心から感謝いたします。

研究の細部にわたり御討論、御助言いただいた藤川和利助手(現在、奈良先端科学技術大学院大学助手)に感謝致します。

日頃から温かい御支援、御助言をいただいた村田正幸助教授、山口英助教授(現在、奈良先端科学技術大学院大学助教授)、松浦敏雄助教授(現在、大阪市立大学学術情報総合センタ教授)、西尾章治郎教授(現在、大阪大学大学院工学研究科教授)に感謝致します。

著者の在学中、御討論いただいた大阪大学基礎工学部情報工学科情報ネットワーク学講座の方々に心から感謝致します。特に、本研究に御協力いただいた上原征樹氏(現在、NTTデータ)、島村栄氏(現在、NEC)に深く感謝致します。

音楽情報処理ならびに楽曲分析理論の調査をともにさせていただいたイメージ情報科学研究所の片寄晴弘博士(現在、和歌山大学システム工学部助教授)、兵庫教育大学の竹内好弘氏(現在、京都府立亀岡高校教諭)および音楽情報科学勉強会の皆様に感謝致します。

参考文献

- [1] F. Lerdahl, R. Jackendoff, "A generative theory of tonal music," MIT Press, Massachusetts, 1983.
- [2] G. Diener, "Ttrees: a tool for the compositional environment," The Well-Tempered Object, S. Pope, pp.157-168, MIT Press, Massachusetts, 1991.
- [3] M. Backer, "Design of an intelligent tutoring system for musical structure and interpretation," Understanding Music with AI, M. Balaban, et al., pp.466-489, MIT Press, Massachusetts, 1992.
- [4] C. Graeffe, D. Wahila, J. Maguire, O. Dasna, "Muse digital music stand for symphony musicians," acm interactions, vol. III.3, pp.27-34, May 1996.
- [5] W. Hewlett, "Musical information," Computing in Musicology, vol.8, W. Hewlett, E. Selfridge-Field, pp.61-134, CCARH, California, 1992.
- [6] ISO, "Information Technology - Standard Music Description Language (SMDL)," ISO DIS 10743, 1995.
- [7] S. Pope, "Introduction to MODE: The Musical Object Development Environment," The Well-Tempered Object, S. Pope, pp.157-168, MIT Press, Massachusetts, 1991.
- [8] ISO, "Information Technology Hypermedia/ Time-Based Structuring Language (HyTime) First Edition," ISO 10744, 1994.
- [9] Y. Masuda, Y. Ishitobi, M. Ueda, "Frame-Axis Model for Automatic Information Organizing and Spatial Navigation," Proc. ECHT'94, Edinburgh, England, pp.146-157, September 1994.
- [10] 植田学, 増田佳弘, 石飛康浩, "CastingNet: 情報の組織化と可視化ブラウジングのためのハイパーテディア," コンピュータソフトウェア, vol.12, no.4, pp.56-70, July 1995.
- [11] Y. Hara, Y. Kasahara, "A Set-to-Set Linking Strategy for Hypertext Systems," Proc. of COIS90, Massachusetts, America, pp.131-135, April 1990.
- [12] J. Rumbaugh, "Object-Oriented Modeling and Design," Prentice Hall Inc., New Jersey, 1991.
- [13] M. Sarkar, M. Brown, "Graphical Fisheye Views of Graphs", Proc. of CHI'92, PP.8-91, 1992.