平成 25 年度 修士論文

讃美歌データベースを用いたベイジアンネットワークによる自動和声付けの研究

指導教員 夜久竹夫教授

日本大学大学院総合基礎科学研究科地球情報数理科学専攻

鈴木峻平

2014 年 2 月 提出

概　　要

音楽の自動生成の分野において、与えられた主旋律にコード進行などを付与する和声付けは、主要な課題のひとつである。更に、和声付けの自動化は、作曲支援だけでなく、音楽理論をモデル化するという重要な課題でもある。その中で、四声体和声付けは様々な研究がされてきた。これらの研究の古くはエキスパートシステムなどのルールベースのアプローチがとられてきた。しかし、後述する音楽的同時性と音楽的連続性を両立する網羅的なルールセットを設計するのは難しく、近年では、隠れマルコフモデルやベイジアンネットワークなどの機械学習を用いることが一般的である。

本研究では、確率推論に基づいた機械学習技術の一種であるベイジアンネットワークを用いた四声体和声の自動生成について検討する。ベイジアンネットワークを用いる際に最も重要な課題は、モデル (ベイジアンネットワークを構成するノードとアークの集合をさす) をどのように設計するかである。この設計を行う上で、主に 2 点の留意点がある。ひとつは音楽的同時性と音楽的連続性を両立すること、もうひとつは和音の適切な粒度で表現することである。

「音楽的同時性と音楽的連続性の両立」とは、楽譜における縦と横の関係の依存関係をともに考慮することをさす。ここで、音楽的同時性とは同時になるいくつかの音において、声部間の協和な関係を保つこと、音楽的連続性とは声部内で滑らか旋律を作りだすことをいう。この 2 つの片方が欠けると音楽的に不適切な和声になる。

「和音の適切な粒度による表現」とは、学習データが有限であることを前提に、

有限のデータで学習可能な程度の粒度で和音を表すことである。和音を表す上で、最もよく用いられるのがコードネームである。コードネームは表現能力の高い表記方法として広く知られているが、ヴォイシングやテンションノート、ベース音を区別して記述すると数万種類になってしまい、有限のデータから出現確率を学習することは困難である。これらの区別を行わなければコードネームの種類が大幅に減らすことできるが、これらを区別せずに音楽的に適切な和声を生成することは困難である。

本研究では、ベイジアンネットワークを次のように設計することで、これらの問題を解決する。前者の問題に対しては、音楽的同時性と音楽的連続性をそれぞれ適切なノード間のアークとして表現し、全体の確率が最大となる和声を推定する。後者の問題に対しては、コードシンボルを含まないモデルを設計する。コードシンボルを考慮せずに、4 声部間のみの関係を考慮することで、コードシンボルの曖昧性の問題を解決し、適切な和音を推論することができる。

このモデルの有効性を確かめるために、比較対象としてコードシンボルを含めたモデルを作成し、コードシンボルを含めないモデルとの結果の比較を行った。評価には、音楽的同時性と音楽的連続性の観点から客観評価と主観評価を用いて行った。結果として、コードシンボルを含めないモデルの結果で不協和音がわずかに多くなってしまったが、コードシンボルを含まないモデルの結果の方が、豊富な和音の種類の出現や、各声部の旋律やコード進行が滑らかになるなど、音楽的連続性の観点において多くの点で優れていた。

目　　次

図 目 次

第1章　序　　論

和声付けは、音楽の自動生成における主要なテーマのひとつである。和声付けは通常、与えられた主旋律に対してコード進行またはヴォイシングを含めた伴奏部の音符列を付与する問題と定義される。この自動化は、作曲支援に有効なだけでなく、数理的モデル化により音楽理論がより精緻化される可能性があるという点でも重要な研究テーマのひとつである。

一般的に、このタスクは2つのタイプに分けられる。ひとつは、与えられたメロディに対して C-F-G7-C のようなコードの連なりを出力する手法である [1, 2, 3]。もうひとつは、メロディに対して具体的な音を他の声部に付与する手法である。後者の手法の典型的なものに、メロディであるソプラノに対してアルト、テノール、バスを付与する四声体和声付けがある。四声体和声付けは、バッハなど西洋の有名な作曲家の曲に基づき作成した規則であり、音楽理論として伝統的な手法の一つである。それ故に、他の音楽生成タスクと同様に多くの研究者が四声体和声付けを自動化する研究に取り組んでいる [4, 6, 5, 7, 8, 9, 10, 11, 12]。

四声体和声付けを行う際に、重要なこととして同時性と連続性の考慮があげられる。同時性とは、声部間同士の関係のことであり、連続性とは、声部内での音の繋がりである。同時性を考慮することによって、和音の関係が考慮され不協和音などを減らすことができる。連続性を考慮すると、前後の音の繋がりが考慮され、音の繋がりが滑らかになる。既存研究では、様々な手法によって同時性と連続性を考慮している。最も一般的な方法としてエキスパートシステムなどを用いたルールベースシステムがあげられる。Ebcioğlu はプログラミング言語 BSL を用

いて和声の知識を実現した [4]。このシステムのルールは多様であり、矛盾を起こさずに統一するのは困難である。その為、和声付けを行うルールベースシステムの開発は困難であると考えられる。特に和声付けにおいて、同時性と連続性を考慮することは重要である。同時性と連続性の考慮によってそれぞれが異なる音を選んだ場合、このような矛盾を解決するメタルールが必要である。しかし、このようなメタルールを作ることは困難であり、時にそれは不可能でもある。

そこで、近年では機械学習を用いた研究が多く見られる。バッハのコラールのような既存の四声体和声から機械学習を行うことによって、和声付けの規則を矛盾なく学習することができる。その際に、これらの確率モデルを用いたほとんどの既存研究では、コードシンボルを表すノードや状態をモデルに導入している。しかしコードシンボルには曖昧性があり、モデルに対して適切にコードシンボルを導入することは困難である。例えば、コード C-E-G[1]、E-G-C、G-C-E は同じコードシンボル “C” で表される。“G-A-C-E” もまたテンションノートは時に省略されることもあるので “C” と表される場合がある。これらのコードシンボルをモデルが学習する場合、ヴォイシングやテンションノートを区別せずにコードを学習する。ヴォイシングは前後の和音の繋がりを滑らかにするために重要なため、これを考慮しなければ不適切なコードヴォイシングが行われる可能性がある。一方、“C6/G” の様にヴォイシングを区別した表記方法は、コードシンボルの数が膨大になり、データスパーネスの原因となってしまう。それによって、適切な和声が生成されない可能性がある。そこで本論文では、コードシンボルを含めないモデルを用いて和声付けを行う。このモデル (ノンコードモデル) は 4 声部間の依存関係のみを考慮する。実験として、コードシンボルを含めたモデル (コードモデル) も用いて本論文では以下の仮説を確かめる。

1. ヴォイシングやテンションノートを区別しないコードモデルでは、異なるヴォイシングが好ましい場合でさえも、同じコードに対して同じヴォイシングが

[1] コード “C-E-G” は C, E, G からなるコードを表す。

生成されてしまう傾向にある。結果として、生成された和声は滑らかさに欠ける。

2. ヴォイシングやテンションノートを区別したコードモデルでは、限られた学習データではデータスパース問題が発生し、適切な和声は生成されない。

3. ノンコードモデルでは、コードモデルの結果よりも滑らかな和声が生成される。

多くのモデルがベイジアンネットワークで実現されている為、本研究で扱うモデルもベイジアンネットワークで実現する。

第2章　関連研究について

この章では、本研究に関連する研究について述べる。

2.1　ルールベースシステム

いくつかの研究では、直接音の配置のルールではなく、充足制約問題や遺伝的アルゴリズムとして和声付けの実現を試みている。これらの研究は、いくつかの和声付けの基本的なルールに従って、制約や fitness 関数を作成した [5, 6]。Phon-Amnuaisuk らは、使用者がシステムの和声付けのふるまいを操作できるルールを含めた音楽知識表現によってプラットフォームを開発した [7]。

2.2　機械学習を用いた研究

近年では、機械学習を用いた四声体和声付けの研究は増えている [8, 9, 10]。機械学習では、同時性と連続性は既存の音楽データから学習するため、人間がルールを導入する必要はない。例えば、Hild らは J. S. Bach-style choral harmonization system using several neural networks を開発した [8]。Racyźnski らは、隠れマルコフモデルによる和声付けを提案し [13]、それに対して Allan と Williams は異なる隠れマルコフモデルを用いた [9]。Yi と Goldsmith はマルコフ決定過程に基づいて四声体和声付けの手法を提案した [10]。

Yi と Goldsmith のモデルは全ての声部の連続する 2 つの音からなる組み合わせを状態と定義した [10]。そのような状態の前後関係をマルコフ決定過程として扱っ

ている。このモデルは、同時性と連続性の両方を考慮した和声付けを行うことができる。しかし、多くの状態を持っていることによって、膨大な学習データが必要であるが、彼らは実際にコーパスを用いた学習の結果の報告はしていない。

Hild らは 3 つのニューラルネットワークを統合することによって四声体和声付けを試みた [8]。ニューラルネットワークの 1 つは、harmonic skeletons と呼ばれメロディから機能和声を生成する。2 つめは、harmonic skeletons から具体的な音の配置を行い、3 つめで装飾音として、8 分音符を挿入する。彼らは 3 つのニューラルネットワークに長調と短調のバッハのコラールを 20 曲 1 セットとしてそれぞれ学習させている。

Allan らは隠れマルコフモデルを用いた [9]。このモデルの隠れ状態はコードであり、観察可能なメロディラインから推論される。状態は 0:4:9:16 / T (T はトニック) のように、ソプラノの音からアルト、テノール、バスの音の音高の差のリストとしてコード化されている。このモデルは同時性と連続性を考慮しているが、このモデルも多くの状態を保持しているため、多くの学習データを必要とする。実際に、彼らのモデルはそれぞれの同じコードを唯一のボイシングとして識別している。

Racyzński らもまた、和声付けの為の隠れマルコフモデルに基づいたシステムを提案した。彼らもまたコードシンボル C_t を隠れ状態とし、メロディシンボルを M_t として観測状態を表した。彼らは同時性を $P(M_t \mid C_t)$ で定義し、連続性を $P(C_t \mid C_{t-1})$ として定義した。これらをマルコフ連鎖を用いて推論する。

Buys らは、重み付き有限状態トランデューサー (WFST) を四声体和声付けに取り入れた [11]。彼らの WFST は、コードを推定するモデル、バスの音を推定するモデル、内声であるアルト、テノールを推定するモデルの 3 つのモデルから成っている。アルト、テノール、バスの音は上記のモデルとは異なり、別々にコード化されている。

第3章　提案手法

この章では、ベイジアンネットワークを用いた四声体和声付けの手法について述べる。

3.1　問題設定と音楽知識の表現について

本研究の目的はソプラノ声部に従ってアルト、テノール、バスの旋律を生成する方法を探求することである。ここでは、ソプラノメロディを $\{S_1, \cdots, S_N\}$ と表す。アルト、テノール、バスの旋律はそれぞれ $\{A_1, \cdots, A_N\}$, $\{T_1, \cdots, T_N\}$, $\{B_1, \cdots, B_N\}$ と表す。簡単のため、音符の総数とそれぞれの音符の発音時刻、消音時刻の時間は全て同じであるとする。音符が持つ情報は、音高と音の長さである。そのうちの音の長さはすべて同じなため、それぞれの音の音高のみを決定する。一般的な四声体和声の音域に沿って、S_i、A_i、T_i、 B_i が取りうる音域はそれぞれ [60, 81]、[55, 76]、 [48, 69]、 [41, 62] とする。コードモデルでは、学習の際にコードシンボルが必要になるので、それぞれのソプラノに対して既にコードシンボル C_i が表記されているコーパスを用いることとする。本研究では、コードシンボルの詳細の度合いによる影響を調査するために、同じ曲に対して 2 つの異なるコード表記方法を用いる。ひとつは $\{\mathrm{C}, \mathrm{C}\sharp, \cdots, \mathrm{B}\} \times \{\mathrm{maj}, \mathrm{min}\}$ [1]からなる 24 種類のコードラベルである。もうひとつは、ヴォイシングやテンションノートを含む可能な限りより多くの詳細を表記したものである。既存の四声体和声において、全てが声部間のリズムが同じとは限らない。そこで、本論文ではそのような曲において、声

[1]異名同音は区別しない。例えば、音楽的文脈においてコード $\mathrm{E}\flat\mathrm{m}$ は $\mathrm{D}\sharp\mathrm{m}$ として表す。

部ごとで同時になっている音符が他声部よりも長い音符の場合、その音符を分割する (図 3.1)。

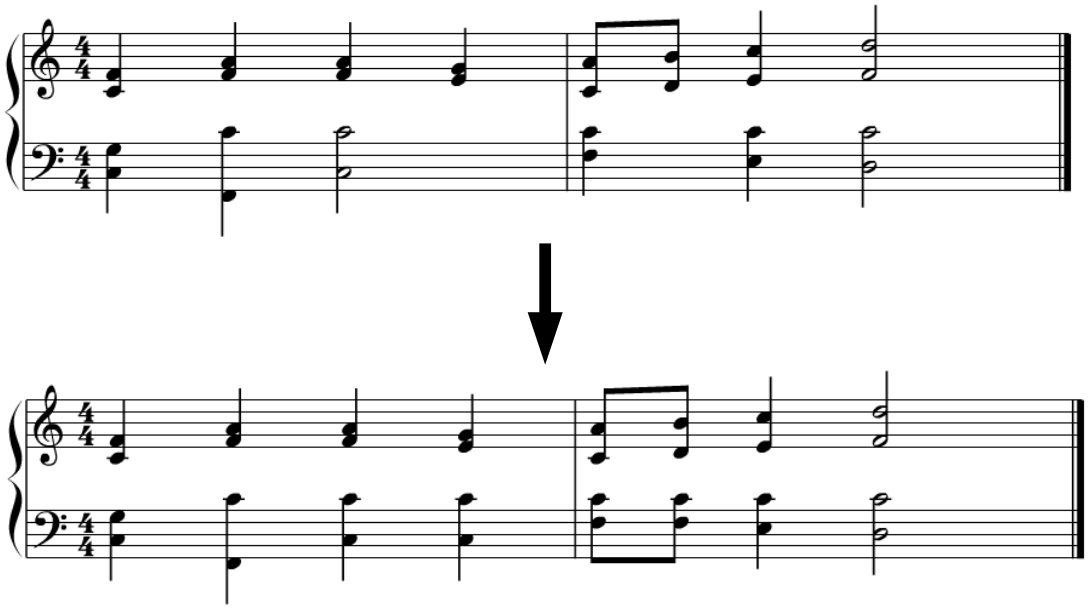


図 3.1: 音符の分割

次に和声付けでの音楽知識の表現方法について述べる。1 章で述べたように、和声には縦の関係である同時性と横の関係である連続性の両方がある。故に、これら 2 つの関係を可能な限り同時に表現することは重要である。本論文では、有向グラフに基づいた確率モデルの一種であるベイジアンネットワークを用いる。ノードはグラフの確率変数を表し、対してアークは確率変数間の依存関係を表す。それぞれの声部内のノードをつなぐアークは連続性を表し、声部間のノードをつなぐアークは同時性を表す。

