様　式　Ｃ－１９、Ｆ－１９－１、Ｚ－１９　（共通）

科学研究費助成事業　　研究成果報告書

令和　５ 年　 ５ 月 ３１ 日現在

機関番号：３７１１１
研究種目：基盤研究(C)（一般）
研究期間：2019～2022
課題番号：１９Ｋ１２２４４

研究課題名（和文）公開天文画像の天球への再投影による統合的データ管理とその可視化システムの開発

研究課題名（英文）Development of a Homogeneous Data Management and Visualization System for Astronomical Multi-Wavelength Open Images Enabled by a Reverse Projection Method onto the Celestial Sphere

研究代表者

江口　智士（Eguchi, Satoshi）

福岡大学・理学部・助教

研究者番号：４０６４７２０２

交付決定額（研究期間全体）：（直接経費）　　3,400,000円

研究成果の概要（和文）：宇宙望遠鏡や世界中の地上望遠鏡で得られた観測データは、「天文データ・アーカイブ」という形で一般公開されている。これら天文データ・アーカイブを世界共通のプロトコルで相互接続したネットワークを「バーチャル天文台(VO)」と呼ぶ。
VOの利便性をさらに向上させるため、本研究では分散処理フレームワークDaskを用いて画像FITSのメタデータを正しく再構築するためのソフトウェア開発を行った。その際、科学計算用ライブラリが充実しているPythonを採用し画像FITSをDaskのチャンクに分解し、各チャンクをSciPy.ndimage.map_coordinate()関数で天球に再投影した。

研究成果の学術的意義や社会的意義
本研究の主眼は「高精細でファイル・サイズが巨大な『天体写真』を元の天球座標に正しく座標変換する」ということだが、今回開発したアルゴリズムは一般化されており、変換先の座標系は球面に限定されない。従って、並列計算を用いて、高解像度で撮影された文化財の写真を元の3次元的形状に再構築する用途にも使用可能である。Dask本家でもSciPy.ndimage.map_coordinate()の移植が試みられており、今後検証作業が完了すれば、そちらへのプルリクエストも可能である。

研究成果の概要（英文）：Observational data obtained with space and ground telescopes are publicly available as astronomical data archives. The Virtual Observatory (VO) is a collection of those interoperating data archives over the VO protocols.
In our work, to enhance its convenience, we developed software to reconstruct metadata of image FITS files in VO correctly with Dask, a graph computation framework implemented in Python. A major part of our work is fractionalization of an image FITS file into multiple Dask chunks and reprojection of each chunk into the sky coordinate by utilizing SciPy.ndimage.map_coordinate().

研究分野：データベース天文学

キーワード：バーチャル天文台　座標変換　画像変換　並列処理　グラフ・コンピューティング　Dask

様　式　Ｃ－１９、Ｆ－１９－１、Ｚ－１９（共通）

１．研究開始当初の背景

宇宙望遠鏡や世界中の地上望遠鏡による観測データの多くは、「天文データ・アーカイブ」という形で一般公開される。これらアーカイブの設計や実装は、それを提供する機関に完全に委ねられている。一方、現在の天文学研究の多くは、様々な波長の観測データを組み合わせて対象を多角的に分析することで成り立っている。そのため、異なるアーカイブ同士を相互連携させる統一化された仕組みが必要であり、それが「VO (Virtual Observatory)インターフェース」である。また、VO インターフェースにより相互連携するアーカイブの集合体を「VO」と呼ぶ。

VO を実際に利用してみると、「検索結果の一覧に『存在する』として表示されたデータを実際にダウンロードして解析してみると、必要なデータが含まれていなかった」状況によく遭遇する。特に画像データの場合この原因は、一般に望遠鏡の視野が複雑な形状をしているにも関わらず、検索に使用されるメタデータには、実際の視野を内包する大幅に簡略化された図形(円・長方形・多角形)が登録されているためである。これは多くの VO 初心者に「VO は使いにくい」という印象を持たせる原因となっていた。

