

要約

新しくギターを始める人々のために、ギター演奏のハードルを下げるシステムを提案します。ギターは幅広い音楽ジャンルで愛される楽器ですが、奏法やコード、リズムが複雑で初心者にとって難しいと感ずることがあります。私たちは、FFT（高速フーリエ変換）を使用してギターの演奏音を解析し、ユーザーが直感的に演奏できる環境を作成します。ギターの音をマイクで拾い、その周波数スペクトルを解析することで、演奏中の音の周波数成分や強度を把握します。これにより、ユーザーが意図する音を正確に判別する。

キーワード：ギター、タブ譜

1 はじめに

多くの人々が楽器演奏を始める際に抱くハードルを低くすることです。ギターは美しい音色と幅広い音楽ジャンルでの使用が可能であり、多くの音楽愛好者にとって魅力的な楽器です。しかし、ギターは奏法やコード、リズムなどの複雑さから入門者にとっては演奏が難しいと感ずられることがあります。そこで、直感的にギターを演奏できるシステムを提案することで、演奏へのアプローチをより手軽にし、初心者でも楽しさを感じながらギターを演奏できる環境を整備することを目指します。

2 従来の研究

2.1 音楽とpythonプログラムでできることの関係

音楽の生成と合成について説明する。Pythonプログラムを使用して音楽を生成し、合成することができます。音声処理ライブラリ（例: librosa）を活用して波形データを操作し、異なる楽器の音色やエフェクトを再現することができます。また、音階やコード進行の理論をプログラムに組み込んで、自動的にメロディやハーモニーを生成することも可能です。

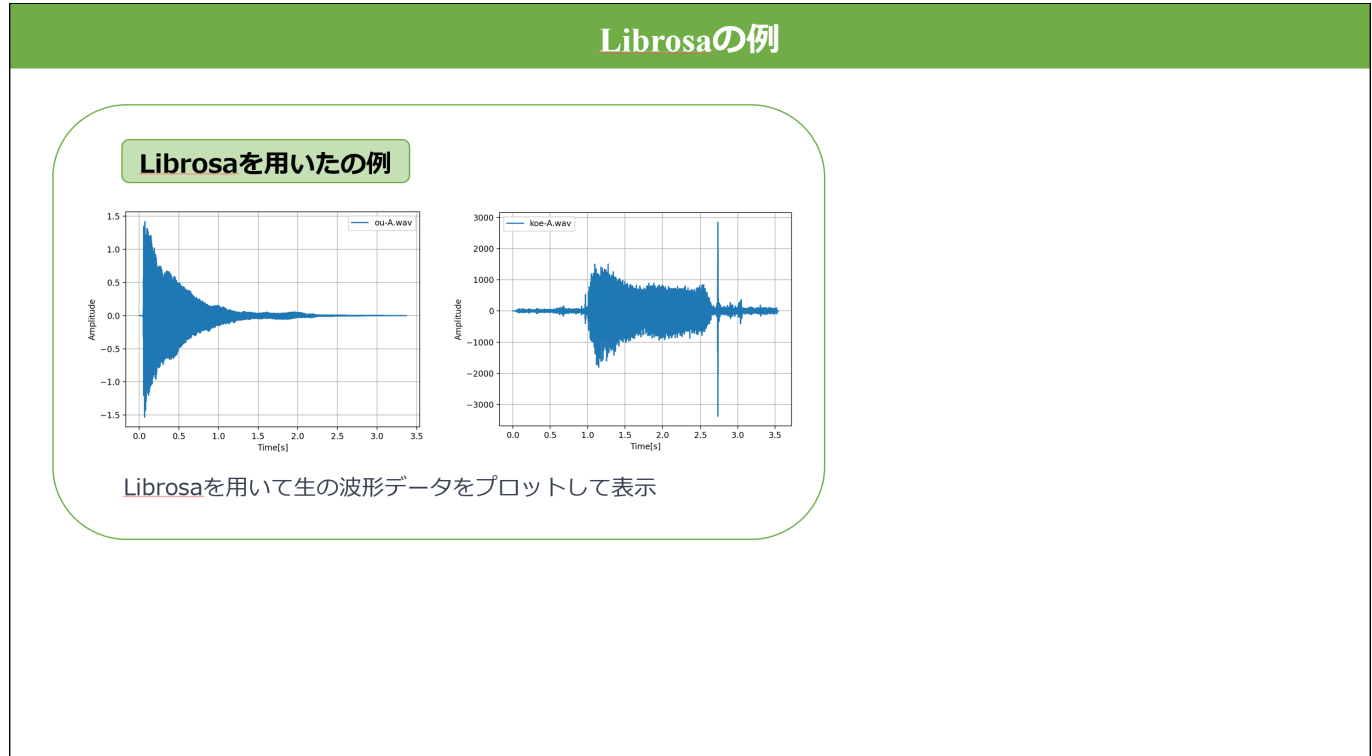


図1 librosaの例

Pythonを使用して音楽データを解析し、波形、スペクトログラム、テンポ、音高などの情報を可視化することができます。これによって、楽曲の構造や特徴を理解し、音楽理論やアレンジの向上に役立てることが可能です。