3.2 ベイジアンネットワークの設計

ここでは問題を定式化し、以下の式を最大化する A_i、T_i、 B_i を導く。

$$P(A_1, \cdots, A_N, T_1, \cdots, T_N, B_1, \cdots, B_N | S_1, \cdots, S_N).$$

これは以下の式の最大化と等価である。

$$P(S_1, \cdots, S_N, A_1, \cdots, A_N, T_1, \cdots, T_N, B_1, \cdots, B_N).$$

この同時確率を計算する際に、変数が多すぎると計算が困難である。そこで、以下の制約を導入することで近似した解を求めることとする。

1. マルコフ性として各声部の i 番目の音は $i-1$ 番目の音のみに依存するとみなす。

2. 条件付き変数の総数が 3 を超える場合、条件付き確率 (CPT) は 500 万以上のセルを有するほど大きくなり過ぎてしまうので、実際の計算コストを考え条件付き確率の為の条件付き変数の総数を 3 つまでとする。

一つ目の制約により、上記した式は以下の様に展開される。

$$\begin{aligned} &P(S_1, \cdots, S_N, A_1, \cdots, A_N, T_1, \cdots, T_N, B_1, \cdots, B_N) \\ &\quad = P(S_1, A_1, T_1, B_1) \prod_{i=2}^{N} P(S_i, A_i, T_i, B_i | S_{i-1}, A_{i-1}, T_{i-1}, B_{i-1}). \end{aligned}$$

ここで、コードや音符がないことを表す特別な変数として “0” を導入する。S_0、A_0、 T_0、 B_0 に 0 がセットされると、$P(S_1, A_1, T_1, B_1 | S_0, A_0, T_0, B_0)$ における S_1、A_1、 T_1、 B_1 の条件付き確率は、前の音符はない状態、つまり曲の始まりとして計算される。変数 “0” も用いた式は以下のように展開される。

$$P(S_1, \cdots, S_N, A_1, \cdots, A_N, T_1, \cdots, T_N, B_1, \cdots, B_N)$$
$$= \prod_{i=0}^{N+1} P(S_i, A_i, T_i, B_i | S_{i-1}, A_{i-1}, T_{i-1}, B_{i-1}),$$

同様にして曲の終わりに関しても、S_{N+1}、 A_{N+1}、 T_{N+1}、 B_{N+1} に 0 がセットされると、曲の最後として S_N、 A_N、 T_N、 B_N は計算される。

同時にこれらの変数を最適化することは困難なので、本研究では時間軸にそって順番に最適化を行う。A_i、 T_i、 B_i が推論されるとき,$A_1, \cdots, A_{i-1}$, $T_1, \cdots, T_{i-1}$, $B_1, \cdots, B_{i-1}$ は既に決定されているとする。このように一定の確率を省略することで、A_i, T_i, B_i を最大化する以下の式を得る。

$$P(S_i, A_i, T_i, B_i | S_{i-1}, A_{i-1}, T_{i-1}, B_{i-1}) P(S_{i+1}, A_{i+1}, T_{i+1}, B_{i+1} | S_i, A_i, T_i, B_i).$$

縦と横の関係のみ考慮されるので (すなわち例えば B_i と S_{i-1} の関係は考慮しない)

$$P(S_i, A_i, T_i, B_i | S_{i-1}, A_{i-1}, T_{i-1}, B_{i-1})$$
$$= P(A_i | T_i, B_i, S_i, A_{i-1})\ P(T_i | B_i, S_i, T_{i-1})\ P(B_i | S_i, B_{i-1})\ P(S_i | S_{i-1}).$$

2 つ目の制約として、A_i と T_i は独立とみなす。それにより、

$$P(S_i, A_i, T_i, B_i | S_{i-1}, A_{i-1}, T_{i-1}, B_{i-1})$$
$$= P(A_i | B_i, S_i, A_{i-1}) P(T_i | B_i, S_i, T_{i-1}) P(B_i | S_i, B_{i-1}) P(S_i | S_{i-1}).$$

同様にして、$P(S_{i+1}, A_{i+1}, T_{i+1}, B_{i+1} | S_i, A_i, T_i, B_i)$ は以下のように展開される。

$$P(S_{i+1}, A_{i+1}, T_{i+1}, B_{i+1} | S_i, A_i, T_i, B_i)$$
$$= P(A_{i+1} | B_{i+1}, S_{i+1}, A_i)\ P(T_{i+1} | B_{i+1}, S_{i+1}, A_i)\ P(B_{i+1} | S_{i+1}, B_i)\ P(S_{i+1} | S_i).$$

従って最大化を得る為の式は以下のようになる。

$$P(A_i|B_i,S_i,A_{i-1})P(T_i|B_i,S_i,T_{i-1})P(B_i|S_i,B_{i-1})P(S_i|S_{i-1})$$
$$\times P(A_{i+1}|B_{i+1},S_{i+1},A_i)\ P(T_{i+1}|B_{i+1},S_{i+1},A_i)\ P(B_{i+1}|S_{i+1},B_i)\ P(S_{i+1}|S_i).$$

図 3.2 はこの式に基づいたベイジアンネットワークモデルである。

次にコードモデルについて述べる。本研究では 2 つのコードモデルを設計したが、2 つの違いはコードノードの要素数のみで、構造は同じものである。それ故に、2 つのコードモデルについては同時に述べる。コードモデルではコードノードも含めるので、最大化する式は以下のようになる。

$$P(S_i,A_i,T_i,B_i,C_i|S_{i-1},A_{i-1},T_{i-1},B_{i-1},C_{i-1})$$
$$\times P(S_{i+1},A_{i+1},T_{i+1},B_{i+1},C_{i+1}|S_i,A_i,T_i,B_i,C_i)$$

コードモデルはノンコードモデルよりも多くのノードを持つことになるので、声部間の関係も同時に考慮することはより困難である。従って、それぞれの声部はコードノードにのみ依存しているとみなし、

$$\begin{aligned}
&P(S_i,A_i,T_i,B_i,C_i|S_{i-1},A_{i-1},T_{i-1},B_{i-1},C_{i-1})\\
&\qquad\times P(S_{i+1},A_{i+1},T_{i+1},B_{i+1},C_{i+1}|S_i,A_i,T_i,B_i,C_i)\\
&=P(C_i|C_{i-1})\ P(S_i|C_i,S_{i-1})\ P(A_i|C_i,A_{i-1})\ P(T_i|C_i,T_{i-1})\ P(B_i|C_i,B_{i-1})\\
&\qquad\times P(C_{i+1}|C_i)\ P(S_{i+1}|C_{i+1},S_i)\ P(A_{i+1}|C_{i+1},A_i)\ P(T_{i+1}|C_{i+1},T_i)\\
&\qquad\times P(B_{i+1}|C_{i+1},B_i)
\end{aligned}$$

とする。図 3.3 はこの式に基づいたベイジアンネットワークモデルである。

3.3 学習過程

条件付き確率表を作成する為に、本研究では以下の条件に基づきコーパスを作成した。

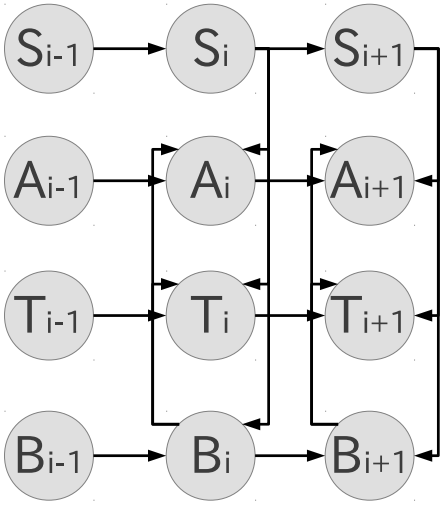


図 3.2: ノンコードモデル

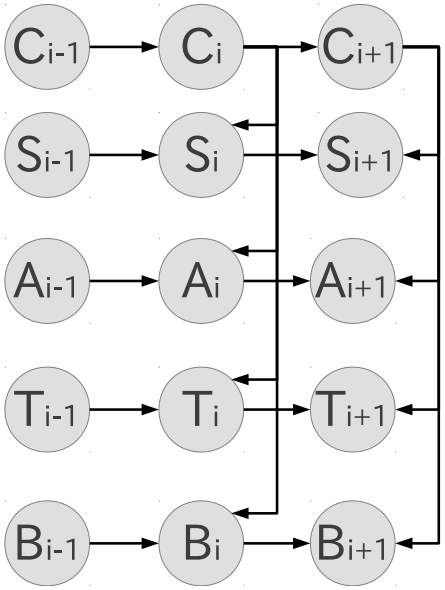


図 3.3: コードモデル

- 他の声部よりも長い音は分割する。
- それぞれの音にコードを付与する。

本研究では、ノンコードモデルは $(S_{i-1},\ A_{i-1},\ T_{i-1},\ B_{i-1},\ S_i,\ A_i,\ T_i,\ B_i,\ S_{i+1},\ A_{i+1},\ T_{i+1},\ B_{i+1})$ の組み合わせの集合、もしくはコードモデルは $(S_{i-1},\ A_{i-1},\ T_{i-1},\ B_{i-1},\ C_{i-1},\ S_i,\ A_i,\ T_i,\ B_i,\ C_i,\ S_{i+1},\ A_{i+1},\ T_{i+1},\ B_{i+1},\ C_{i+1}$ の組み合わせの集合をコーパスから獲得し、これらのデータから上記した式の全ての条件付き確率を条件付き確率表として学習した。$i = 1$ のときは、$S_{i-1},\ A_{i-1},\ T_{i-1},\ B_{i-1},\ C_{i-1}$ には “0” がセットされる。また、$i = N$ の場合も同様に $S_{i+1},\ A_{i+1},\ T_{i+1},\ B_{i+1},\ C_{i+1}$) にも “0” がセットされる。条件付き確率表の学習は単純にデータの統計をとり、その結果から直接集計をした。

3.4 推論過程

上記したように、$\{A_i\}$,$\{T_i\}$, $\{B_i\}$ は $i = 1$ から順番に推論される。まず最初に、(A_1, T_1, B_1) が推論される。次に、(A_2, T_2, B_2) は既に決定している (A_1, T_1, B_1) を踏まえた上で推論される。推論は 3.2 章で示した式を最大化する。(A_i, T_i, B_i) が推論される際に、$(A_{i+1}, T_{i+1}, B_{i+1})$ と滑らかな繋がりになるような (A_i, T_i, B_i) も同時に推論される。$(A_{i+1}, T_{i+1}, B_{i+1})$ は次の推論において再度推論される。$i = N$ となるような最後の音を推論する際に、A_{i+1}, T_{i+1}, B_{i+1}, C_{i+1} には “0” がセットされる。

第4章　ノンコードモデルとコードモデルの比較（実験1）

上記したように、本研究ではベイジアンネットワークを用いた四声体和声の生成を行う実験を行う。この実験の目的は3つの和声付けモデルの比較を行うことであり、ノンコードモデルと2つのコードラベルの詳細の程度が異なるコードモデルを用いる。第1章で示したように、本研究では以下の仮定を立てた。

1. ヴォイシングやテンションノートを区別しないコードモデルでは、異なるヴォイシングが好ましい場合でさえも、同じコードに対して同じヴォイシングが生成されてしまう傾向にある。結果として、生成された和声は滑らかさに欠ける。

2. ヴォイシングやテンションノートを区別したコードモデルでは、限られた学習データではデータスパース問題が発生し、適切な和声は生成されない。

3. ノンコードモデルでは、コードモデルの結果よりも滑らかな和声が生成される。

本論文では、客観評価と主観評価を用いて3つの仮説の妥当性を評価する。

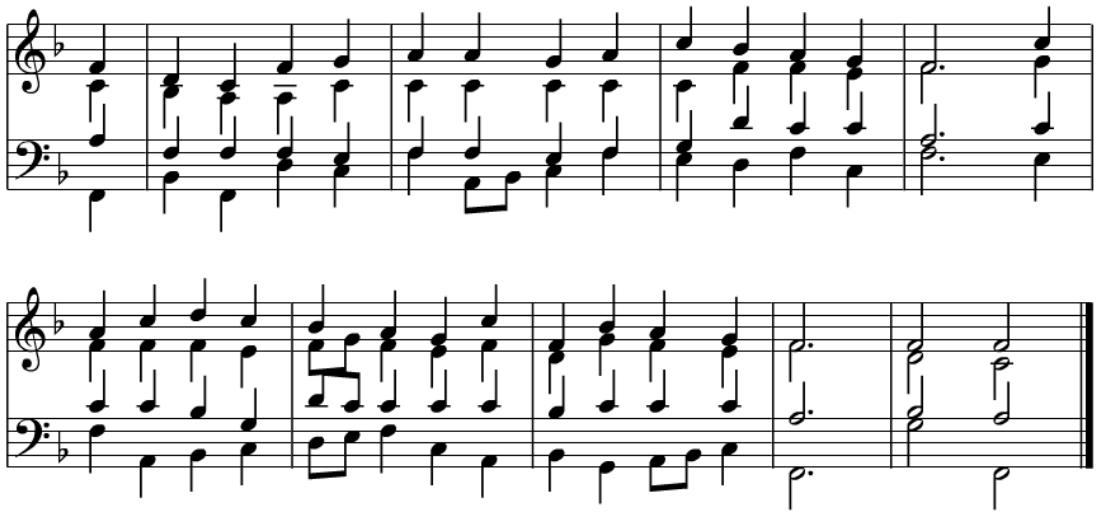


図 4.1: 学習データとして用いた讃美歌コーパスの例

4.1　コーパスを用いた条件付き確率の学習

ベイジアンネットワークの条件付き確率表を学習するために、本研究では讃美歌集 [14] から 254 曲の四声体和声を選びコーパスを作成した[1]。例としてコーパスの 1 曲を図 4.1 に示す。讃美歌は長調の曲に制限し、全てをハ長調に移調してコーパスに用いた。コーパスの讃美歌の 1 曲あたりの曲の長さは約 16 小節である。加えて、Band-in-a-Box[15] のコード推定機能を用いて機械的にそれぞれの音符に対してコードラベルの付与を行った。その際に、Band-in-a-Box のコード推定機能の正確さを調査するために、コーパスの 254 曲の中からランダムに 30 曲を選び、適切なコードラベルが付与されているかチェックを行った結果、コードラベルの正確さは 1 曲あたり平均約 95%という結果になった (図 4.2)。

4 声部のリズムが全て同じだと仮定したので (すなわち 4 声部の発音時刻と消音時刻の時間が同じ)、学習データもこの制約に従うように修正しなければならない。2.1 章で述べたように、全ての声部のリズムは同じとしたので、同時になっている

[1]実装上の制限のため、16 分音符よりも短い音を含む曲は除外した。

音符が他の声部よりも長い場合、その音を分割する。このルールに従って修正したデータで条件付き確率表を学習する際に、同じ音が連続する確率が極端に大きくなってしまう。そこで、分割前の音符の総数に対する分割後の音符の総数の割合にから導いた 0.4 を用いて極端に同じ音が連続する確率にかけることによって確率を下げる。

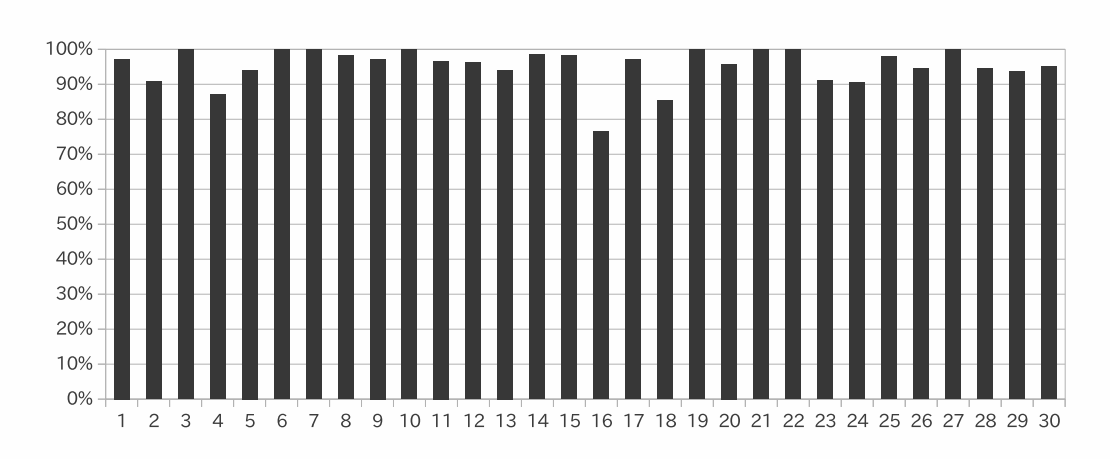


図 4.2: コードラベリングの精度 (横軸はコーパスのデータ番号)

