

Pythonで動かして始める 量子化学計算

野田秀俊 著

コロナ社

まえがき

科学技術計算の分野で Python やオープンソースソフトウェアの利用が一般的になってしばらく経ちました。筆者が専門とする化学の分野では長らく商用ソフトウェアの利用が主流でしたが、潮目が変わりつつあるのを感じています。

本書は量子化学計算に関する初学者向けの入門書です。特に、プログラミング言語 Python とオープンソースライブラリ Psi4 を用いて量子化学計算の基本を学ぶことを目的としています。計算ソフトに依存しない普遍的な概念を習得できるように心がけて執筆しました。

本書のプログラムは Python 3.10 と Psi4 1.8 を用い、Mac (Apple シリコン) および Linux 上で動作確認しています。また下記の GitHub リポジトリにはすべてのサンプルコードとその実行結果に加え、Google Colab 上で実行可能な Jupyter ノートブックを用意しています。ぜひ学習に役立ててください。

<https://github.com/hidt4/python-compchem-book>

本書の原稿を査読くださり、有益なご助言を賜りました山田ひと美博士に深謝いたします。正確を期すべく内容には十分な注意を払いましたが、本書に間違いがあればすべて筆者の責任です。また、本書の出版にあたり企画段階からサポート頂きましたコロナ社の皆様に多謝申し上げます。最後に、素晴らしいソフトウェアを開発・公開してくださっている科学技術計算コミュニティの方々を中心に感謝いたします。彼らの献身的な活動がなければ本書は存在しませんでした。

それでは Python を用いて量子化学計算を始めましょう。

2024 年 1 月

野田 秀俊

目 次

第 1 章 プログラミング言語 Python

1.1	プログラミング言語 Python とはなんだろう	1
1.1.1	プログラミング言語 Python	1
1.1.2	Python の歴史	2
1.2	科学技術計算で大活躍する Python	3
1.3	第 1 章のまとめ	5
	引用・参考文献	5

第 2 章 コンピューターと化学

2.1	化学とコンピューターのいろいろな関係	6
2.1.1	計 算 化 学	7
2.1.2	ケモインフォマティクス	7
2.2	量子化学計算とはなんだろう	7
2.2.1	量子化学計算の種類について知ろう	7
2.2.2	量子化学計算用ソフトウェアについて知ろう	9
2.2.3	量子化学計算のできること	11
2.3	第 2 章のまとめ	12
	引用・参考文献	12

第 3 章 Python で量子化学計算を始めよう

3.1	Python の実行環境を構築しよう	13
-----	--------------------	----

3.1.1 conda とはなんだろう	13
3.1.2 Jupyter ノートブックとはなんだろう	14
3.2 Psi4 のセットアップを試みよう	16
3.3 水分子のエネルギー計算を試みよう	16
3.3.1 Psithon 形式で計算を実行してみよう	17
3.3.2 PsiAPI 形式で計算を実行してみよう	18
3.4 第3章のまとめ	19
引用・参考文献	20

第4章 分子をコンピューターで扱う方法について知ろう

4.1 分子を視覚的に表現する方法について知ろう	21
4.1.1 針金モデル	23
4.1.2 棒モデル	23
4.1.3 球棒モデル	23
4.1.4 空間充填モデル	23
4.2 分子を文字列として表現する方法について知ろう	24
4.2.1 XYZ 形式	24
4.2.2 MOL ファイルと SDF	25
4.2.3 Z-マトリックス	27
4.2.4 SMILES 形式	29
4.3 分子を扱うのに便利なソフトウェアについて知ろう	30
4.3.1 Avogadro	30
4.3.2 Open Babel	35
4.3.3 py3Dmol	35
4.3.4 RDKit	36
4.4 Psi4 の分子を使いこなそう	37
4.4.1 量子化学計算に必要な分子の情報を知ろう	37
4.4.2 Psi4 の Molecule オブジェクトについて知ろう	37
4.4.3 py3Dmol で分子を描画しよう	40