２．研究の目的

そこで我々は、以前開発した「VO を巡回してデータを自動収集する『VO クローラー』」と、クラウド・コンピューティングによる分散並列計算で広く用いられるフレームワーク「Hadoop」とを組み合わせ、VO 上の画像データを個別にダウンロードして「本当にデータが存在する部分」の情報を抽出し、その画像のメタデータを正しく再構築することを考えた。その際、データの「輪郭」を多角形で表現するのではなく、天球全体に同じ形・立体角の画素を一意に割り当て、その画素にオリジナルの画像データを再投影することにした。光赤外望遠鏡と電波望遠鏡、あるいは X 線望遠鏡では光学系の違いから、同じ空の領域を同じように観測しても、「『現像された』天体写真の『写真乾板』上のどの位置にその星が写るか?」が変わってしまう。我々の手法では、この「写真乾板」に対して適切な座標変換を施して「正しい天球上の位置」に引き戻すため、波長の異なる画像データを統一的に取り扱うことが可能であり、さらに天体の座標情報を予め数値化したカタログ・データも統一的に扱える。

３．研究の方法

天球を等立体角のピクセルに分割するアルゴリズムとして、HEALPix が広く使われている。画像データ中の各ピクセルを「何らかのプログラム」で HEALPix ID に変換し、**2 つの回転した座標系間で画像の補間を行うアルゴリズム**を用いれば、その結果をデータベースに登録すれば目標は達成できる。実際、Hadoop 上で動く SQL-like なデータベース・エンジン「Hive」のユーザ定義関数として、「天球座標 : (赤経, 赤緯)と検索半径」と「HEALPix ID」の相互変換機能を実装した。問題は「座標系間の画像補間アルゴリズムをどう実装するか?」であった。

天文学の標準的データ・フォーマットは「FITS」であり、特に画像データを保存した FITS を「画像 FITS」と呼ぶ。そこで、画像 FITS を補間しつつ HEALPix ID に変換する機能を実装することに焦点を絞った。歴史的に VO 関係のツールは Java で実装されてきたこと、画像 FITS を取り扱う Java ライブラリも存在していたこと、そして Hadoop 自体が Java で実装されていることから、補間機能を Hadoop 上の分散プログラムとして Java で実装することを試みた。しかしこれは頓挫した。敗因は、

- 画像 FITS の解像度がそのまま HEALPix に変換するには高すぎること
- 他のプログラミング言語(C++など)と比較して、Java はメモリ関係の制約が厳しいこと
- 既存のライブラリが分散環境下でリソース競合を起こさないか検証が必要だったことである。

そこで、すべてを Hadoop で処理することは一旦諦め、別のフレームワークで HEALPix データに変換したのち、最後にローカルのファイル・システム経由で Hive テーブルに変換する方針を検討した。例年参加している天文学関連のソフトウェアの国際研究会「Astronomical Data Analysis Software and Systems (ADASS)」で、5 年程前から話題になっているのが「グラフ・コンピューティング」である。これはクラウドのような分散システムにおいて、計算内容と必要なデータの組をフレームワークに与えると、フレームワークがそれをグラフ(「グラフ理論」の「グラフ」の意)に変換し、データ間の依存関係を解いて分散システム全体に処理を分配する、というものである。その中でも「Dask」は Python を用いたフレームワークであり、「ベラルービン天文台(旧 Large Synoptic Survey Telescope プロジェクト)」のパイプライン開発にも使用

されており、開発中のパイプラインのソースコードが GitHub に公開されている。従って、サンプルコードが豊富なため、本研究で開発する座標変換および補間プログラムを Dask で実装することにした。