4.2 実験内容

テストデータとして、本研究では日本の音楽学校で和声の学習のために広く用いられている教科書 [16] からハ長調 32 曲のソプラノメロディを用いた。本来ならば、テストデータと学習データの音楽ジャンルは同じべきではあるが、これらのテストデータは和声の第一段階の学習において用いられるソプラノメロディであるため、適切であると考える。これらのテストデータに対して、ノンコードモデルと 2 つのコードモデルを用いて四声体和声の生成を行う。2 つのコードモデルについては、1 つはコードノードが 24 個の要素を持っているコードモデル 1、もうひとつはテンションノートやベース音を区別しコードノードがより多くの要素を

持つコードモデルである。コードモデル 1 の学習では、テンションノートやベース音を削除したコードラベルを用いたが、コードモデル 2 の学習では、そのような削除は行わなかった。本来コードシンボルの候補となりうるものは全部で 41472 になるが、本研究ではコーパスに実際に現れた 110 種類のコードシンボルのみを用いた。条件付き確率表の学習のために、Weka 3.6.8[17] を使い、その後客観評価と主観評価を行った。

4.3　客観評価

本研究では、客観的な基準に基づきノンコードモデルとコードモデルの和声付けの結果を比較した。完璧な客観的な基準を現代の音楽知識で定義することはほとんど不可能であり、客観的な基準での和声の評価非常に難しい。そこで、音楽の専門家との議論より、以下の基準を見出した。

1. 同時性として、基本的に不協和な音程やノンダイアトニックノートは避けるべきである。

2. 異なる声部で同時になる音が同じ音名になってしまうことは単調さの原因となってしまうため、不協和音ではないもののこれもまた避けるべきである。

3. ハ長調の曲では、最後のコードが C で終わることは基本的なルールである。

4. 連続性として、連続する跳躍進行は曲の滑らかさを損なうため避けられるべきである。

5. 始まりから終わりまで同じ音名が続くような極端に単調な旋律もまた避けたほうがよい。

これらの基準はプロが作曲する際に必ずしも適切であるとは限らない。しかし、和声の第一段階として単純に音楽を評価する本研究には適している。連続 5 度やオ

クターブの禁則を用いることもより適切ではあるが、本研究ではそのようなルールをモデルが考慮していないため、用いないものとする[2]。

Criterion1 不協和音を含む和音の総数

不協和音は、短2度、長2度、減5度、短7度、長7度の関係が含まれている和音である。ただし、G7コード (例えば ソプラノ: D, アルト: F, テノール: B, バス: G) や Dm7 コード (例えば ソプラノ: C, アルト: A, テノール: F, バス: D) はその関係が含まれているが不協和音からは除外する。コードモデルではコードと音の関係を明確に学習しているので、非コードモデルよりも不協和音が少なくなると考えられる。

Criterion2 ノンダイアトニックノートの総数

我々が用いたメロディーは実際に音楽大学の和声学で用いられているため、オーソドックスなメロディーになっている。それ故に、結果はノンダイアトニックノートを含まないことが望ましい。

Criterion3 最後のコードが C かどうか

ここで用いられているメロディは全て C コードで終わることが望ましい。もし、非コードモデルの結果がこの基準を満たすならば、最後がトニックコードで終わるという和声の重要なルールの一つを適切に学習できたと言える。一方、コードモデルの結果では、コードの連続性において直接学習しているので、容易にこの基準を満たすと考えられる。

Criterion4 3声部以上で同じ音名の音が選ばれている総数

[2] これらのルールを考慮するためには、ベイジアンネットワークにより多くアークが必要である (例えば S_{i-1} と B_i の間)。

異なる声部で同じ音名が同時に選ばれること (例えば、 ソプラノ: C, アルト: C, テノール: E, バス: C) は不協和音の原因にはならないが音楽を単調にしてしまう。それ故に、このような音の割り振りは避けるべきである。

Criterion5 バス内で連続する音の跳躍 (完全 4 度以上) の総数

連続する音の跳躍は、声部内での音の繋がりの滑らかさを減らしてしまう。特に、バスはメロディを和声的に支える役割を持つ為、バス内での連続する跳躍は避けられるべきである。跳躍の連続を避ける為に、コードモデルではコードの転回形を学習しなければならない。そして、与えられた前後の音から適切に転回を行わなければならない。もし、モデルがこれを十分に学ばなければ、バスはそれぞれのコードのルート音を選び、滑らかさは失われてしまう。

Criterion6 各声部内での音名の総数

2 つや 3 つの音名だけが特定の声部内で現れると、音楽は単調になってしまう。それ故に、この数はなるべく低くあるべきである。

4.4　主観評価

我々は、作曲経験者 3 名に出力結果を聞いてもらい、以下の基準に従って 5 段階評価を行ってもらった (5: 最も良い, 1: 最も悪い)。

A1 不協和な響きの少なさ (同時性)

A2 各声部の旋律の滑らかさ (連続性)

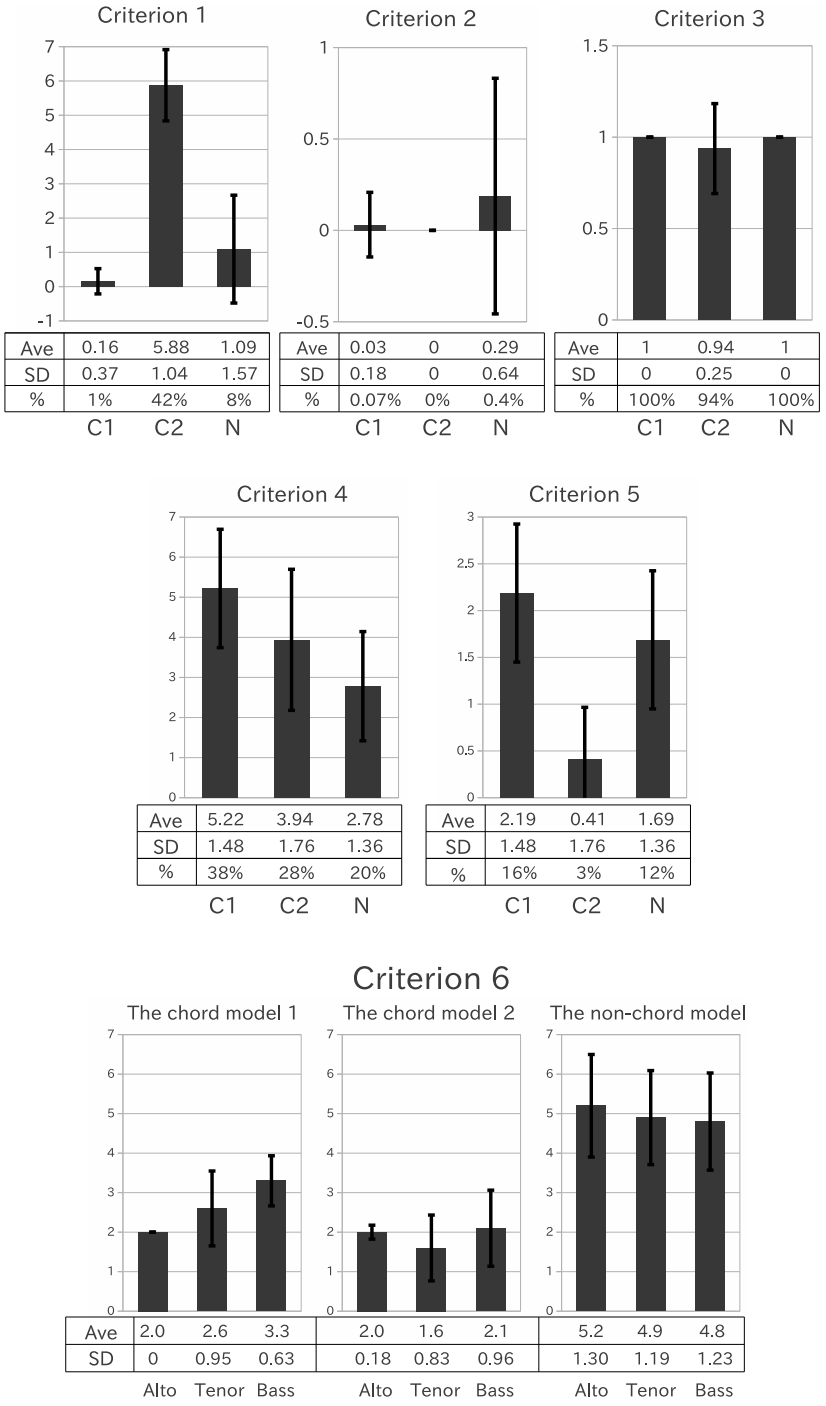


図 4.3: 客観評価の結果 (C1:コードモデル 1; C2:コードモデル 2; N:ノンコードモデル)

4.5　実験結果

図 4.3 と図 4.4 は客観評価と主観評価の結果をグラフにしたものである。

不協和音に関しては、コードモデル 1 では全体の和音の 1%、ノンコードモデルでは全体の 8% が不協和音だった。コードモデル 1 ではコードと音符間の関係を直接学習したので、適切なコードが推論されれば適切なコードトーンが選ばれた。ノンコードモデルでは声部間の関係のみを学習したのでほとんどの結果で不協和な音程を避けていた。対照的に、コードモデル 2 では全体の和音の 42%が不協和音であった。おそらくこれはコードモデル 2 はそれぞれのコードノードが 110 個もの要素を持っているため、解決困難なデータスパースネスが引き起こされたからだと考えられる。後にも述べるが、実際にこのモデルはほとんどのソプラノに対して、同じ音 (アルト: C, テノール:G , バス: C) を選んでいた。これは、データが制限されている場合に大きな規模の確率モデルで学習を行った時の典型的な現象である。

主観評価に関しては、同時性に関しての基準に関してはノンコードモデルとコードモデル 1 の結果は平均 4.5 以上となった。コードモデル 2 では他と比べて低くなった。これはこのモデルで生成された和声は多くの不協和音があったからだと思われる。

ノンコードモデルではほとんどの不協和な音程は計算コストを下げるために省略したアルトとテノール間に現れた。アークによって互いにつながっている声部では不協和音がないので、もしより多くの学習データによってアークを追加することが出来れば、不協和音を減らせると考える。

コードモデル 1 の結果ではノンダイアトニックノートはひとつだけであった。それに対して、ノンコードモデルではいくつかのノンダイアトニックノートが現れた。これらの結果より、これらのモデルはハ長調で扱われる典型的な音の使い方を学習したと考えられる。コードモデル 2 ではノンダイアトニックノートはひとつもなかった。これはおそらくモデルが学習する際に生じたデータスパーネスの

ため、結果のほとんどが同じコード (アルト: C, テノール: G, バス: C) となったためだと考えられる。

全ての曲においてノンコードモデルとコードモデル 1 では最後のコードは C メジャーコードで終わっていた。これよりこれらのモデルは最後のコードがトニックコードで終わるという基本的なルールを学習したと考えられる。これとは対照的に、コードモデル 2 では C メジャーコードで終わっていない結果は 2 つだった。

コードモデル 1 と 2 では、3 声部以上で同じ音名からなる和音 (例えば 4 声部で C-G-C-C になる) はそれぞれ全体の和音の 38%と 28%であった。一方コードモデルの結果に比べ、ノンコードモデルでは 20%となった。これによって、ノンコードモデルの結果はコードモデルの結果よりも単調さが低くなっていた。

コードモデル 1 の結果では、バスでの連続する大きな (完全 4 度もしくはそれ以上) 進行がノンコードモデルと比べ多い結果となった。この原因として、同時に推論されるノードであるコード (C_i) とバス (B_i) の依存関係がバスのノードである (B_{i+1}, B_i, B_{i+1}) よりも強くなったからだと考えられる。結果として、コードモデルではほとんどのバスの音はコードのルート音をとっており、それが原因で単調になってしまった。コードモデル 2 ではコードモデル 1 よりも大きな跳躍は少なかった。これはモデルがバス声部でのなめらなか旋律の流れを学習したのではなく、ほとんどの場合でバスは C の音になっていたからである。

2 つのコードモデルでは、各声部の音名の総数は 2 から 4 だった。一方、ノンコードモデルでは 4 以上となった。実際に、コードモデルの結果でのアルトとテノールでは 1,2 種類の音が繰り返されているだけなので単調になっていた。しかし、ノンコードモデルでは各声部で滑らかなメロディになっていた。客観評価においても、2 人の被験者による連続性の評価はコードモデル 1 よりもノンコードモデルの方が少しではあるが高くなっていた。

以上のことより、以下の仮定は証明された。

1. ヴォイシングやテンションノートを区別しないコードモデルでは、異なるヴォ

イシングが好ましい場合でさえも、同じコードに対して同じヴォイシングが生成されてしまう傾向にある。結果として、生成された和声は滑らかさに欠ける。

2. ヴォイシングやテンションノートを区別したコードモデルでは、限られた学習データではデータスパース問題が発生し、適切な和声は生成されない。

3. ノンコードモデルでは、コードモデルの結果よりも滑らかな和声が生成される。

4.6　考 察

ここでは、いくつかの結果を例にノンコードモデルとコードモデルがどのような和声を学習したのか考察する。

4.6.1　音の配置の学習

図 4.5 は Sample 3 を用いたノンコードモデルの結果である。この結果では不協和音は見られない。この結果では C, Dm, Em, F, G7, Am のような様々なコードが見られる。これより、ノンコードモデルが讃美歌から適切に音の配置について学習できたことを示している。同じ Sample 3 を用いたコードモデル 1 の結果でもまた不協和音は見られないが、コードは C, F, G の 3 つしか見られない。更に、音の配置もノンコードモデルとは異なっている。ノンコードモデルでコード C が連続する場合、ソプラノが C ならテノールには G、ソプラノが E ならテノールには C が配置されており、同じコードであっても異なる配置をすることで曲が単調になることを避けている。コードモデル 1 において、アルトは全て C か B、テノールでは G か A しか使われておらず、これが単調さの原因となっている。コードモ

デル 2 の結果では 6 つの不協和音が見られた (図 4.7)。しかし、これら 3 つの結果は連続 5 度やオクターブのようないくつかの禁則を犯している。

4.6.2 典型的なカデンツの学習

図 4.8 は Sample 11 を入力としたノンコードモデルでの結果である。この結果でコード進行は以下のようになった。

|C Dm C F |G7 C G7 G7 |C G7 C Dm |C G7 C |

各声部で全てコードトーンになっているため、学習の過程でコードシンボルを与えなくても、よく使われるコードのコードトーンを適切に学習できたと考えられる。加えて、コード進行は以下のような機能和声になっていた。

C (tonic) → G7 (dominant) → C (tonic),

Dm (subdominant parallel) → C (tonic) → G7 (dominant).

