

コード進行に関するモーフィングの初期検討

植村 あい子^{1,a)} 北原 鉄朗^{1,b)}

概要: ある楽曲のコード進行を変更することで楽曲の雰囲気などに変化をつけることをリハーモナイゼーションという。元のコード進行とリハーモナイゼーション後のコード進行を少数学習し、モーフィングを行うことで、多彩なりハーモナイゼーションを生成できると期待される。本研究では、Variational Auto Encoder を用いてこのモーフィングを実現する。実験の結果、内挿によって元のコード進行をベースとした新しい響きのコード進行の生成が確認できた。

Preliminary Study on Morphing of Chord Progression

AIKO UEMURA^{1,a)} TETSURO KITAHARA^{1,b)}

Abstract: Reharmonization is changing the chord progression of music to change the impression of music. To generate a variety of reharmonization, morphing between a small number of instances of reharmonization is expected to be an effective approach. In this paper, we propose a method of morphing between reharmonized chord progressions using a variable auto encoder (VAE). Experimental results show that our method generates new chord progressions based on interpolation between prepared chord progressions in the latent space obtained through the VAE.

1. はじめに

ハーモニーのニュアンスを変化させる手段の一つに、リハーモナイゼーションがある。リハーモナイゼーションとは、メロディは変更せずに元のコードを別のコードに置換し、曲の雰囲気に変化を付ける操作のことをいう。コードの進行・連結に関する概念を機能と声というが、機能と声と同様でありながら、テンションやヴォイシングといった非和音が使用されると、音楽の印象が変化することがある。従来、ジャズ分野ではアドリブにおいてテンションやヴォイシングが多用されてきた。最近では動画投稿サイトやSNSにおいて、あるジャンル風やあるアーティスト風に演奏してみたなど、アレンジの動画も多数投稿されており、ハーモニーのニュアンスを変化させるアレンジの需要が増加している。

このような背景の下、リハーモナイゼーションを自動で

行う研究が行われている [1,2]。平田らの提案したパーピープン [1] では、Deductive Object-Oriented という知識表現・推論機構を用いてジャズらしいコード進行にリハーモナイゼーションを行った。これは、和声的文脈を考慮しながら事例に基づく推論により、ユーザの意図を反映できるようにした。Yoo らはコード進行におけるテンションの変化度を表すテンションカーブ [2] を提案し、ユーザがカーブを編集することで元の音楽をリハーモナイゼーションできるようにした。

一方で、少数のリハーモナイゼーションのコード進行から中間的なコード進行を得ることで、多種多様なリハーモナイゼーションを実現することもアプローチの一つとして考えられる。このように、複数のデータの中間的なものを生成する手法はモーフィングと呼ばれ、メロディやコード、ドラムパターン、ベースラインを対象に研究が行われている [3,5,6]。平田らは音楽理論 Generative Theory of Tonal Music (GTTM) で分析の結果得られたタイムスパン木から、メロディを部分簡約し、メロディのモーフィングを可能にした [3]。また、音楽構造に依らないモーフィングでは、Roberts らは Variational Auto Encoder (VAE) [4] を用いてメロディ・ドラム・ベースのモーフィングを実現し

¹ 日本大学
Nihon University

a) uemura@chs.nihon-u.ac.jp

b) kitahara@chs.nihon-u.ac.jp

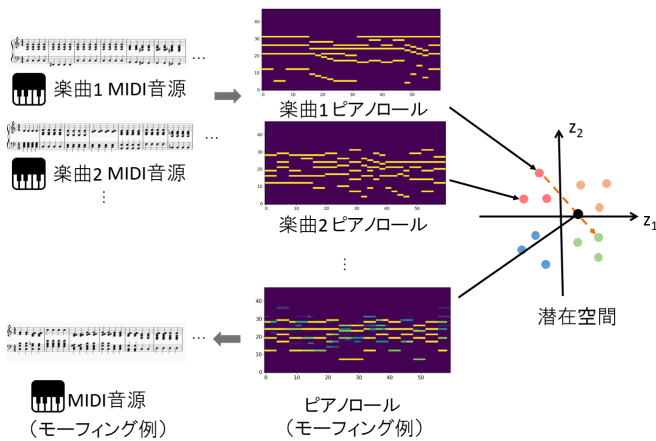


図 1 コードモーフィングの概念図

た [5]. 村田らは、メロディとコードのモーフィングを行った [6]. これらの研究では、VAE で特徴を低次元で表す潜在空間を求めることができ、連続的な補完を行えるという特長を利用している。