4.5 第4章のまとめ	41
引用・参考文献	41

第5章 量子化学計算の基礎について理解を深めよう

5.1 量子化学計算の流れについて知ろう	43
5.1.1 計算対象の分子を作成しよう	43
5.1.2 どの種類の計算を行うか決めよう	44
5.1.3 どの計算レベルで実行するか決めよう	44
5.2 いろいろな計算レベルについて知ろう	45
5.2.1 シュレーディンガー方程式とはなんだろう	45
5.2.2 ハートリー・フォック法とはなんだろう	45
5.2.3 電子相関理論とはなんだろう	47
5.2.4 密度汎関数法とはなんだろう	48
5.2.5 基底関数系とはなんだろう	49
5.3 Psi4のログファイルを見てみよう	51
5.3.1 計算手法の異なるエネルギー計算のログファイルを見てみよう	52
5.3.2 異なる基底関数系を用いたログファイルを見てみよう	54
5.4 適切な計算レベルを選択することの大切さを知ろう	56
5.4.1 塩化ナトリウムの計算	57
5.4.2 アルゴン二量体の計算	61
5.5 第5章のまとめ	63
引用・参考文献	64

第6章 分子の構造を最適化してみよう

6.1 分子の構造を最適化するとはどういうことだろうか	65
6.1.1 構造最適化のアルゴリズムについて知ろう	65
6.1.2 Psi4を用いて構造最適化をしてみよう	66
6.2 局所最適解と全体最適解の違いを知ろう	70

6.2.1	1,2-ジクロロエタンの安定構造を計算しよう	71
6.2.2	<i>N</i> -メチルアセトアミドの安定構造を計算しよう	74
6.3	構造最適化のコツについて知ろう	77
6.3.1	初期構造をできるだけ精密に作成しよう	78
6.3.2	エラーに対処しよう	86
6.4	第6章のまとめ	90
	引用・参考文献	91

第7章 振動数計算をしてみよう

7.1	振動数計算とはなんだろう	92
7.1.1	振動数計算について知ろう	92
7.1.2	Psi4で振動数計算をしてみよう	93
7.2	計算した振動数を可視化してみよう	94
7.2.1	水分子の振動モードを可視化してみよう	94
7.2.2	炭素-炭素結合の伸縮振動を比較してみよう	95
7.3	分子のIRスペクトルを求めてみよう	99
7.3.1	調和振動子近似による誤差について知ろう	99
7.3.2	スケール因子を使って実験値を予測してみよう	100
7.4	熱力学的パラメータを求めてみよう	104
7.4.1	反応エンタルピーを計算してみよう	105
7.4.2	反応の自由エネルギー変化を計算してみよう	108
7.4.3	より高精度な計算をしてみよう	111
7.5	第7章のまとめ	112
	引用・参考文献	113

第8章 分子軌道を見てみよう

8.1	量子化学計算における分子軌道とはなんだろう	114
8.2	分子軌道の描画に必要なものを知ろう	116

8.2.1	fchk ファイルとはなんだろう	117
8.2.2	cube ファイルとはなんだろう	117
8.2.3	Psi4 を用いて fchk ファイルおよび cube ファイルを作成しよう	118
8.3	分子軌道を可視化してみよう	121
8.3.1	fchk ファイルを用いて分子軌道を可視化しよう	121
8.3.2	cube ファイルを用いて分子軌道を可視化しよう	122
8.4	フロンティア軌道から分子の反応性を予測してみよう	126
8.4.1	基底関数系を変えて分子軌道を見てみよう	126
8.4.2	軌道エネルギーから反応性を予測してみよう	128
8.5	第 8 章のまとめ	131
	引用・参考文献	132