特に、C → Dm → C → G7 → C のように典型的なコード進行で曲が終わっている。トライトーンの解決のために、直前のコード C に G の代わりに G7 が適切に使われている。バスの旋律も基本的に順次進行と 5 度の跳躍進行によって構成されているため、音楽的に自然ば旋律になっている。しかし、この結果に関してもいくつかの連続 5 度やオクターブの禁則を含んでいる。

図 4.9 は Sample 11 を入力したコードモデル 1 の結果である。コード進行は C, F, G の 3 つだけで構成されており、単調さの原因となっている。ただし、テノールでの第 3 音は A ではなく G、アルトの第 5 音では C ではなく B になっている。バスの音は全てコードのルート音になっており、これによってより単調さが増している。

図 4.10 はコードモデル 2 の結果である。Sample 3 のように、アルト、テノール、バスのそれぞれの声部でほとんどが C, G, C の音のみになっている。これは不協

表 4.1: ノンコードモデルで出力された典型的なカデンツ

Samples	Cadences		
Samples 1, 6, 9, 30, 32	F	G7	C
Samples 2, 28	C	Bm(-5)/D	C
Sample 3	C/G	G7	C
Samples 4, 16, 29, 31	G7	G7	C
Samples 5, 10, 11, 13, 22	C	G7	C
Samples 17, 20	C	F	C
Sample 7	C	G/D	C

和音や単調さの大きな原因となっている。これらの結果も同様に多くの連続 5 度やオクターブの禁則に反している。

ノンコードモデルでは他の結果に関しても同様に典型的なカデンツとなっていた。表 4.1 はそれを表にしたものである。これらのカデンツの全ては実際の音楽に一般的に使われており、ノンコードモデルはこれらのカデンツのほとんどを学習できたと思われる。

4.6.3　不協和音が現れた結果

ノンコードモデルでの結果で不協和音が現れた結果を図 4.11 に示す。この結果では、11 番目の和音と 12 番目の和音が不協和音となっていた。これらの不協和音はテノール・バス間もしくはソプラノ・テノール間で長 2 度や短 2 度の音程にあったため生じた。実際に、この曲の同時性に関しての評価結果は 1 と 32 曲の中で一番低かった。コードモデル 1 の結果 (図 4.12) では、不協和音は 1 つだけだった。しかし、Sample 3(図 4.6) と同様にバスのほとんどの音がルート音をとっていたり、コードが C, F, G のみだったりと単調な曲になった。コードモデル 2 の結果

(図 4.13) では、それぞれの声部で同じ音を繰り返しているため、7 つの不協和音が見られた。これらの結果にも、多くの連続 5 度やオクターブが見られた。

4.6.4 連続 5 度とオクターブ

3 つのモデルの全ての結果には連続 5 度やオクターブが含まれていた。ノンコードモデルの結果である図 4.16 では、連続 5 度とオクターブは 8 つ見られた。これはモデルに、T_i と B_{i-1} または T_{i-1} と B_i の間を考慮するアークがないためである。理想を言えば、そのような箇所にアークを加えるべきだが、それによってより多くの学習データが必要になってしまう。また、コードモデル 1 の結果 (図 4.14) では 10 箇所、コードモデル 2 では 22 箇所に連続 5 度やオクターブが現れた。

4.6.5 まとめ

コードモデル 1 ではほとんどの結果で、コード進行に C, F, G しか現れなかったり、バスの旋律はほとんどがコードのルート音をとっていたので、単調な結果になってしまった。コードモデル 2 では多くの不協和音や各声部での同じ音の繰り返しが見られた。一方、ノンコードモデルでは様々なダイアトニックコードを使うことで滑らかな和声を実現し、バスの旋律ではほとんどが順次進行と 5 度の進行になっていた。しかしながら、禁則という部分ではまだ多くの連続 5 度やオクターブが見られた。

4.7 讃美歌の再生成

もう一つの評価のアプローチとして、既存の讃美歌のメロディに対して、再度四声体和声付けを行った。もし、それぞれのソプラノに対する和声付けの結果が元の讃美歌と同じならば、和声付けの結果は正しいと見なせる。しかし、結果が

同じだからと言って、この基準が音楽的に正しいということではないことに注意しなければならない。本当に音楽的に正しいかを現代の音楽知識で定義することはほとんど不可能である。そのため、本研究では既に述べているようにいくつかの単純な基準を用いて実験を行った。ここでは、違う基準として元の讃美歌と同じ和声になっているのかという基準で評価を行う。

既に述べてある 254 曲の讃美歌コーパスを用いて、その中からテストデータとしてランダムに 30 曲を選び出した。ベイジアンネットワークには残りの 224 曲を学習させ、残りの 30 曲で四声体和声の生成を行う。図 4.17 は 30 曲それぞれの讃美歌で、元の讃美歌に対する生成された結果の一致度の割合をプロットしたものである。トータルでは、コードモデル 1 では生成された音の 35%、ノンコードモデルでは 43%が元の讃美歌 [14] の音と同じであった。アルトとテノールについては、ノンコードモデルの一致度はコードモデル 1 よりも 10%以上高くなっていた。バスに関しては大きな差は見られなかった。しかし、これらの結果は 35%もしくは 43%が音楽的に適切だったという意味ではない。これらの割合は、音楽的に許容できるが元の讃美歌と一致していない音に関しては除外した結果である。

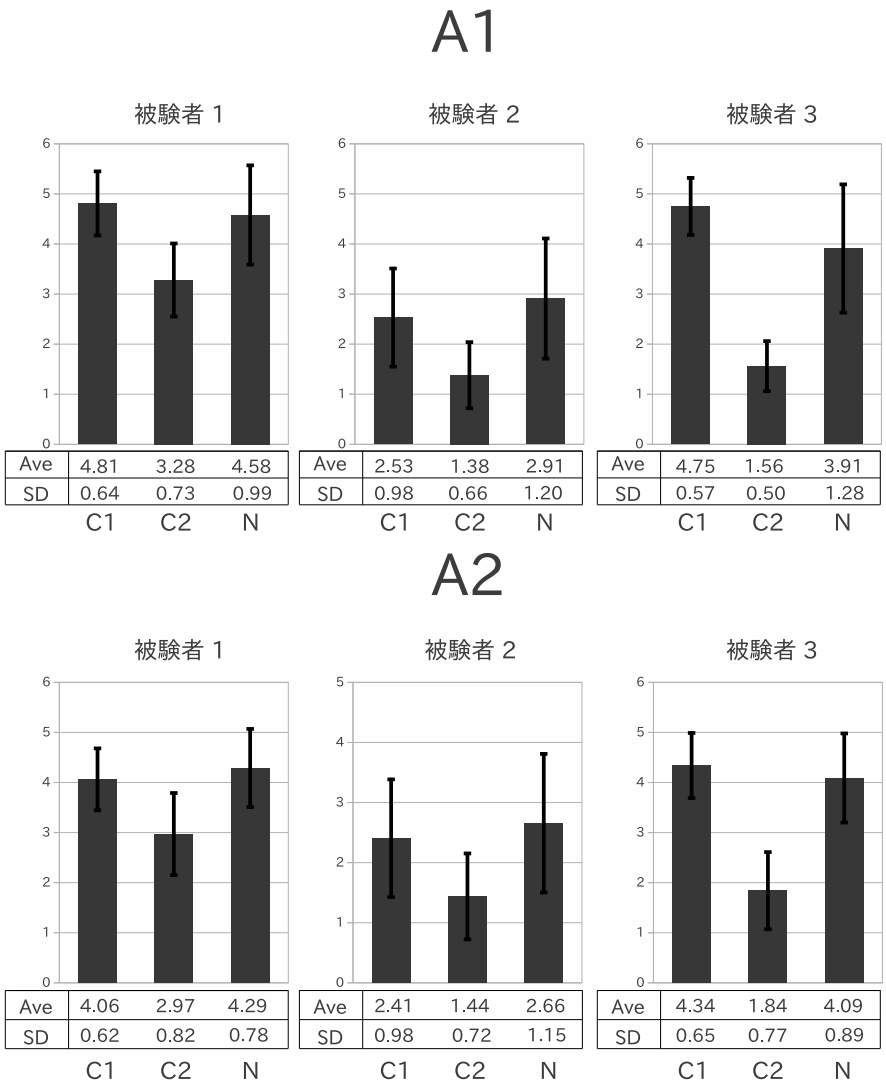

A1
被験者 1
被験者 2
被験者 3
Ave 4.81 3.28 4.58
SD 0.64 0.73 0.99
Ave 2.53 1.38 2.91
SD 0.98 0.66 1.20
Ave 4.75 1.56 3.91
SD 0.57 0.50 1.28
C1 C2 N
A2
被験者 1
被験者 2
被験者 3
Ave 4.06 2.97 4.29
SD 0.62 0.82 0.78
Ave 2.41 1.44 2.66
SD 0.98 0.72 1.15
Ave 4.34 1.84 4.09
SD 0.65 0.77 0.89
C1 C2 N