４．研究成果

画像 FITS を HEALPix ID に変換するプログラムを Dask を用いない普通の Python プログラムとして実装する場合、

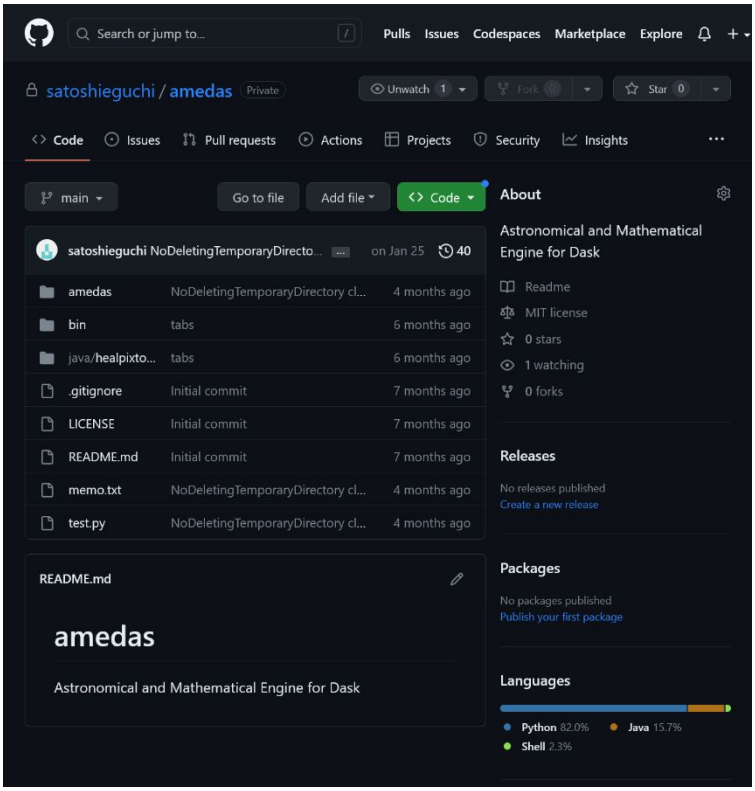
- Astropy（天文学関係の諸計算・データ入出力を行うライブラリ、HEALPix ID の計算に使用）
- SciPy.ndimage（多次元イメージをある座標系から別の座標系に内挿しながら変換するライブラリ）

を用いればよい。Dask は内部的にはデータを「チャンク」という小さな断片に分解して、NumPy や SciPy などのごく普通の Python 用科学計算ライブラリを用いて処理している。そこで本研究で開発した Dask アルゴリズムでは、以下のようにして画像 FITS を HEALPix ID に変換する：

1. 画像 FITS の画像の端 4 辺上のピクセルについて、検出器座標から(赤経，赤緯)に変換し、それをさらに 3 次元直交座標系中の半径 1 の球面上の点(x，y，z)に変換する。
2. 上記 1 のすべての点を内包する最小の球の中心座標と半径を計算する。
3. 上記 2 の球と上記 1 の共通部分を計算し(単純な幾何学)、画像 FITS の全ピクセルを**含む可能性のある** HEALPix ID を探索するための「天球上の中心点の(赤経，赤緯)と検索半径」を求める。
4. Java の HEALPix ライブラリを用いて、上記 3 の「中心点の(赤経，赤緯)と検索半径」に重なる「**『HEALPix ID の始まりの値と終わりの値』**のリスト」を計算する。
5. NumPy の memmap()機能を用いてディスク上に 1 次元の NumPy 配列を作り、上記 4 で求めた「『HEALPix ID の始まりの値と終わりの値』のリスト」を通常の 1 次元 NumPy 配列に変換する。
6. 上記 5 で求めた HEALPix ID を、元の画像 FITS の検出器座標(2 次元配列)に変換する。
7. 上記 6 の検出器座標の配列をチャンクに分解する。
8. 上記 7 の各チャンクに対し SciPy.ndimage.map_coordinate()関数を適用し、元の画像 FITS を補間しつつ HEALPix ID に変換する。