第 9 章 分子の性質を計算してみよう

9.1	Psi4 の Wavefunction オブジェクトについて知ろう	133
9.2	Psi4 における波動関数の解析法について知ろう	136
9.3	電子密度解析をしてみよう	139
9.3.1	Mulliken 電荷を計算してみよう	140
9.3.2	電子密度解析の基底関数系への依存性を知ろう	142
9.4	双極子モーメントを計算してみよう	144
9.4.1	ベンゼンの双極子モーメントを計算してみよう	144
9.4.2	ピリジンの双極子モーメントを計算してみよう	145
9.5	分子の静電ポテンシャルを可視化してみよう	146
9.5.1	Avogadro を用いて ESP を可視化してみよう	146
9.5.2	py3Dmol を用いて ESP を可視化してみよう	148
9.6	原子間の結合次数を求めてみよう	150
9.6.1	アミドの二重結合性を評価してみよう	150
9.6.2	二重結合と芳香族結合の違いを計算してみよう	152
9.7	第 9 章のまとめ	155

引用・参考文献	156
---------	-----

第 10 章 遷移状態を計算してみよう

10.1 遷移状態とはなんだろうか	157
10.2 遷移状態を求めてみよう	158
10.3 遷移状態の初期構造作成について学ぼう	159
10.3.1 反応座標軸に沿った構造のスキャン	160
10.3.2 簡単な遷移状態から始める	163
10.4 IRC 計算で反応経路を求めてみよう	165
10.5 第 10 章のまとめ	168
引用・参考文献	168

第 11 章 分子どうしの相互作用を計算してみよう

11.1 相互作用エネルギーとはなんだろう	169
11.2 基底関数重なり誤差について知ろう	170
11.2.1 基底関数重なり誤差とはなんだろう	170
11.2.2 CP 法とはなんだろう	170
11.2.3 Psi4 を用いた BSSE の補正法について知ろう	171
11.2.4 基底関数系の大きさと BSSE の関係を知ろう	173
11.3 歪曲・相互作用モデルについて知ろう	175
11.4 SAPT 計算で相互作用の内訳を調べてみよう	178
11.4.1 分子間相互作用を形成する分子間力について知ろう	179
11.4.2 SAPT 計算とはなんだろう	180
11.4.3 分子間相互作用を SAPT 計算で調べてみよう	181
11.5 第 11 章のまとめ	192
引用・参考文献	192

第 12 章 励起状態の性質を計算してみよう

12.1 励起状態とはなんだろう	193
12.1.1 励起状態の基本的な考え方を知ろう	193
12.1.2 TD-DFT 法で励起状態を計算する際の注意点を知ろう	195
12.2 Psi4 で励起状態の計算をしてみよう	195
12.2.1 Psi4 で TD-DFT 計算をしてみよう	195
12.2.2 1 電子吸収スペクトルの計算をしてみよう	199
12.2.3 電子円二色性スペクトルの計算をしてみよう	202
12.3 第 12 章のまとめ	206
引用・参考文献	208

第 13 章 お わ り に

13.1 本書で扱わなかったトピック	209
13.1.1 開殻分子の SCF 計算	209
13.1.2 有効内殻ポテンシャル	209
13.1.3 溶媒モデル	210
13.2 今後の学習に向けて	210
13.2.1 量子化学計算の専門書で学ぶ	210
13.2.2 量子化学計算以外の分野を学ぶ	211
引用・参考文献	211

索引	212
----------	-----

第1章 | プログラミング言語 Python

本書は Python というプログラミング言語を用いた量子化学計算の入門書です。本章では Python について簡単に説明します。

1.1 プログラミング言語 Python とはなんだろう

1.1.1 プログラミング言語 Python

Python はさまざまな分野で用いられている汎用プログラミング言語の1つです。Python は、初めてプログラミングを学ぶ人にもわかりやすい文法を備えています。また、Python のコードは第三者にも理解しやすいという特徴を有しています。さらに、事前コンパイルなくすぐにコードを実行可能であるため、ちょっとした点でつまずきがちなプログラミング初心者優しい言語です。

コード 1.1 からコード 1.3 ではよく使われるプログラム言語である Java, C, Python で「Hello World」と出力するプログラムを比較したものです。Python の文法が直感的で読みやすいことがわかると思います。