図 4.4: 主観評価の結果

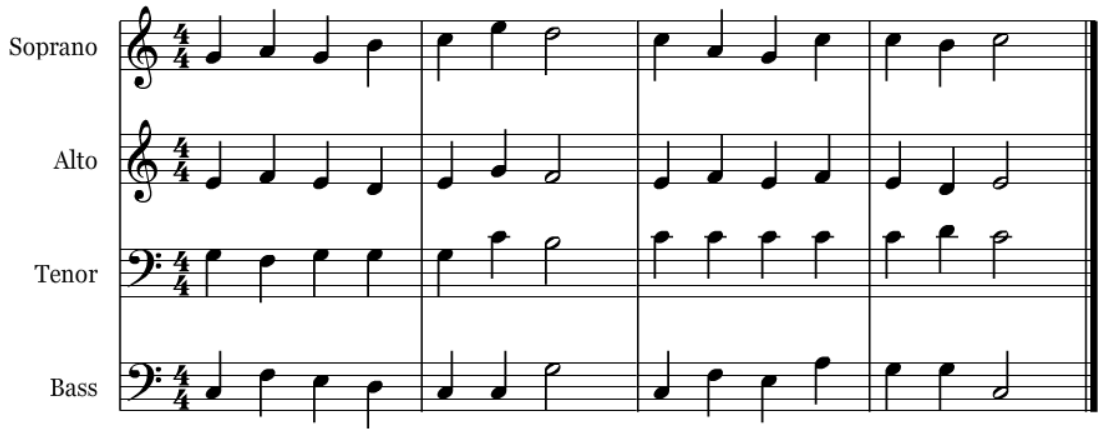


図 4.5: ノンコードモデルを用いた Sample 3 の結果

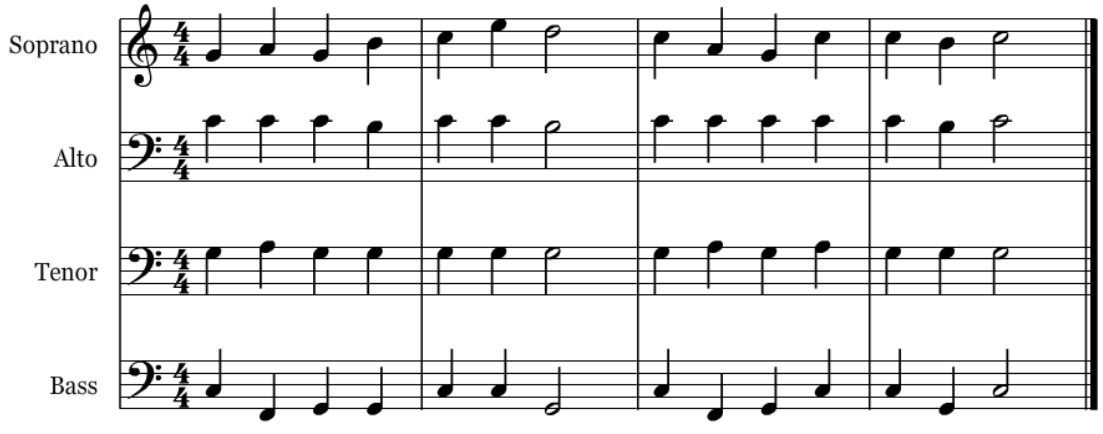


図 4.6: コードモデル 1 を用いた Sample 3 の結果

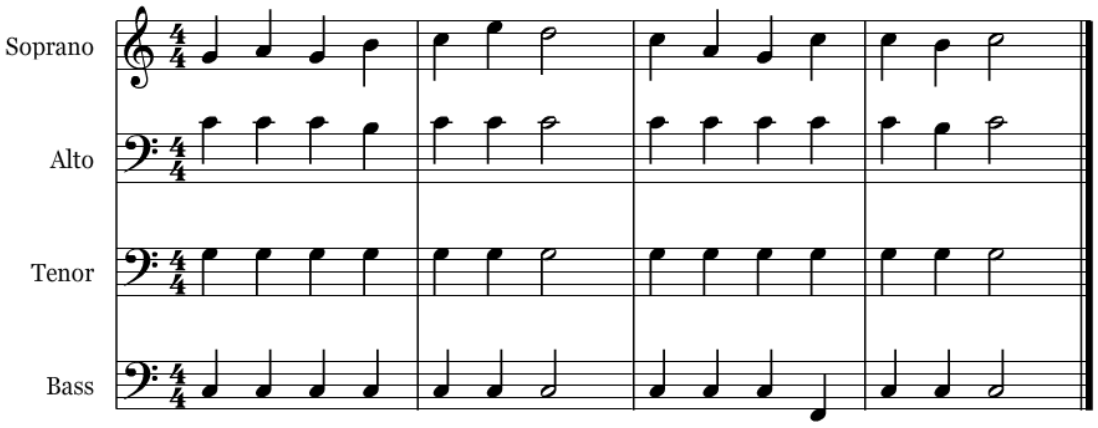


図 4.7: コードモデル 2 を用いた Sample 3 の結果

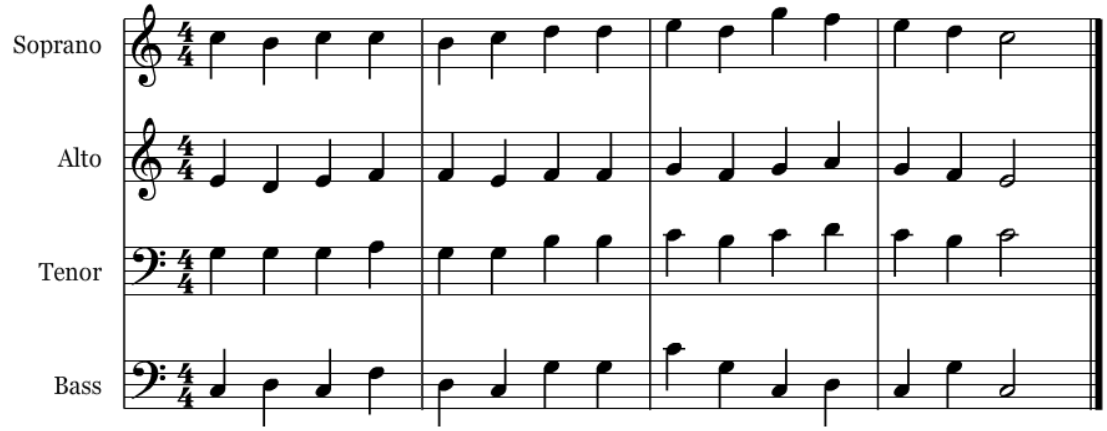


図 4.8: ノンコードモデルを用いた Sample 11 の結果

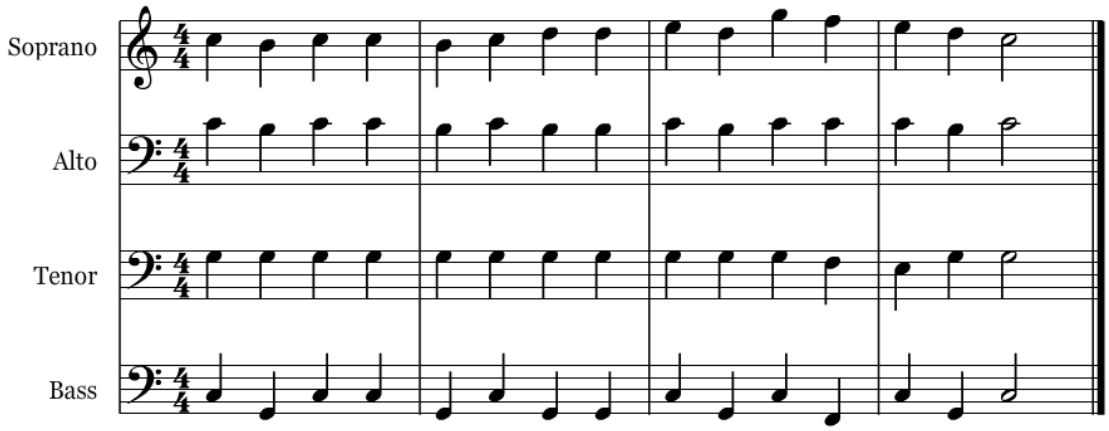


図 4.9: コードモデル 1 を用いた Sample 11 の結果

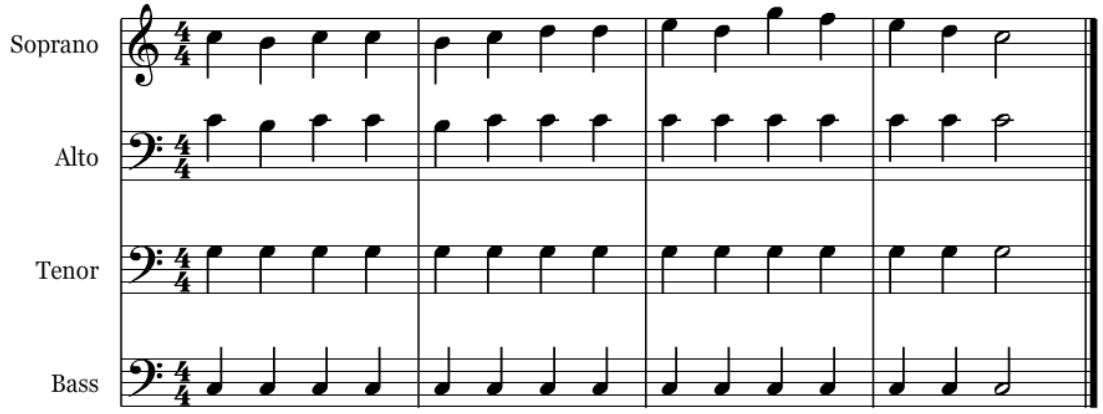


図 4.10: コードモデル 2 を用いた Sample 11 の結果

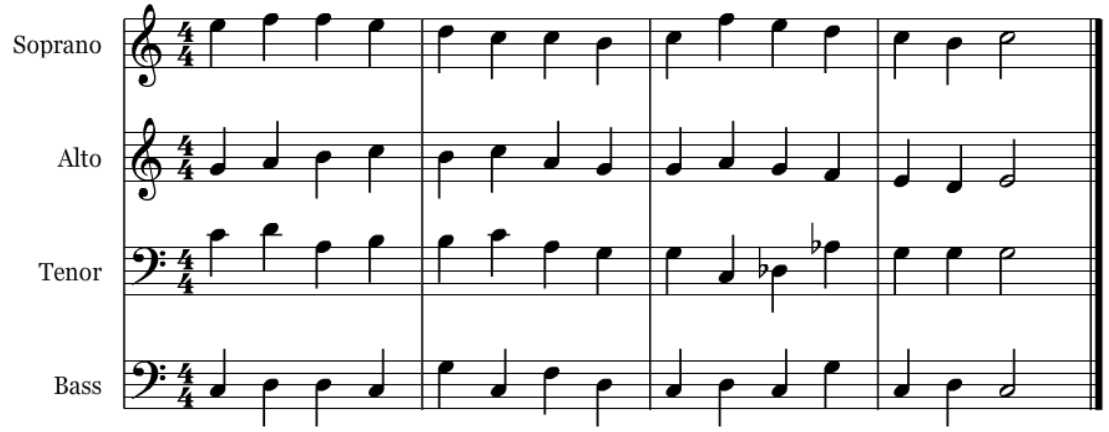


図 4.11: ノンコードモデルを用いた Sample 10 の結果

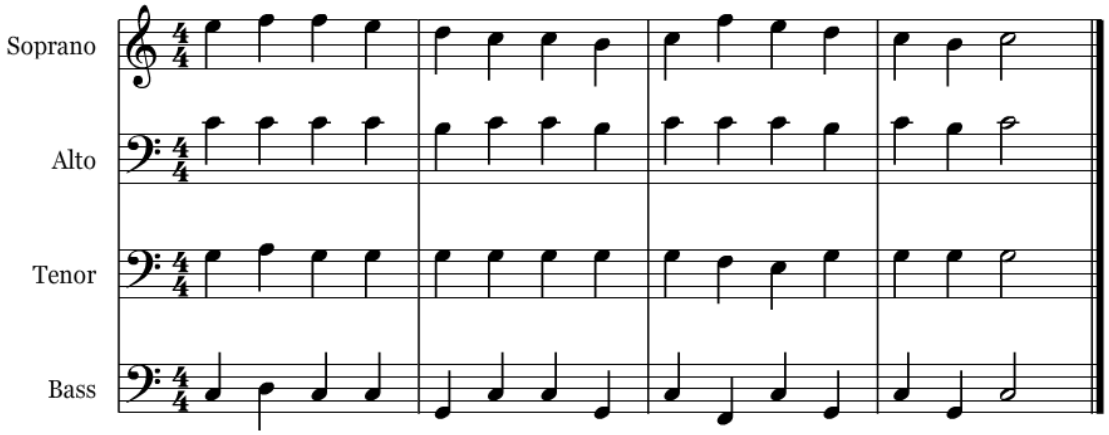


図 4.12: コードモデル 1 を用いた Sample 10 の結果

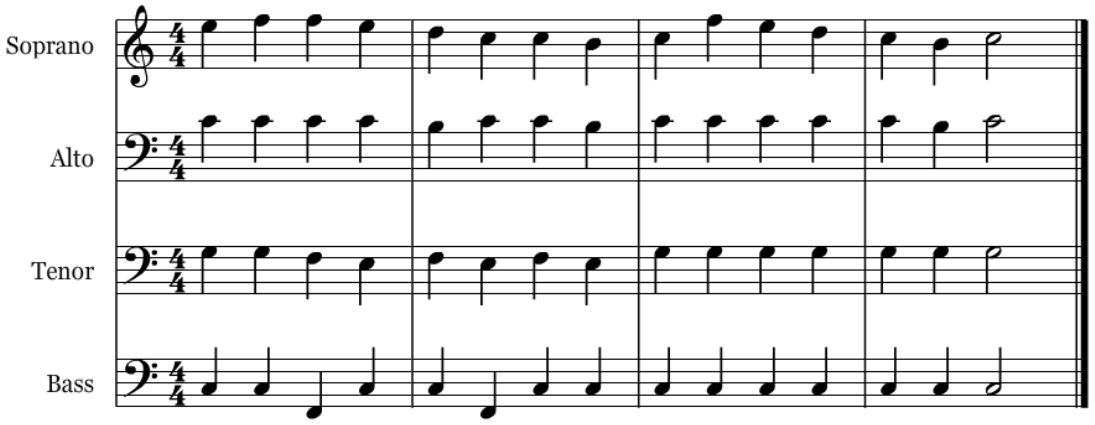


図 4.13: コードモデル 2 を用いた Sample 10 の結果

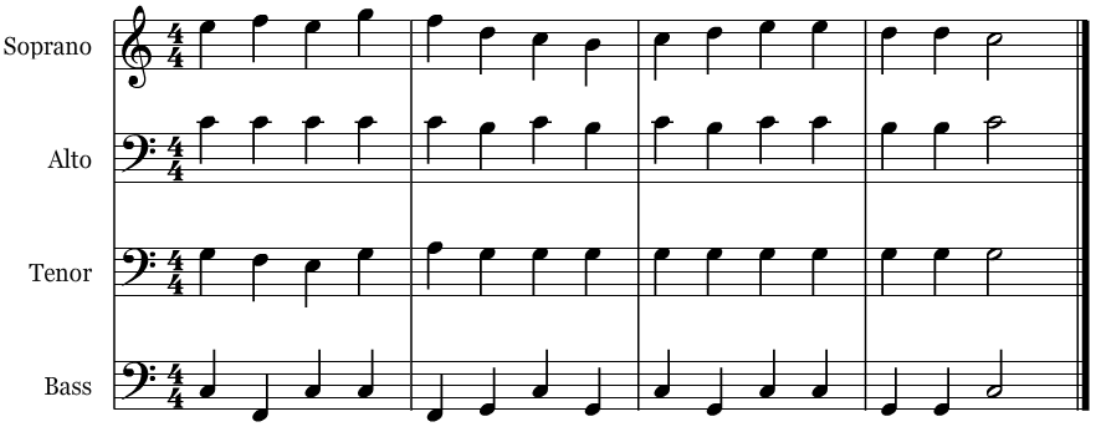


図 4.14: コードモデル 1 を用いた Sample 4 の結果

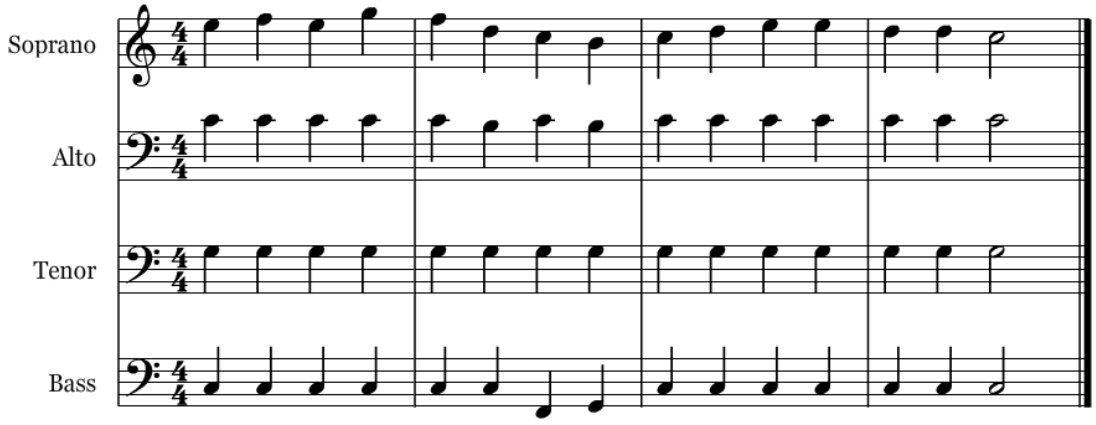


図 4.15: コードモデル 2 を用いた Sample 4 の結果

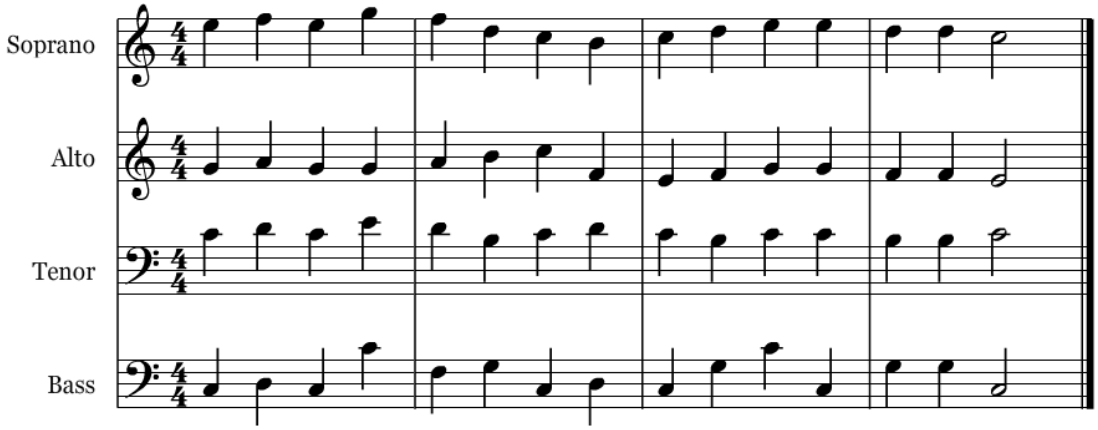


図 4.16: ノンコードモデルを用いた Sample 4 の結果

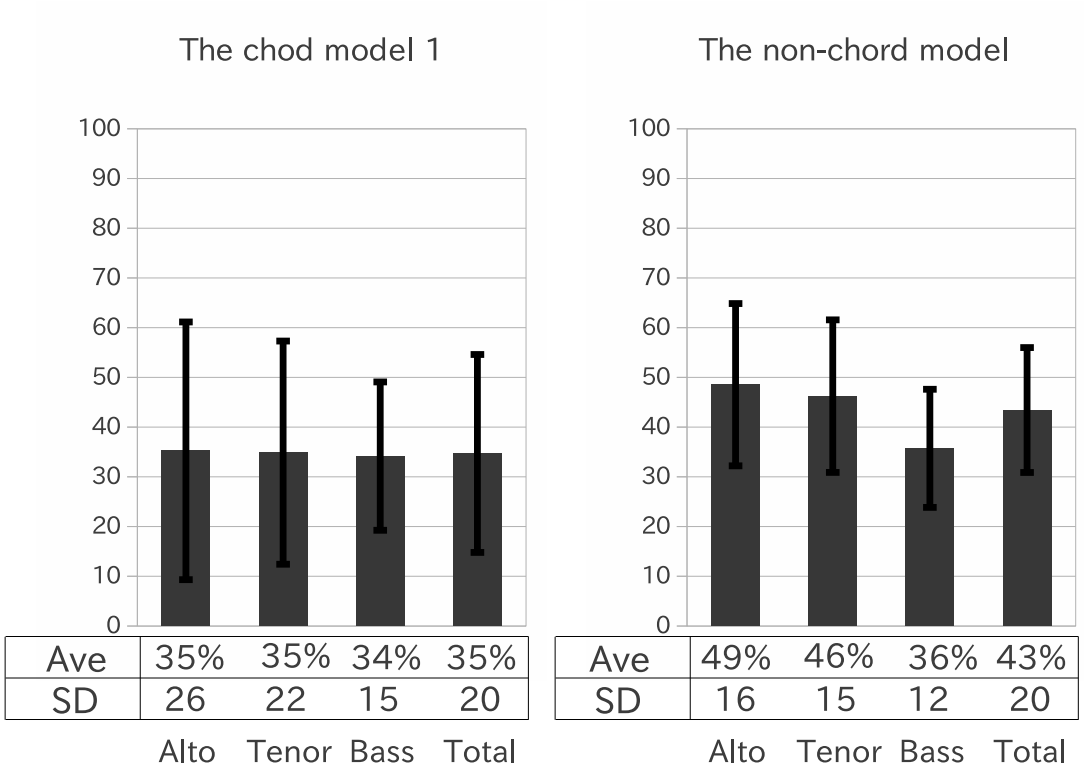


図 4.17: 各声部（アルト・テノール・バス）の一致度と全体の一致度

第5章　ノンコードモデルを用いたモデル構造の評価(実験2)