2.2 ギターに関する専門知識

まず最初に覚えておきたいことが、弦とフレット（指板に打ち込まれている金属のバー）の呼び方です。弦は細い方から1弦、2弦、3弦...と呼びます。そしてフレットは、ナット（ヘッドと指板の境目で弦を支えているパーツ）に近いものから1フレット、2フレット...と数えます。これらはギターの基礎知識です。

弦を表す6本の線の上に、押さえるフレット数を書いたものが「TAB譜」です。つまり、この図2では、5弦3フレットと4弦2フレットを押さえる、ということです。

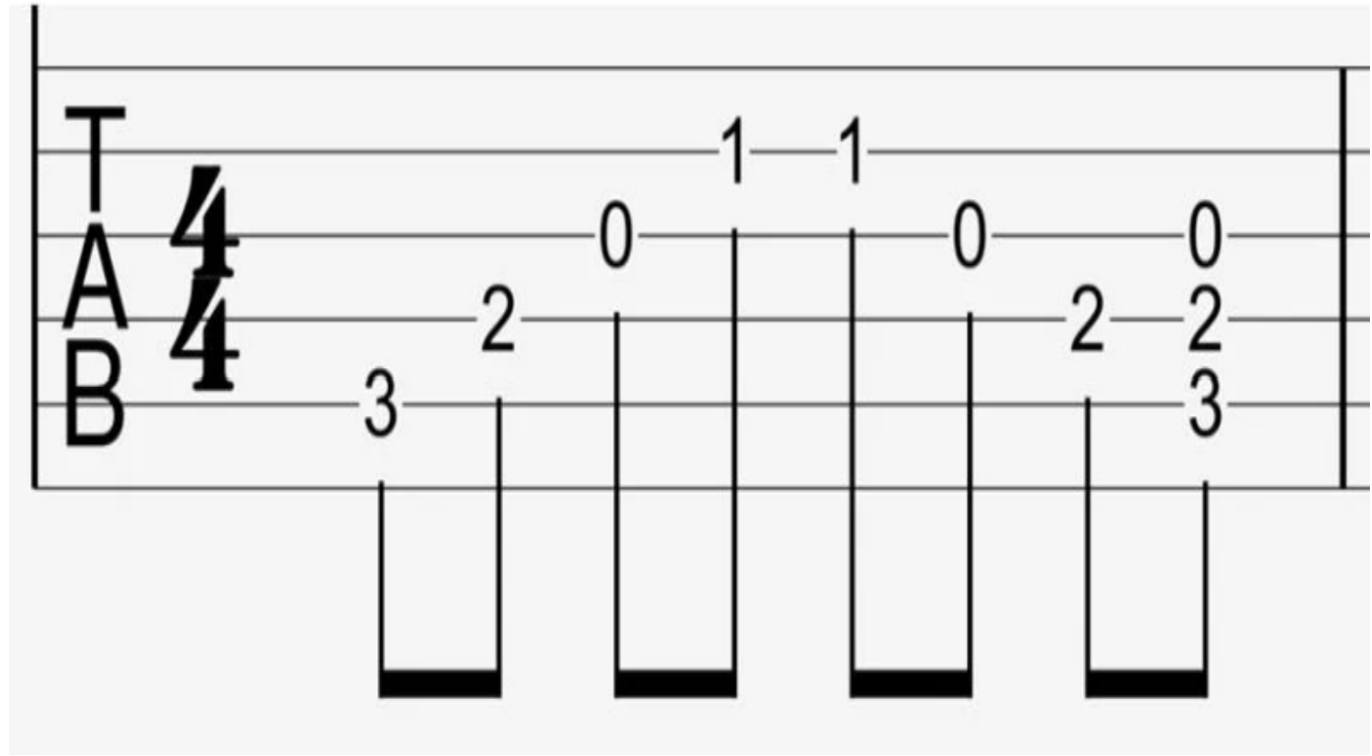


図2 TAB譜

2.3 歌からギターのタブ譜生成

GAは、多目的最適化問題を解くときに複数の目的関数及び制約条件の値を繰り返し評価する必要があり、膨大

な計算時間が必要となる問題がある。このため、並列処理により計算時間を短縮することは重要な課題となる。並列分散処理とは、複数台のコンピュータを用いて複数のCPUや、メモリを使うことで一つの計算処理を行い、性能や計算速度を向上を図ることである。本研究ではいくつかの数値実験を行い従来の並列分散モデルとの比較を通して提案するモデルの有効性の検証を行う。