本研究では、Roberts ら [5] や村田ら [6] の考え方を踏襲し、異なるスタイルでリハーモナイズされたコード進行を潜在空間内にマッピングし、この空間内で内挿や外挿を行うことで、新たなコード進行を生成する。このとき、潜在空間上である 2 つのスタイルのコード進行の内挿を試みたとき、生成結果は音楽的にもこれらの中間的なスタイルを持つのが望ましい。こういった現象が実際に起こるかを実験を通じて検証する。モーフィングによりリハーモナイズデータが生成されれば、訓練データの拡張を行うデータ・オーギュメンテーション技術 [7] への応用も期待できる。

また、本稿では潜在空間上で操作できるアプリケーションについても報告する。ユーザは、潜在空間上にプロットされた元のコード進行及びリハーモナイズされたコード進行を参考に、新たにコード進行を生成し、視聴することができる。コード名や譜面でテンションやボイスが理解できないユーザでも、直感的にコードの聴こえ方や響き方から好きなコード進行を選択することができる。

2. VAE によるコードモーフィング

本稿では、VAE に基づく生成モデルにより、あるメロディに付与されたコード進行と、それを他のジャンルにリハーモナイズしたコード進行データの特徴を低次元で表す潜在空間を求める。特徴は潜在空間上で連続的に補完されることから、図 1 のようにモーフィングを行う。

VAE は、データを表現する特徴を獲得するためのニューラルネットワークであり、潜在変数に対して多変量標準正規分布を仮定している。これにより、潜在変数で表される抽象的な表現に基づいて、具体的なデータが生成されることをモデリングしている。

本研究では、コード進行の特徴を表す潜在変数 z を仮定

する。従来通り、 z は平均ベクトル $\mathbf{0}$ 、共分散行列 I の多変量標準ガウス分布に従うものとする。楽曲 \mathbf{x} はコード進行系列とし、4 分音符単位で音高オンセット系列で表現する。ここでは、C2~B5 の 4 オクターブ分を扱い、ある時刻の音高のオンセットは、ビット 0,1 で表現される。1 曲あたり、アウフタクトを除いた 15 小節分のデータを用いて、入力の特徴数は 48×60 とする。MIDI データは、読み込み時に音高オンセット区間が 1、それ以外は 0 となるように変換し、モデルを学習する。

VAE の学習では、楽曲データ \mathbf{x} 、それに対応する潜在変数を z とし、周辺尤度 $p_\theta(\mathbf{x})$ を最大にする $p_\theta(\mathbf{x}|z)$ の分布を求めるが、この計算は困難である。そこで、VAE では潜在変数の分布を生成する $q_\phi(z|\mathbf{x})$ という分布をおく。このとき、

$$\log p_\theta(\mathbf{x}) = D_{KL} [q_\phi(z|\mathbf{x})||p_\theta(z|\mathbf{x})] + \mathcal{L}(\theta, \phi, \mathbf{x}) \quad (1)$$

であり、 D_{KL} は Kullback-Leibler (KL) ダイバージェンス、 \mathcal{L} は変分下限を表す。変分下限は、

$$\begin{aligned} \mathcal{L}(\theta, \phi; \mathbf{x}) &= \log p_\theta(\mathbf{x}) - D_{KL} [q_\phi(z|\mathbf{x})||p_\theta(z|\mathbf{x})] \\ &= -D_{KL} [q_\phi(z|\mathbf{x})||p(z)] \\ &\quad + \mathbb{E}_{q_\phi(z|\mathbf{x})} [\log p_\theta(\mathbf{x}|z)] \end{aligned} \quad (2)$$