本研究で提案したノンコードモデルは3.2章で示したように、(1) バスからアルト・テノールにアークを持ち、(2) $(i+1)$ 番目のノードを持つ。この章では、これらの効果を評価するために、以下のモデルを比較する。

モデル 1: 3.2章で述べたモデル

モデル 2: $(i+1)$ 番目のノードがないモデル

モデル 3: バスからアルト・テノールにアークがないモデル

モデル 4: $(i+1)$ 番目のノードも、バスからアルト・テノールにアークがないモデル

5.1　実験内容

実験内容は4章で示したものと同じである。実験1で用いた32曲のソプラノメロディを用いて、表5.1で示された4つの異なるモデルから四声体和声を生成する。図9-12で示したこれら4つの異なるモデルはそれぞれモデル1, モデル2, モデル3, モデル4と呼ぶ。モデル1は実験1で用いたノンコードモデルである。

表 5.1: 実験2で用いるモデル

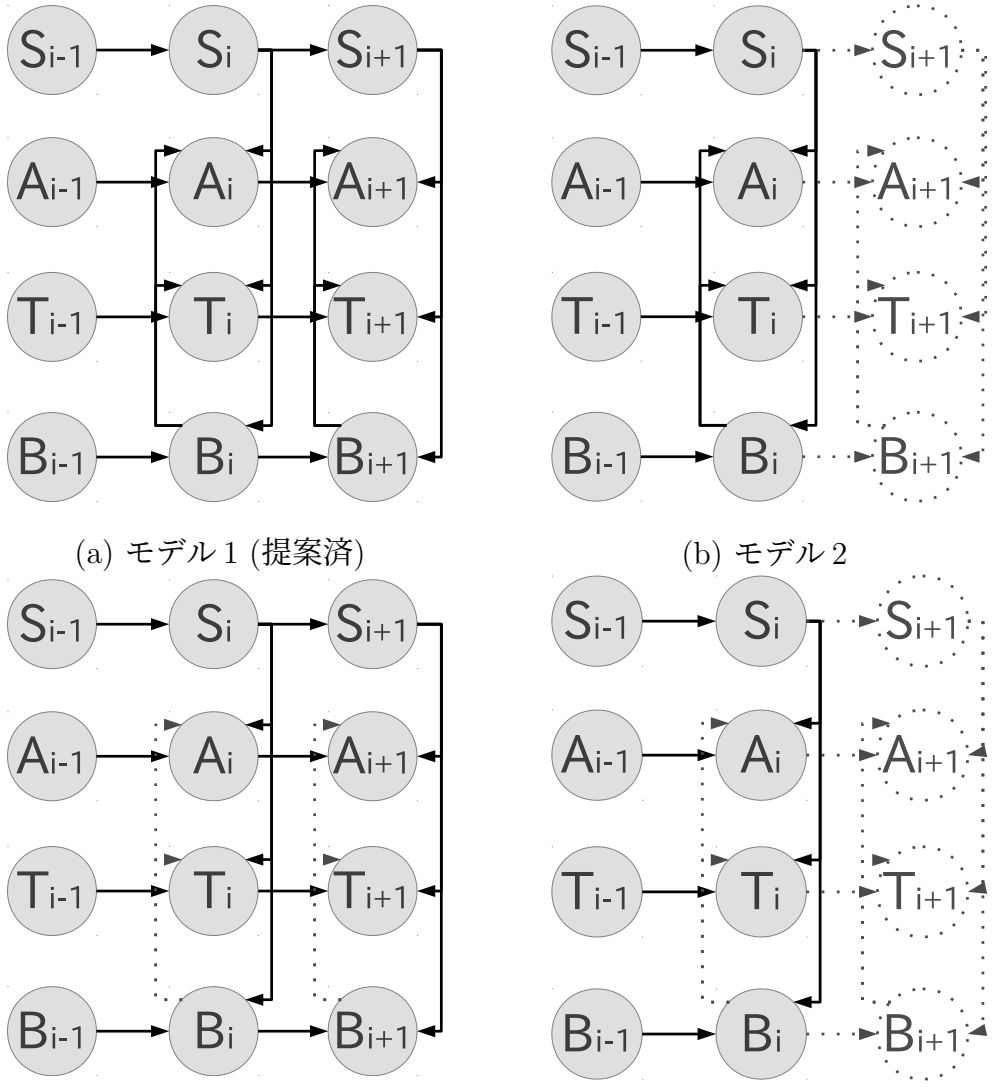
	バスからアルト/テノールへのアーク	$(i+1)$ 番目の音符のノード
Model 1	○	○
Model 2	○	×
Model 3	×	○
Model 4	×	×

5.2 実験結果

図5.2は実験結果をグラフにしたものである。不協和音について、バスからアルトとテノールにアークを持つモデル1とモデル2の結果は、モデル3とモデル4の結果よりも不協和音が少なかった。この理由として、モデル3と4ではアルト・バス間とテノール・バス間のアークを省略したために、不協和音がより多く現れたと考えられる。実際に、モデル3での不協和音105個のうち不協和音はアルト・バス間で26個、テノール・バス間でも26個の不協和音が現れていた。

モデル1と3では、$(i+1)$ 番目と i 番目の音の確率が最大になる音の繋がりを考慮した上で、i 番目の音を選んでいる。これによって、もしも $(i-1)$ 番目の音が与えられ i 番目の音が不協和音となる音の組み合わせの確率が高くなってしまっても、$(i+1)$ 番目を考慮することによってその確率を下げ、不協和音を避けれるかもしれないと考えた。実際に、不協和音の総数では、不協和な音程を $(i+1)$ のノードを導入することで減らすことができた。

それに対して、$(i+1)$ 番目のノードがないモデル2とモデル4では、i 番目の音を推論する際に $(i+1)$ 番目の音は考慮にいれていない。その為、一度不協和音の原因となる音の組み合わせが決まってしまったら、システムは、$(i+1)$ 番目の音を i 番目の和音が不協和音の条件で推論しなければならない。その場合、学習データにほとんど不協和な音の組み合わせはないので、$(i+1)$ の音の組み合わせに関し



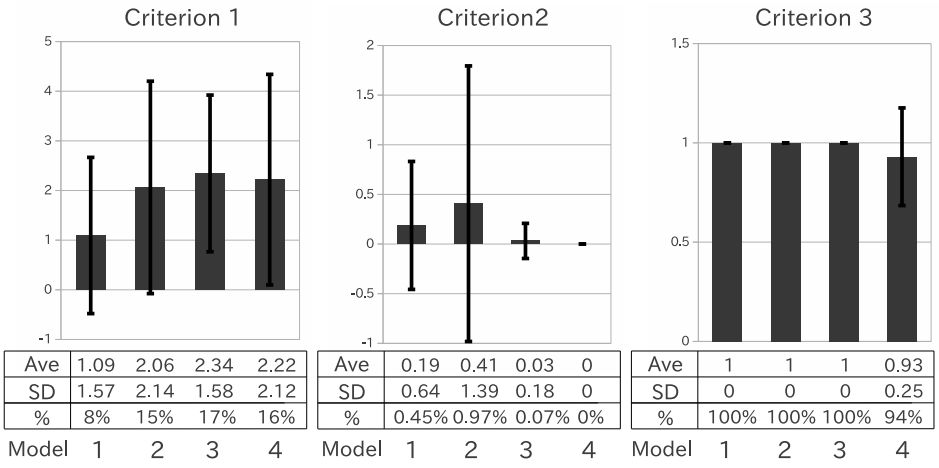
(a) モデル 1 (提案済)　(b) モデル 2

(c) モデル 3　(d) モデル 4

図 5.1: 4 つのノンコードモデル

ても適切に推論することができないので、結果的に不協和な音の組み合わせを選んでしまうかもしれない。実際にモデル 2 の結果では、Sample 10 の結果の図 5.3 と Sample 18 の結果である図 5.4 では連続して不協和音が生成された。

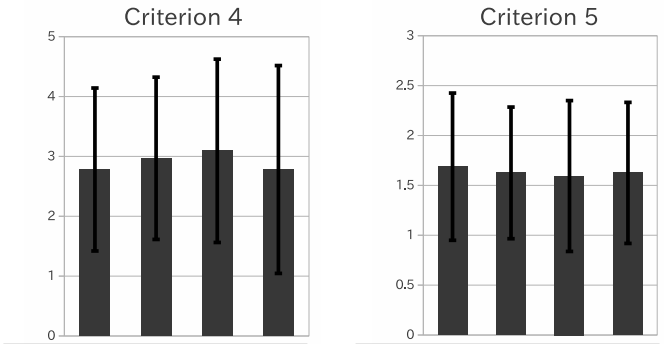
ノンダイアトニックノートの総数 (Criterion 2) に関しても、不協和音の総数 (Criterion 1) についての理由と同様に $(i+1)$ 番目のノードを導入することによって減らすことができた。モデル 3 はモデル 1 よりもノンダイアトニックノートは



	Model 1	Model 2	Model 3	Model 4
Ave	1.09	2.06	2.34	2.22
SD	1.57	2.14	1.58	2.12
%	8%	15%	17%	16%

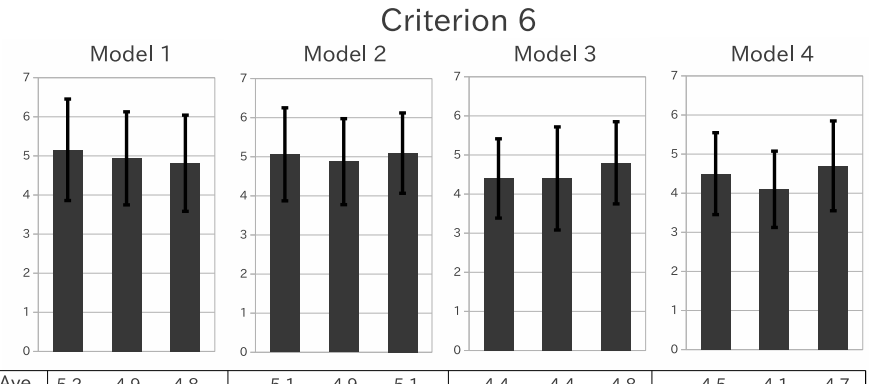
	Model 1	Model 2	Model 3	Model 4
Ave	0.19	0.41	0.03	0
SD	0.64	1.39	0.18	0
%	0.45%	0.97%	0.07%	0%

	Model 1	Model 2	Model 3	Model 4
Ave	1	1	1	0.93
SD	0	0	0	0.25
%	100%	100%	100%	94%



	Model 1	Model 2	Model 3	Model 4
Ave	2.78	2.97	3.09	2.78
SD	1.36	1.36	1.53	1.74
%	20%	21%	22%	20%

	Model 1	Model 2	Model 3	Model 4
Ave	1.69	1.63	1.59	1.63
SD	0.74	0.66	0.76	1.63
%	12%	12%	11%	12%



	Model 1			Model 2			Model 3			Model 4		
Ave	5.2	4.9	4.8	5.1	4.9	5.1	4.4	4.4	4.8	4.5	4.1	4.7
SD	1.30	1.19	1.23	1.19	1.10	1.03	1.01	1.32	1.05	1.05	0.98	1.15
	Alto	Tenor	Bass	Alto	Tenor	Bass	Alto	Tenor	Bass	Alto	Tenor	Bass

図 5.2: 客観評価の結果

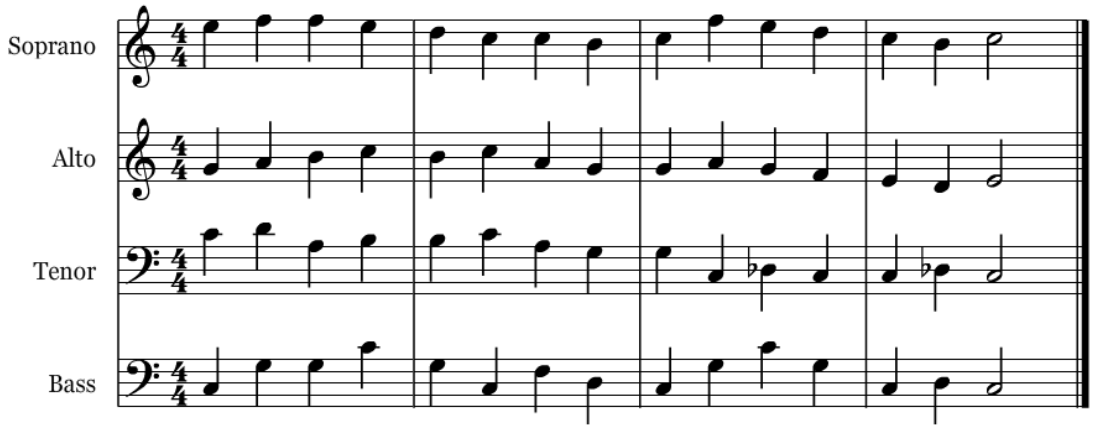


図 5.3: モデル 2 を用いた Sample 10 の結果

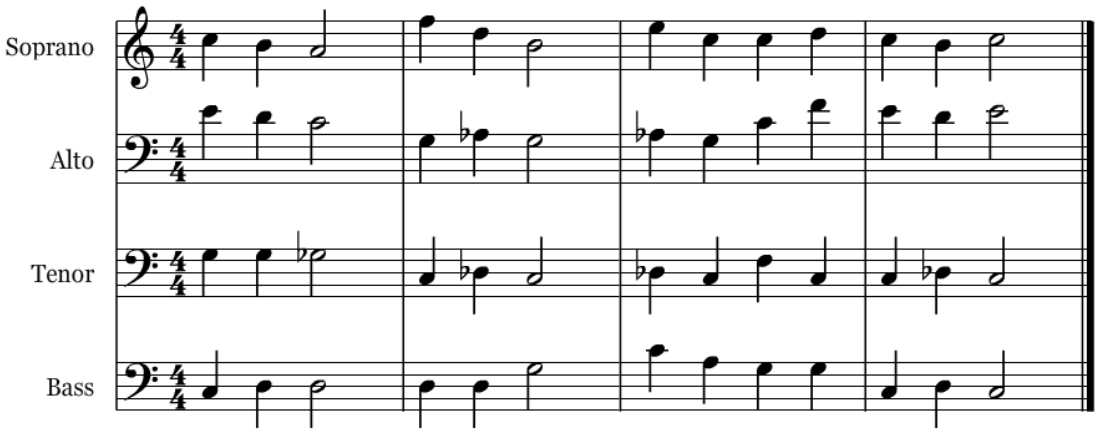


図 5.4: モデル 2 を用いた Sample 18 の結果

少なかった。Criterion 3 ではモデル 4 の 2 つの結果で最後のコードが C で終わっていなかった。対して、他のモデルでは全ての結果が最後のコードが C で終わっていた。最後の音の組み合わせを推論する際に、最後の音かどうかをモデル 1 と 3 では考慮している。モデル 2 では、ソプラノが C の場合にコード C の音を選ぶ傾向にあるため、3 つのモデルでは最後のコードが全て C になったと考えられる。一方、モデル 4 の結果では Sample 18 (図 5.5) と Sample 30 (図 5.6) の結果で最後のコードが Am で終わっていた。Am で曲が終わることは、偽終止と呼ばれよく実

際の音楽にも使われるような進んだ技術である。しかしながら、オーソドックスな和声の観点から必ずしもこれは適切とは限らないと考える。Criterion 4, 5, 6 に関してはモデル間で大きな差は見られなかった。