3 未定

3.1 2つの楽譜の数値化による比較

レシピサイトによっては、掲載されているレシピの種類に偏りがある場合がある。先行研究でレシピサイトとして参考にしていた「ボブとアンジー」には主食が掲載されていないため、出力結果に主食が出力されていないという問題が生じていた[2]。そこで本研究では「ボブとアンジー」に加え、「EatSmart」と「おいしい健康」の2つのサイトからスクレイピングを行い、多種多様なレシピを参照し、より実用的な献立作成を提案する。

「EatSmart」は株式会社EatSmartが運営しているレシピサイトであり、主に主食の栄養素を掲載している。また、「おいしい健康」は管理栄養士が考案したレシピが掲載されており、生活習慣病にあった制限食が掲載されている。「EatSmart」からは主食を、「おいしい健康」からは制限食をスクレイピングし献立をより実用化できることを目的とする。レシピデータのスクレイピング情報を図3に示す。



図3 他レシピデータのスクレイピング情報

3.2 楽曲(ギターあり)からギター音抽出

librosaは、音楽信号の解析や処理に広く用いられるPythonのライブラリです。ギター音抽出は、音楽信号からギターの音を特定するタスクであり、librosaを利用することで効果的に行うことができます。

まず、音楽ファイルを読み込むためにlibrosaのload関数を使用します。これにより、音楽ファイルの波形データとサンプリングレートが取得されます。波形データは時間ごとの音の強さを表し、サンプリングレートは1秒あたりのサンプル数を示します。

次に、波形データからスペクトログラムを計算します。スペクトログラムは、時間と周波数にわたる信号の強度を視覚化するものであり、音楽の周波数成分の変化を捉えるのに役立ちます。その後、メルフィルタバンクを適用してメルスペクトログラムを得ます。メルフィルタバンクは、人間の耳の感じ方に近い周波数スケールを提供するため、音楽の聴覚的特性を反映します。メルスペクトログラムは、メルフィルタバンクを適用したスペクトログラムです。

最後に、ギター音を抽出します。これには、メルスペクトログラムからギター音の特徴的な周波数成分を選択することが含まれます。ギター音は特定の周波数帯域に現れるため、その帯域を選択することでギター音を抽出します。しきい値を設定して、そのしきい値を超える帯域をギター音として抽出することが一般的です。

3.3 歌(ギターなし)からギターの楽譜作成

先行研究においてはプログラムの高速化として並列分散処理を施しており、ライブラリはDaskを用いていた。DaskはPythonのライブラリの1つであるNumpyや、データ解析を支援するために、時系列データや数表を操作できるデータ構造とその演算を提供している、Numpyと同様にPythonのライブラリであるPandasを、Daskは簡単に並列・分散して処理を行うことが可能である。

しかし大量のデータを扱う場合、分散処理においてApache SparkがDaskよりも高速に処理できることが示されている[6]。そこで本研究で使用する並列分散ソフトウェアとしてApache Sparkを用いることで処理の高速化を図る。Apache Sparkは大規模データを使用する場合に使われ、分散コンピューティングエンジンの中でもっとも高速なエンジンの1つである。

4 提案手法

本研究で提案する、制限食を考慮した自動献立作成システムの流れを図4に示す。まず最初に、献立作成の最適化に必要な、レシピデータと食材価格データを、Webサイトからスクレイピングし、データベースに蓄積する。次に、ユーザーが身体情報やアレルギー情報、患っている生活習慣病を入力する。その際にアレルギーや嫌いな食品が含まれるレシピをデータベースから削除する。そして蓄積されたレシピデータ、ユーザの身体情報を入力として、摂取栄養素やカロリーなどの制約条件のもと、調理時間、調理コストの最小化を目的関数に設定した最適化問題を、制約条件を考慮した遺伝的アルゴリズムによって解く。最後に、設定した日にち分散立をユーザに出力する。さらに、最適化の工程で複数のPCを利用し、並列処理を行うことでプログラム実行時間の速度向上を図る。