である。 θ と ϕ について $\mathcal{L}(\theta, \phi; \mathbf{x})$ を最大化することは、式 2 より $p_\theta(z|\mathbf{x})$ と $q_\phi(z|\mathbf{x})$ との KL ダイバージェンスを最小化することに相当する。

ネットワーク構造は、エンコーダとデコーダを 2 層の Multi Layer Perceptron (MLP) とし、最適化手法は Adaptive Moment Estimation (Adam) を用いて、パラメータは文献 [8] に従った。活性化関数には [4] と同様に tanh を用いた。

3. 評価実験

3.1 実験条件

学習データには、伝統曲である The Water Is Wide のリハーモナイズ譜面 [9] に基づき、人手で入力した MIDI ファイルを使用した。楽曲は、原曲 (Normal) に対して、ジャズ、ボサノバ、R&B、ファンク、フォーク、J ポップ 1、J ポップ 2 の 7 種で、すべてキーは C である。楽曲中での強弱の変化は考慮せず、すべて Velocity は 100 とした。4 分音符より長い音符は分割し、4 分音符より短い音符は省略をした。

3.2 実験結果

3.2.1 再構成の精度

学習後の 8 楽曲の潜在空間のプロットを図 2 に示す。

コード進行のプロットは $-5.03 \leq z_1 \leq -0.52$, $-5.03 \leq z_2 \leq 5.41$ に分布しているため、潜在空間上で、 $-6 \leq z_1 \leq 0$, $-6 \leq z_2 \leq 6$ で座標を動かし、生成されるコード進行

表 1 1 曲あたりのコード構成音数の平均と各コードのベース音に対して, 7th, b9th, 9th, b13th, 13th の音数を 1 曲ごとに算出した分析結果 (一部).

z_1	z_2	コード構成音数 (平均)	tritone	7th	b9th	9th	b13th	13th
-0.50	-2.00	40.9	60	60	60	60	60	60
-1.5	4.00	6.3	0	42	5	46	24	1
-2.00	0.00	4.2	7	8	0	8	10	0
-3.00	0.50	3.7	0	9	0	8	19	0
-4.50	2.20	4.5	2	26	1	19	19	2
-4.50	2.25	7.9	9	41	9	48	30	9
-6.00	6.00	6.4	0	30	6	36	21	0
Bossa Nova		4.2	2	16	4	30	4	2
Folk		4.2	2	10	0	6	14	2
Funk		5.0	0	60	0	40	4	0
J-pop1		4.4	0	26	1	15	20	2
J-pop2		4.9	2	36	0	39	14	0
R&B		4.8	6	54	6	28	4	6
Jazz		4.8	2	46	8	30	10	2
Normal		4.0	0	0	0	0	0	0

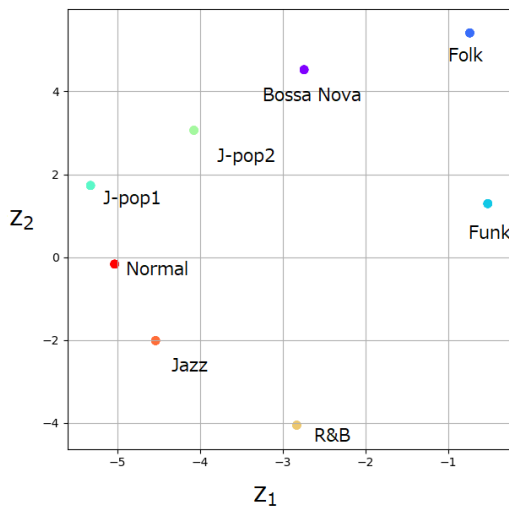


図 2 8 種のコード進行データの潜在空間

について分析を行う。デコードされたピアノロール結果を MIDI 信号に変換する際は, $[0, 1]$ で出力されるので, 閾値を用いて, 閾値以下は 0 とする処理を行った。閾値は実験的に 0.3 とした。ピアノロールを MIDI 信号に起こす際には, 100 倍にした出力を Velocity とした。

表 1 にコード分析結果の一例を示す。コードの分析では, 各コードの最低音 (ベース音) に対して, 7th, b9th, 9th, b13th, 13th の音数を 1 曲ごとに算出した。また, 1 曲あたりのコード構成音数の平均も調査した。