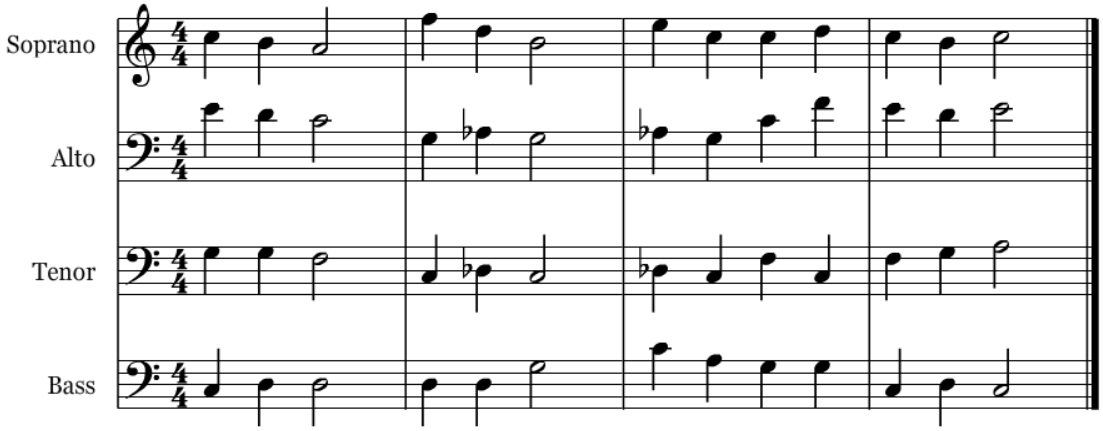


図 5.5: モデル 4 を用いた Sample 18 の結果

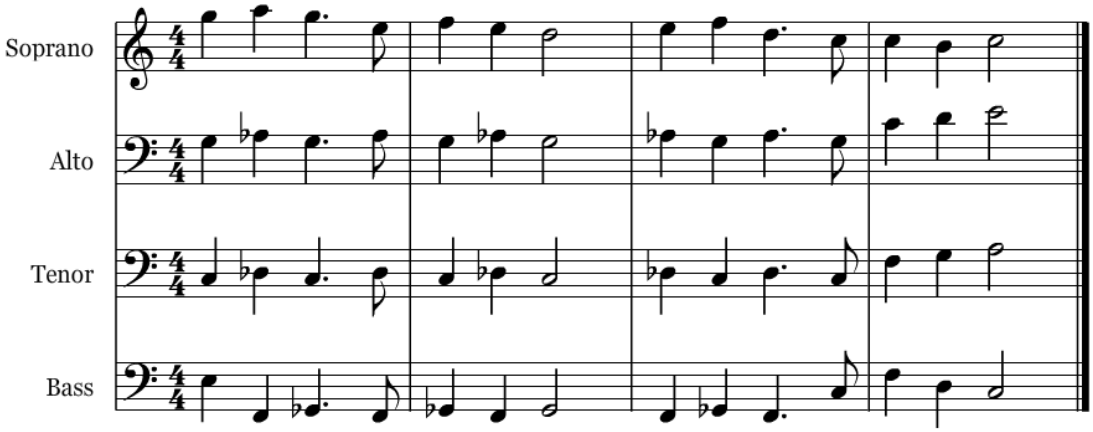


図 5.6: モデル 4 を用いた Sample 30 の結果

第6章　既存研究との比較(実験3)

本論文では、既存研究 [8, 9, 10] と同じメロディを用いて四声体和声付けを行い、その結果と既存研究で比較を行った。ここでは、本研究のモデルからノンコードモデルを用いて実験を行う。我々のモデルではハ長調を前提としているため、既存研究のメロディは一度ハ長調に移調し、和声付けを行った後に、元の調に再度移調を行う。

6.1　Allan らの研究と本研究との比較

図 6.1 は、同じテストデータを用いた Allan らの手法の結果と本研究の手法の結果である。両方の結果では、ほとんどの和音が協和音となっている。連続性の観点として、バスの旋律の動きは両結果で似ている。しかし、本研究の結果ではアルトとテノールは全て順次進行になっているが、Allan らの結果では順次進行といくつかの跳躍進行が現れていた。

図 6.2 は、他のメロディを用いた Allan らの結果と本研究の結果である。Allan らの結果に不協和音は現れていない。しかし、本研究の結果では 20 個の不協和音が現れた。さらに、このメロディをハ長調に移調した場合、一番高い音が A5 よりも高い音になってしまう。我々のモデルはソプラノの音域を C4 から A5 までとしているため、そのようなソプラノに対しては和音を出力することができなかった。

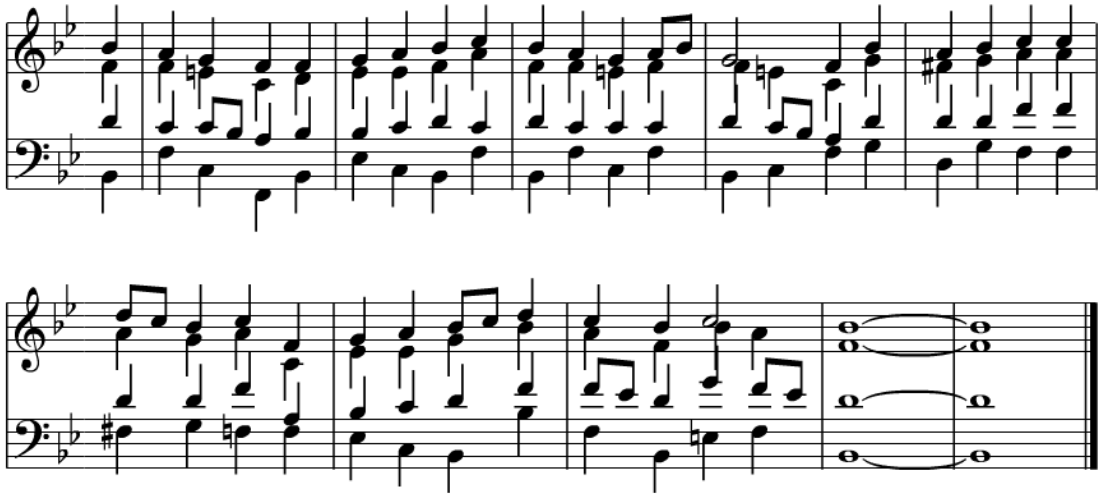
6.2　Hild らの研究と本研究との比較

図 6.3 は、同じメロディを用いた Hild らの結果 [8] と本研究の結果である。各声部での音域が、本研究の結果の方が Hild らの結果よりも高くなっている。

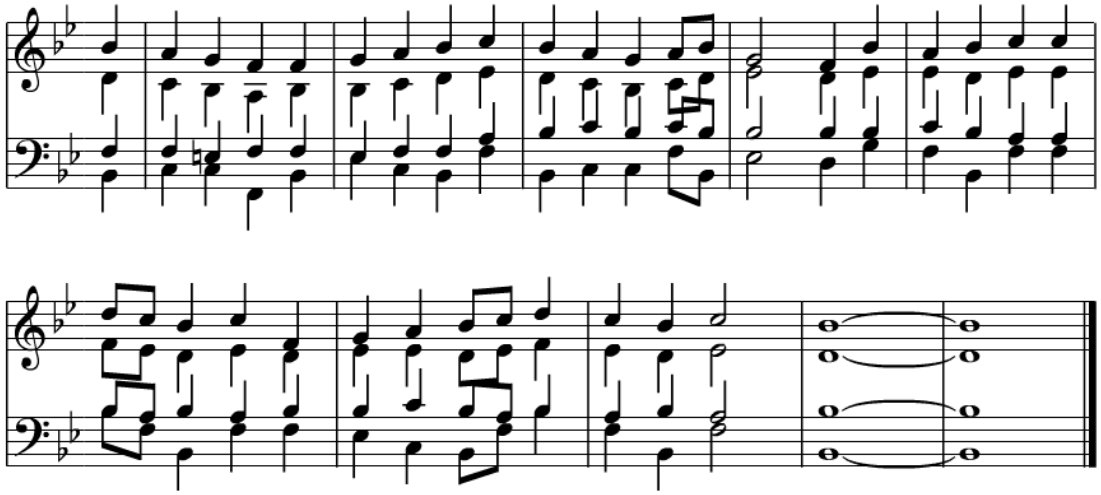
図 6.4 違うメロディを用いた Hild らの結果 [8] と本研究の結果である。本研究の結果では若干の不協和音があるが、Hild らの結果には不協和音はない。

6.3　Yi らの研究と本研究との比較

Yi らの結果 [10] と本研究の結果を図 6.5 に示す。2 つの結果は、両方共に 1 つだけ不協和音がある。最後のコードに関しては、我々の結果では F になっていたが、Yi らの結果では C になっていた。アルトとテノールの旋律では、我々の結果はひとつだけ跳躍進行があるのに対して、彼らの結果ではいくつかの跳躍進行がみられた。これら 2 つの結果には、禁則が含まれているため、これは今後の課題となる。

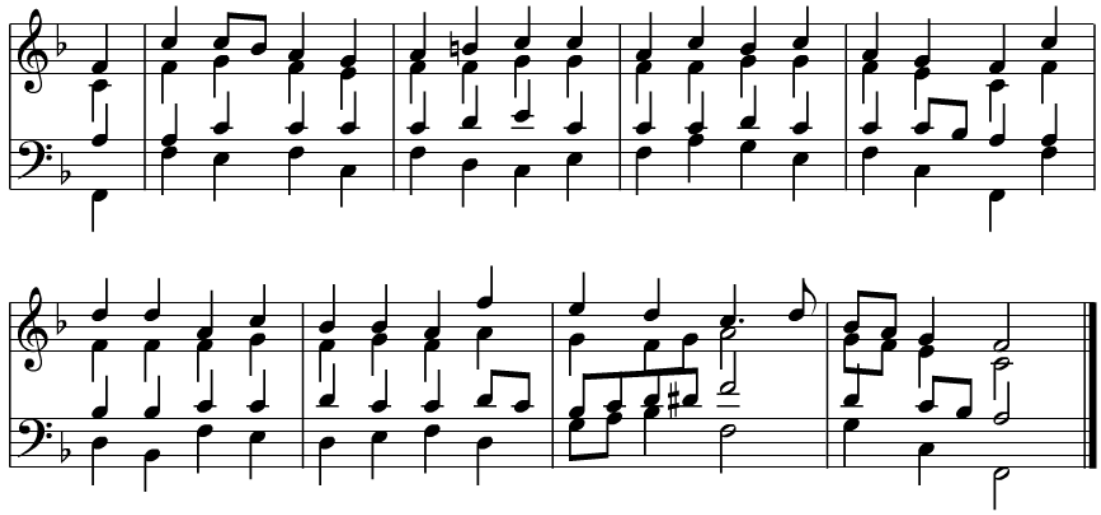


(a) Allan の結果 [9]

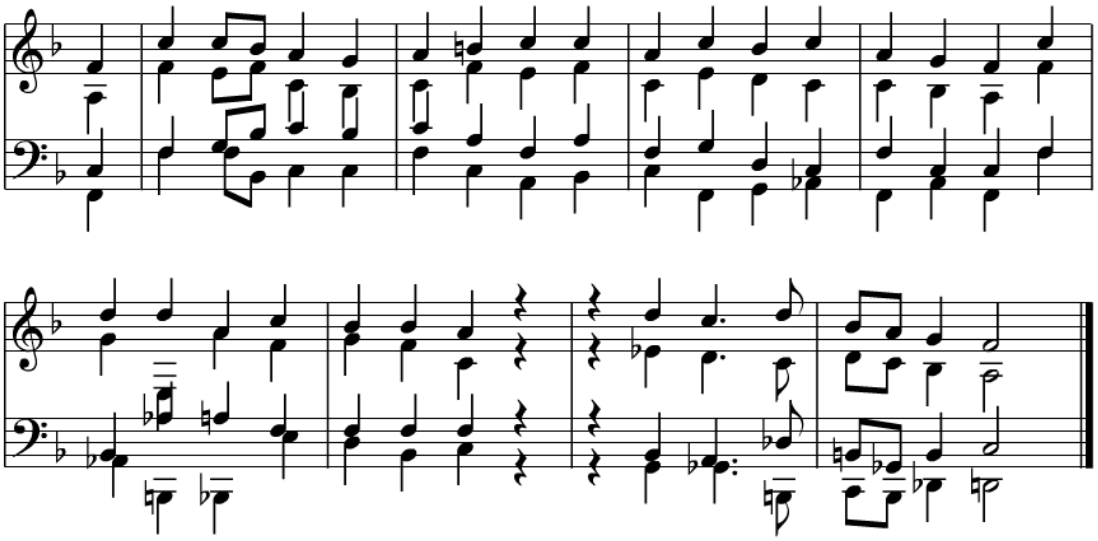


(b) 本研究の結果

図 6.1: Allan らの研究と本研究との比較 1

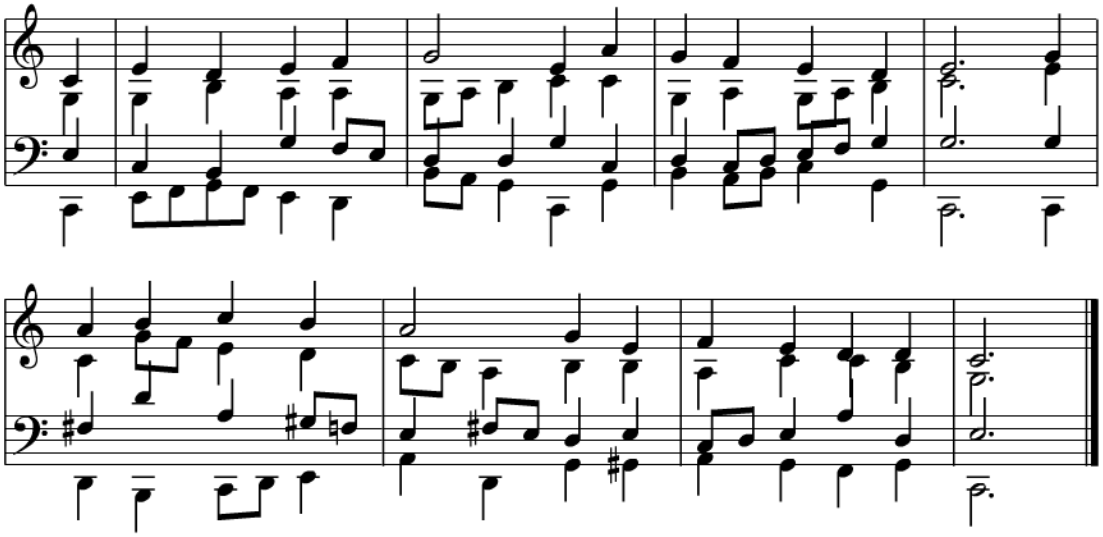


(a) Allan らの結果 [9]