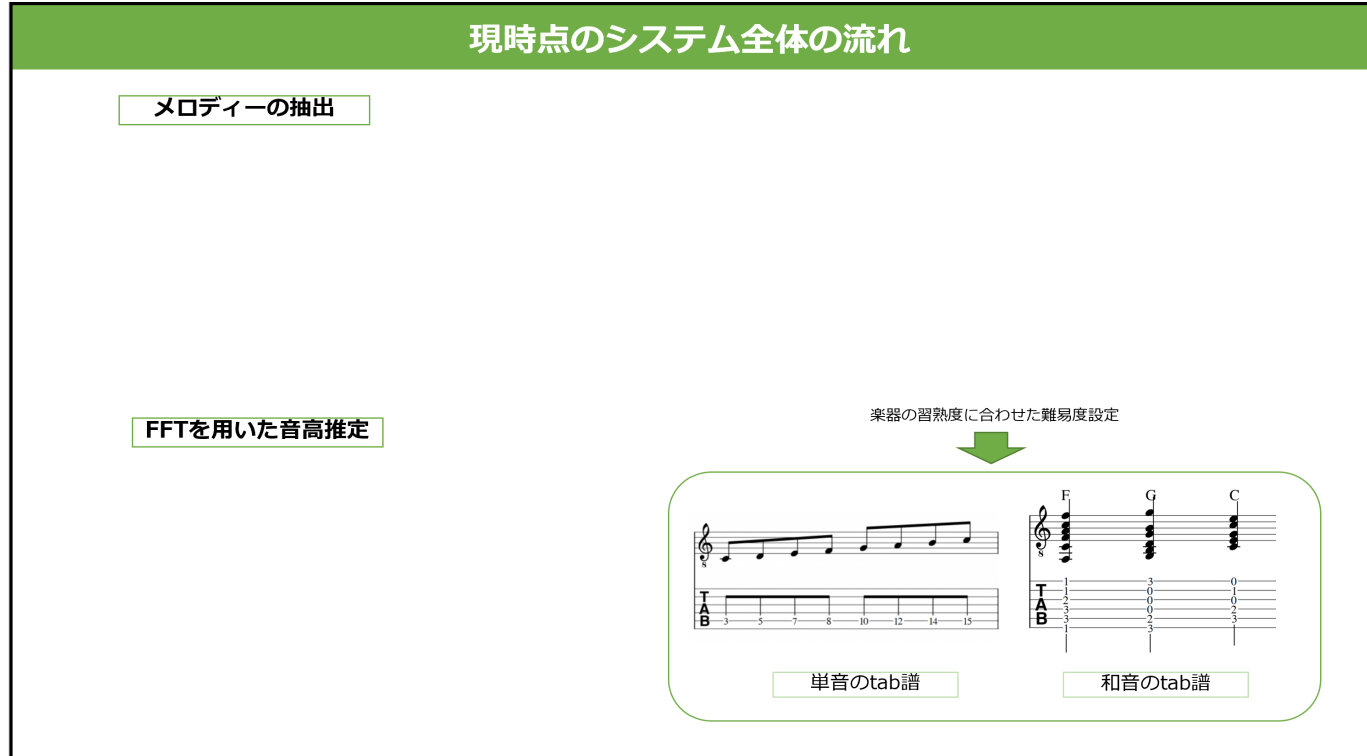


図4 自動献立作成の流れ

5 数値実験並びに考察

今回の実験ではレシピサイトとして「ボブとアンジー」「eat smart」「おいしい健康」の3つのサイトを参考にし、入力情報として年齢を21歳、身長を167cm、体重を67kgとした。また、並列分散としてDaskをもちいた。用いた制約条件と出力までにかかった時間を図5に示す。自動献立作成システムによって出力した1週間分の献立は、設定した制約条件を満たしながら最適化されていることが分かる。しかし、PCを4台用いて並列分散処理を施しても2時間30分かかってしまうのでより効率的な手法を取り入れる必要がある。



図5 実験結果

6 おわりに

今回は、複数のWebサイトのレシピ情報を参照し、レシピデータの多様化を図った。今後の方針として、制約条件を増やし、利用者によって疾患やアレルギーを考慮できるようにすることや、別の並列処理を施すことの処理の高速化などがあげられる。

参考文献

- [1] “エレキギターの基礎知識 - JOYSOUND.com”
<https://www.joysound.com/web/s/joy/gakki/faq>, 閲覧日 2023.8.25
- [2] 安藤祐斗, “Web情報を活用した自動献立作成のための多目的遺伝的アルゴリズムによる並列分散処理” 富山県立大学学位論文, 2022.
- [3] “料理レシピ ボブとアンジー 管理栄養士監修の健康ヘルシーレシピ” <https://www.bob-an.com/>, 閲覧日 2022.11.2
- [4] “小売物価統計調査による価格推移”
<https://jpmarket-conditions.com/>, 閲覧日 2022.11.2
- [5] “アレルギーを含む食品に関する表示, 消費者庁”
<https://www.caa.go.jp/policies/policy/food-labeling/food-labeling-act/pdf/food/labeling/cms101/200716/12.pdf>, 閲覧日 2022.11.2
- [6] “Koalas (PySpark) がDask よりも高速な理由 - SQL クエリ最適化など”
<https://www.databricks.com/jp/blog/2021/04/07/benchmark-koalas-pyspark-and-dask.html>, 閲覧日 2022.11.2