学習に用いたデータがピアノ譜面もしくはギター譜面であることから, コード構成音数は 3~6 がコードとして聴くのに妥当な数値だと考えられる。全体を通して, コード構成音数の平均が 8 を超えた場合は濁った響きとなりやすく, コードの構成音数が増えれば増えるほど濁った響きの進行となった。 $z_1 = -0.50, z_2 = -2.00$ の場合, 潜在空間上ではファンクに近い進行になりそうであるが, 表 1 から

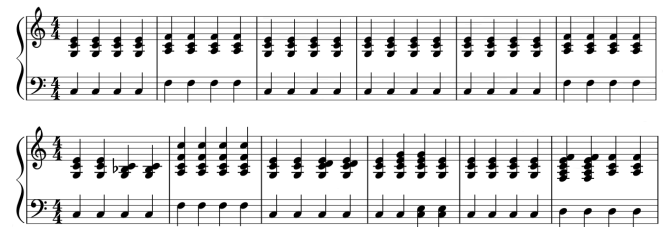


図 3 上: リハーモナイズ前のコード進行 (Normal),
 下: $z_1 = -2.00, z_2 = 0.00$ の出力結果 (冒頭 6 小節).

も分かる通り, 全小節で濁った響きのコード進行が生成された。これは, 潜在空間で表現されるデータセットからかけ離れ, 確率密度が極端に低くなってしまい, 再構成できなかったと考えられる。

3.2.2 モーフィング (内挿)

図 2 の中央部は各データから離れているが, $z_1 = -2.00, z_2 = 0.00$ では, 元のコード進行をベースに, 7th, 9th, b13th が付与された進行が生成された。図 3 に元のコード進行及び $z_1 = -2.00, z_2 = 0.00$ の冒頭 6 小節分の出力結果を示す。表 1 から 7th や 9th, b13th のテンションが付与されていることがわかる。この例ではモーフィングを行うことで, 元のコード進行をベースとした新しい響きになることを確認できた。

図 4 は J-pop1 と J-pop2 の譜面である。この座標間のコード変化については, 段階的に変化していくのではなく, コード進行のジャンルが切り替わる傾向にあった。図 5 のように $z_1 = -4.50, z_2 = 2.00, z_1 = -4.50, z_2 = 2.25$ では, 後者は z_2 を 0.25 増加させたものだが, 前者は J-pop1 寄りの進行であり, 後者はコードの構成音数が多いものの J-pop2 寄りの進行となった。このことから, この座標は各々の分布の境界に位置していると考えられる。

図 6 は $z_1 = -3.00, z_2 = 0.50$ の例である。表 1 から平

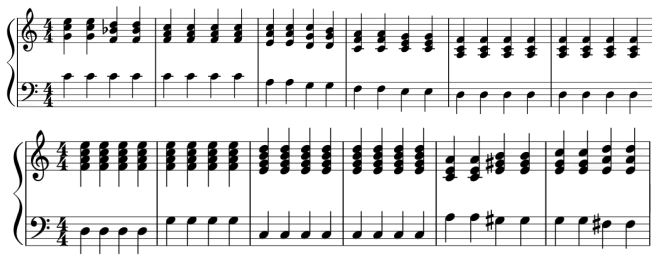


図 4 上: J-pop1, 下: J-pop2 進行の出力 (冒頭 6 小節).



図 5 上: $z_1 = -4.50, z_2 = 2.00$, 下: $z_1 = -4.50, z_2 = 2.25$ の出力結果 (冒頭 6 小節).



図 6 $z_1 = -3.00, z_2 = 0.50$ の出力結果 (冒頭 6 小節).



図 7 $z_1 = -6.00, z_2 = 6.00$ の出力結果 (冒頭 6 小節).