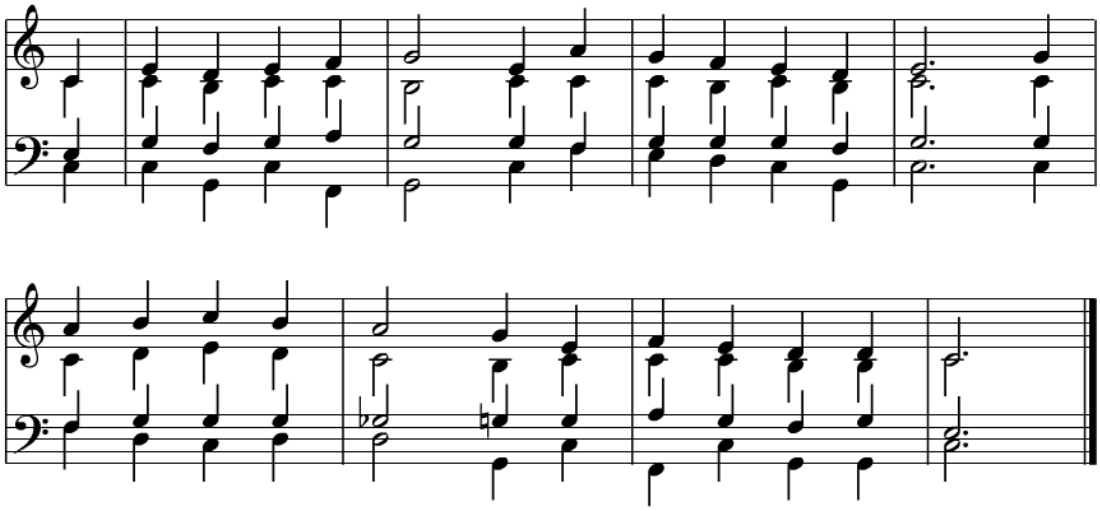


(b) 本研究の結果

図 6.2: Allan らの研究と本研究との比較 2

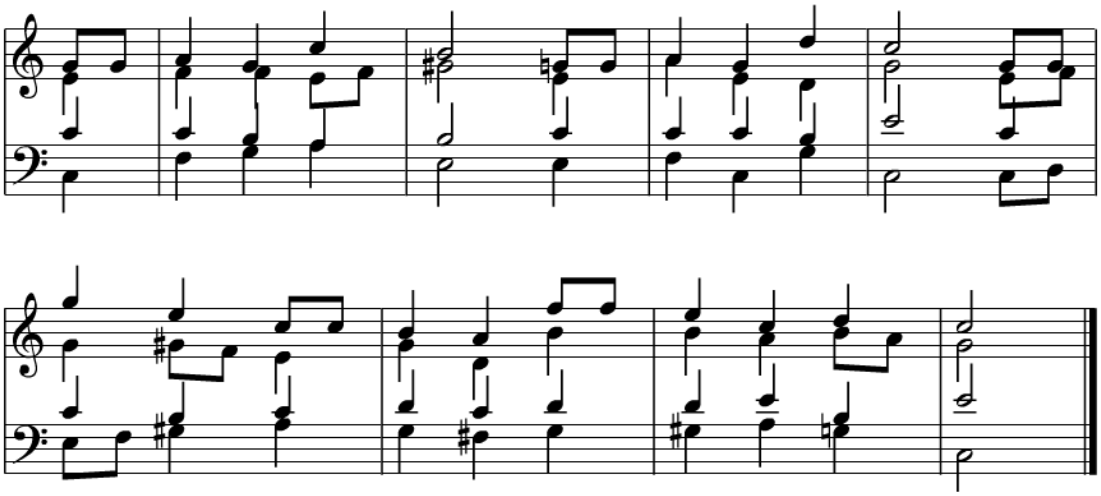


(a) Hild らの結果 [8]

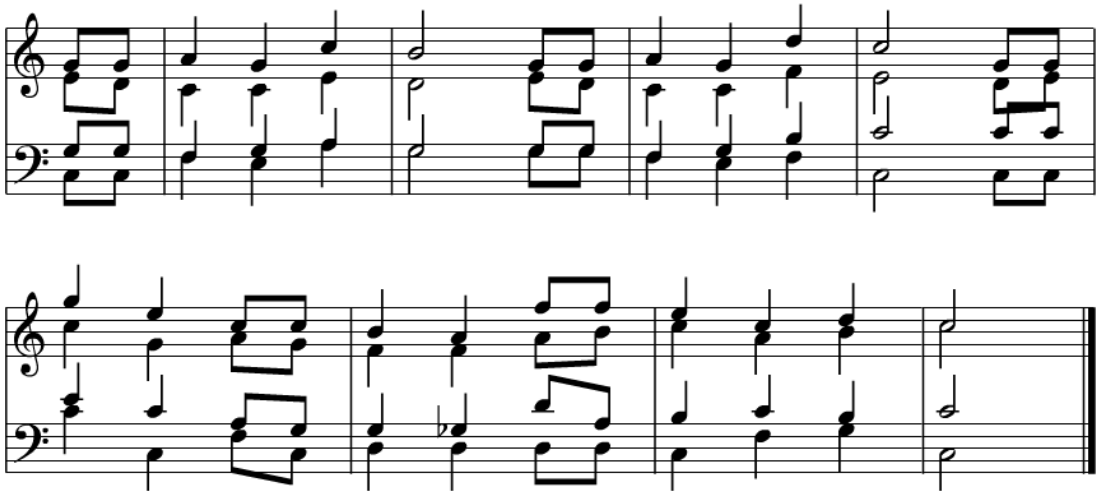


(b) 本研究の結果

図 6.3: Hild らの研究と本研究との比較 1

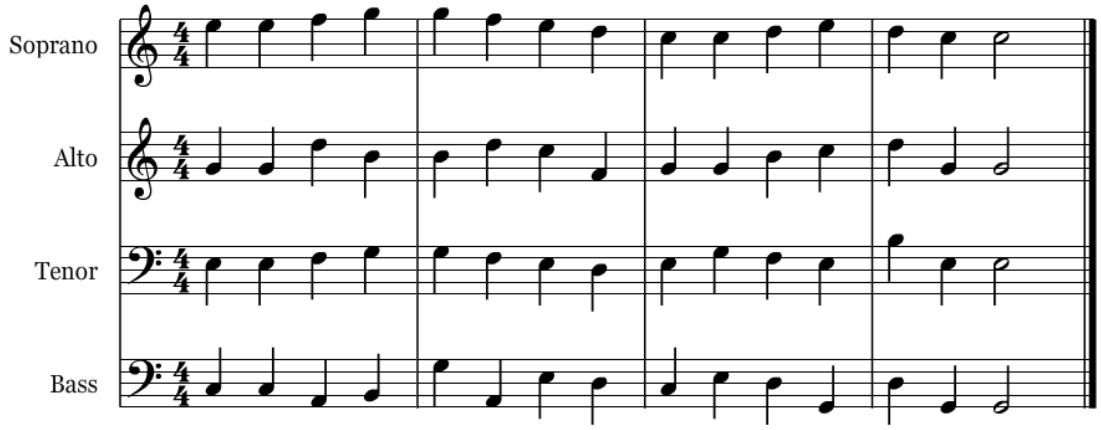


(a) Hild らの結果 [8](彼らの楽譜を適切な表記に修正した。)

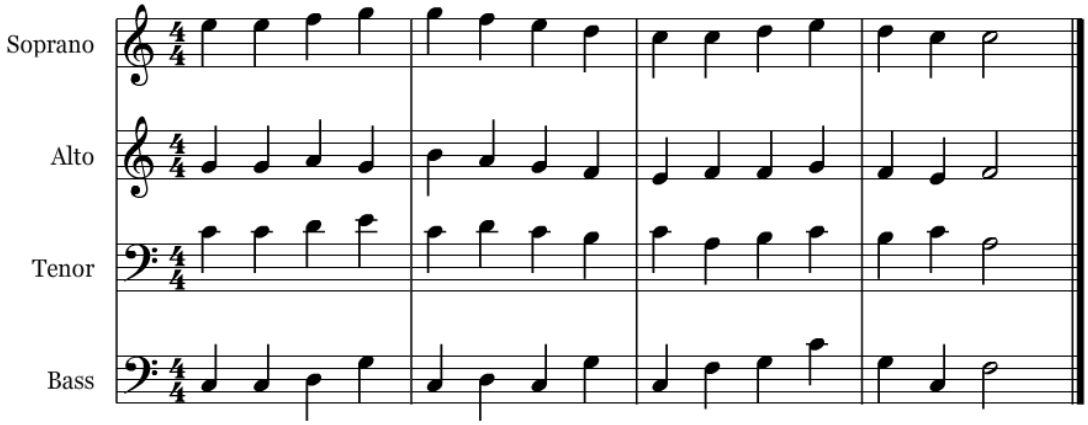


(b) 本研究の結果

図 6.4: Hild らの研究と本研究との比較 2



(a) Yi らの結果 [10]



(b) 本研究の結果

図 6.5: Yi らの研究と本研究との比較

第7章　今後の課題

この章では、今後取り組むべき課題について述べる。

7.1　禁則の考慮

和声において、並行5度のような禁止されている様々なルールが存在する。これらのルールはよく用いられるため、これらを学習することは実際の音楽大学での和声の学習において重要な課題の一つである。また、これらのルールは和声付けシステムの開発においても重要な課題である。実際に、いくつかの研究ではこれらのルールを計算機上で実装し、質の高いルールベースシステムを達成している [4, 6, 5, 7]。ベイジアンネットワークでも、禁則を学習することは可能である。しかしながら、現在のモデルでは禁則に十分に学習することはできない。例えば、並行5度のルールを学習するためには B_i の親ノードとして S_{i-1} にアークを追加しなければならない (現在は B_{i-1} と S_i だけである)。しかし、これらのアークの追加によってより多くの学習データが必要となる。従って、和声付けで同時になる音 (同時性) と各声部の連続性の一貫性を考慮することの目標として禁則の考慮は今後の課題となってくる。

7.1.1　装飾音の挿入

本研究では、すべての声部が同じリズムだと仮定したので、装飾音の挿入については取り組んでいない。リズムを全て同じにすることによって、和声付けの際

に選ばれる候補を減らし、議論を単純化している。それ故に、同じ制約が音楽大学の和声の授業でも一般的に用いられている。しかしながら、現実的な和声付けモデルの達成に装飾音の挿入は欠かすことはできない。実際に、既存研究 [8] では装飾音を挿入するモデル作成することで実現している。我々のモデルに装飾音を導入する確実なメカニズムもまた今後の課題である。

7.1.2　構造学習

我々は、同時性と連続性を考慮するために、我々の考えに基づいてベイジアンネットワークを構築した。一般的に、限られたデータ量において自ら構造を構築することはモデルの作成に適しているが、もし十分な学習データがある場合はそのデータから構造を学習することが可能である。構造学習のいくつかの手法が知られているが [18]、今後の研究として十分な学習データのコーパスを作成後に、そのような手法を適用していく。

7.1.3　大局的な考慮

本研究のモデルの最も重要な問題の 1 つとして、局所的にしか考慮されていないことがあげられる。大局的な依存関係を考慮することは重要だが、決められた学習データではこれは困難である。ある考えとして、ルールベースとの統合や音楽的に認められたいくつかの手法からベイジアンネットワークによく知られているの知識を導入する方法がある。これも将来取り組むべき重要な課題である。

第8章　結　　論

本研究では、コードシンボルの学習に関する3つの仮説を検証するため、ベイジアンネットワークを用いて与えられたソプラノメロディに対して四声体和声を生成する実験の報告を行った。コードシンボルはベイジアンネットワークにノードとして取り入れるべきではあるが、コードシンボルには曖昧性があるため、ベイジアンネットワークに適切にコードシンボルに導入することは、困難である。本論文では、コードシンボルを含まないベイジアンネットワークモデルを提案し、コードシンボルを含むモデルよりも連続性の高い結果がコードシンボルを含まないモデルによって示された。同時性においては、大きな差は見られなかった。

各章で述べた内容の要旨は次の通りである。

第1章では、本研究の背景と目的について明らかにした。四声体和声付けにシステムについて、多くの研究が曖昧性を持つコードシンボルを用いていることを述べ、コードシンボルの有無による結果の仮説を立てた。

第2章では、ルールベースシステムを用いた研究と、機械学習を用いた研究について述べ、四声体和声付けに関する研究分野を概観した。

第3章では、本研究での和声の捉え方、及び、ベイジアンネットワークを用いたコードモデルとノンコードモデルを用いた機械学習を行う手法を提案した。

第4章では、同時性と連続性に関する客観評価と主観評価を行った。コードノードを持たないノンコードモデルの結果の方がコードノードを持つコードモデルよりも連続性の高い結果が、客観評価と主観評価において証明された。

第5章では、ノンコードモデルの構造に関して評価を行い、第6章で既存研究

との比較を行った。

第7章では、禁則の考慮やモデルの大局的な考慮、コーパスの拡張などの課題について述べ、今後の研究の発展について議論した。

今後の課題として、第7章で述べた課題を解決するためにより大きなコーパスを作成し、モデルを改良していきたい。

参考文献

[1] Hu, J, Guan, Y. & Zhou, C. (2010). A Hierarchical Approach to Simulation of the Melodic Harmonization Process. *IEEE International Conference on Intelligent Computing and Intelligent Systems*, 2, 780–784.

[2] Miura, M., Kurokawa, S., Aoi, A. & Yanagida, Y. (2004). Yielding Harmony to Given Melodies. *The 18th International Congress on Acoustics*, 3417–3420.

[3] Kawakami, T., Nakai, M., Shimodaira, H. & Sagayama, S. (2000). Hidden Markov Model Applied to Automatic Harmonization of Given Melodies. *IPSJ SIG Notes*, 99-MUS-34, 59–66.

[4] Ebcioğlu, K. (1990). An expert system for harmonizing chorales in the style of J.S. Bach. *Journal of Logic Programming*, 8, 145–185.

[5] Pachet, F. & Roy, P. (1998). Formulating constraint satisfaction problems on part-whole relations: the case of automatic harmonisation. In *Workshop at European Conference on Artificial Intelligence '98. Constraint Techniques for Artistic Applications*, 1–11.

[6] Phon-Amnuaisuk, S. & Wiggins, G. (1999). The four-part harmonisation problem: a comparison between genetic algorithms and a rule-based system. *Proceedings of the AISB '99 Symposium on Musical Creativity*, 28–34.

[7] Phon-Amnuaisuk, S., Smaill A. & Wiggins, G. (2006). Chorale harmonization: A view from a search control perspective. *Jounal of New Music Research*, 35, 279–305.

[8] Hild, H., Feulner, J. & Menzel, W. (1991). HARMONET: A neural net for harmonizing chorales in the style of J.S. Bach. *Advances in Neural Information Processing*, 4, 267–274.

[9] Allan, M. & Williams, C.K.I. (2005). Harmonising chorales by probabilistic inference. *Advances in neural Information Processing Systems*, 17, 25-32

[10] Liangrong, Y. & Judy, G. (2007). Automatic Generation of Four-part Harmony. *Conference on Uncertainty in Artificial Intelligence-Applications Workshop*, 268,

[11] Jan, B. & Brink, van, der, M. (2012). Chorale Harmonization with Weighted Finite-state Transducers. *Twenty-Third Annual Symposium of the Pattern Recognition Association of South Africa*, 95–101.

[12] Gerhard, N. (2008). *Algorithmic Composition: Paradigms of Automated Music Generation*. Springer.

[13] Racyźnski, S., Fukayama, S. & Vincent, E. (2012). Chorale Melody harmonisation with interpolated probabilistic models. *Research Report N° 8110*. Project-Team METISS.

[14] United Church of Christ in Japan Hymn committee. (1982). Hymn: Hymn the second edition Tomoni Utaou. The Board of Publications The United of Church of Christ in Japan.

[15] Band-in-a-Box. (2009). PG music Inc.

[16] Ikeuchi, Y & Shimaoka, Y. (1967). Harmonics Theory and Exercise -The Separate Volume- Enforcement of the Task. ONGAKU NO TOMO SHA CORP.

[17] Weka(Waikato Environment for Knowledge Analysis). The University of Waikato.

[18] Dimitris, M. (2003). Learning Bayesian Network Model Structure from Data. *Technical Report* (TR-CMU-CS-03-153). Department of Computer Science Carnegie Mellon University.

謝　　辞

本論文を作成するにあたり、北原鉄朗専任講師から、丁寧かつ熱心なご指導を賜りました。ここに感謝の意を表します。