手順 4 で Java 版の HEALPix ライブラリを使用する理由は以下の通りである。上記手順 3 の段階では、元の画像 FITS とは全く重ならない HEALPix ID が大量に出力される(HEALPix ID はモートン順序であるため、画像 FITS の各画素に対応する HEALPix ID をピンポイントで計算できないため)。従って、直接 HEALPix ID のリストを生成すると実行環境(=各計算ノード)の物理メモリを大幅にオーバーする。NumPy では memmap()関数を用いることで、ファイル・システム上に配列を作成でき、しかも配列の必要な部分をファイル・システムからメモリに読み込んだり、逆にメモリからファイル・システムに書き戻したりする処理を NumPy 側で自動的に行ってくれる。ところが Astropy の HEALPix モジュールは、Cython 経由で C++で書かれた HEALPix ライブラリにアクセスしており(ソースコードを確認して判明した)、memmap()関数の恩恵を受けられない。従って、Astropy を使わずに HEALPix ID の計算を自前で行う必要がある。究極的には、本研究で作成したライブラリは分散並列環境へデプロイしたい。そこで、どんな実行環境でもコンパイル不要でライブラリをコピーするだけで機能して欲しい。すると、「Java 版の HEALPix ライブラリを活用するのが妥当」という結論を得る。

最終的には、今回開発した座標変換ライブラリの試験を研究期間内に完了させることができなかった。開発中のソースコードは GitHub で管理(図 1、現在は非公開設定)している。何とか作業を続けて、世に送り出したい。



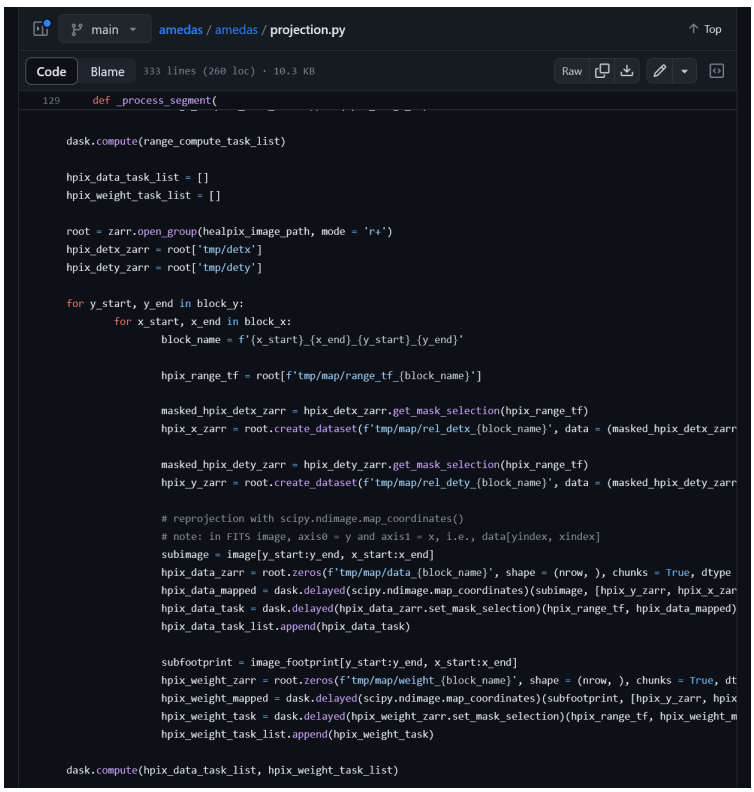


図 1：今回開発した Dask ライブラリの GitHub リポジトリ

５．主な発表論文等

〔雑誌論文〕　計3件（うち査読付論文　2件／うち国際共著　0件／うちオープンアクセス　2件）

〔学会発表〕　計4件（うち招待講演　0件／うち国際学会　3件）

〔図書〕　計0件

〔産業財産権〕

〔その他〕

―

６．研究組織

	氏名 （ローマ字氏名） （研究者番号）	所属研究機関・部局・職 （機関番号）	備考
研究分担者	白崎　裕治 (Shirasaki Yuji) (70322667)	国立天文台・天文データセンター・助教 (62616)	

７．科研費を使用して開催した国際研究集会

〔国際研究集会〕　計0件

８．本研究に関連して実施した国際共同研究の実施状況

共同研究相手国	相手方研究機関