均のコード構成音数は 3.7 と他の出力より小さくなった。実際、三和音は楽曲中に 29 個も含まれており、これは潜在空間で表現されるデータセットから遠ざかり、確率密度が低くなったことが原因であると考えられる。

3.2.3 モーフィング (外挿)

$z_1 = -6.00, z_2 = 6.00$ で外挿を取ったときの出力結果を図 7 に示す。これは、平均のコード構成音数は 6.4 と多いものの J-pop2 寄りの進行となった。表 1 で J-pop2 と比較すると、7th や 9th は減少したが b9th や b13th が増加している。潜在空間上で外挿を取っても確率分布に従い出力されることがわかる。

4. アプリケーションの実装

ブラウザ上で起動するコードモーフィングのアプリケーション画面を図 4 に示す。これは前節で利用したモデルを利用しており、潜在空間上でユーザがクリックを行うと、その座標に基づいてコード進行の再構成が行われる。生成された MIDI は画面下で再生でき、インタラクティブに出力を確認できる。ユーザインターフェースは JavaScript を用いて、潜在空間のマッピングには Chart.js [10], MIDI 再



図 8 ブラウザ上で起動するアプリケーション。潜在空間上でユーザがクリックした座標に基づき MIDI データが出力され、再生することができる。

生には MIDI.js [11] を利用した。ユーザがクリックした潜在空間上の座標を Python に渡し、python 側でデコードと MIDI 生成を行い、JSONP のデータ受け渡しが行われる。

5. おわりに

本研究では、VAE の潜在空間における補完に着目し、リハーモナイズ前のコード進行とリハーモナイズされたコード進行を少数学習し、生成されたコード進行について分析を行った。また、潜在空間上で操作することでコードモーフィングを行うアプリケーションについても報告した。ユーザは、潜在空間上のプロットを辿ることで、元のコード進行とリハーモナイズ後のコード進行を試聴できる。また、コード名や譜面で理解できないユーザでも、直感的に好きなコードの聴こえ方や響き方から選択することが可能である。

現状では、VAE で生成したコード進行について、音楽構造としての正しさの観点からの分析は行えておらず、今後は音楽理論も踏まえた上での生成及び評価についても検討していきたい。使用したデータは少数であったので、今後は同ジャンルアレンジのコード進行を入力したときの潜在空間上での振る舞いについて調査するとともに、VAE のネットワーク構造についても検討を行っていく。

謝辞 本研究の一部は、JSPS 科研費 16K16180, 16H01744, 16KT0136, 17H00749 の助成を受け実施した。

参考文献

- [1] 平田圭二, 青柳龍也: パーピープン: ジャズコードを生成する創作支援ツール, 情報処理学会論文誌, vol. 42, no. 3, pp. 633-641 (2001).
- [2] Yoo, M-J. and Lee, I-K.: Musical Tension Curves and its Applications, In *Proc. International Computer Music Conference (ICMC)*, pp. 482-486 (2006).

- [3] Hirata, K., Tojo, S., and Hamanaka, M.: Melodic Morphing Algorithm in Formalism, In *Proc. International Conference on Mathematics and Computation in Music (MCM)*, pp. 338-341 (2011).
- [4] Kingma, D. P. and Welling, M.: Auto-encoding variational Bayes, In *Proc. International Conference on Learning Representations (ICLR)* (2014).
- [5] Roberts, A.: Hierarchical Variation Autoencoders for Music, In *Proc. Conference on Neural Information Processing Systems (NIPS) Workshop* (2017).
- [6] 村田颯, 坂東宣昭, 糸山克寿, 吉井和佳: Variational Auto Encoder を用いたメロディとコードのモーフィング, 情報処理学会全国大会, 3N-08 (2018).
- [7] Tanner, M.A. and Wong, W. H.: The calculation of posterior distributions by data augmentation, *Journal of the American Statistical Association*, vol. 82, no. 398, pp. 528-540 (1987).
- [8] Kingma, D. P. and Ba, J.: Adam: A Method for Stochastic Optimization, In *Proc. International Conference on Learning Representations (ICLR)*, pp. 1-13 (2015).
- [9] 杉山泰: リハーモナイズで磨くジャンル別コードアレンジ術 作曲&編曲に役立つ音楽理論を実践形式でマスター, リットーミュージック (2011).
- [10] Chart.js (online),
入手先 (<https://www.chartjs.org/>) (2018.05.04).
- [11] MIDI.js (online),
入手先 (<https://github.com/mudcube/MIDI.js>) (2018.05.04).