▶ コード 1.1 Java 言語のコード

```
1 public class HelloWorld {
2     public static void main(String[] args) {
3         System.out.println("Hello World");
4     }
5 }
```

2 1. プログラミング言語 Python

▶ コード 1.2 C 言語のコード

```
1 #include <stdio.h>
2
3 int main() {
4     printf("Hello World");
5     return 0;
6 }
```

▶ コード 1.3 Python のコード

```
1 print("Hello World")
```

また、Python は「バッテリー内蔵 (battery included)」と表現されるように、さまざまな処理が行える標準ライブラリーが付属しています。ファイル操作やネットワーク処理、簡単な統計などの実用的な作業が、Python をインストールしてすぐに利用できます。

さらに、標準ライブラリーに加えて非常に多くのサードパーティーライブラリーが存在します。この成熟したエコシステムが Python をいろいろな用途において便利なものにしていきます。特にデータ分析や機械学習などの分野において、Python は圧倒的な存在感を示しています。Python エコシステムの中でも科学技術計算関連のものについては、1.2 節でさらに説明していきます。このように Python は多分野での応用性と習得のしやすさから、近年非常に人気が高まっているプログラミング言語です。

1.1.2 Python の歴史

Python はヴァンロッサム (G. van Rossum) により 1991 年に初期バージョンが公開されました。英語で「ニシキヘビ」を意味する Python ですが、プログラミング言語 Python は「空飛ぶモンティ・パイソン (Monty Python's Flying Circus)」という番組名に由来します^{1)†}。モンティ・パイソンは英国のコ

[†] 肩付き番号は各章末の引用・参考文献を示します。

メディーグループで、この番組は1970年代前半に放送されていました。ヴァンロッサムは新しいプログラミング言語の名前として、短く、独特で、少し神秘的な名前を欲していました。ちょうど開発中に同番組の台本を読んでいたこともあり、「Python」と名付けることにしたようです。そのため、Python関係の書籍ではヘビの絵を表紙としたものが多く出版されるようになりました。

現在使われているPythonのバージョンには大きく分けて2系と3系が存在します。Python2系は2020年1月1日をもってサポート終了となりました。これ以降にバグが見つかって修正されませんので、これからPythonを学ぶ人がPython2系を使う理由はありません。皆さんもPython3系を使って勉強を始めていきましょう。本書でもPython3系を使っています。

それでも、2010年7月にリリースされたPython 2.7²⁾が10年近くにわたり現役を続けてきたのには理由がありました。2008年12月に最初のバージョンがリリースされたPython3系³⁾では、Python2系に対する後方互換性がありませんでした。つまり、Python2系で動いていたプログラムをPython3系で動かすにはプログラムを書き換える必要があったのです。

当初、Python2系の引退は2015年に予定されていました⁴⁾。しかし、2014年になってもPython3系への移行が進んでいなかったことから、2020年へと引退時期が延長されることになりました。いまでも開発が継続されている有名ライブラリーはほぼすべてがPython3系に対応しています。しかし、メンテナンスがされていない個人開発のライブラリーなどを使いたい場合には、Python3系に書き換える必要があるかもしれません。

1.2 科学技術計算で大活躍する Python

科学技術計算では対象とする自然現象を適切な数理モデルで表現し、正しいアルゴリズムに基づいた数値計算によってそれらの振る舞いを予測します。例えば、地震の予測や天気予報などの自然科学分野や自動車やロケットの設計といった工学分野が具体例としてあげられます。少し前まで科学技術計算の分野

では、C/C++ や Fortran, MATLAB などを用いるのが主流でした。もちろん、これらのプログラミング言語はいまでも使われていますが、科学技術計算における Python の利用は近年急速に増えています。

Python における科学技術計算の中心となるのが **SciPy エコシステム**です⁵⁾。SciPy エコシステムは、行列計算などの高速な数値計算を可能とする NumPy, フーリエ変換や数値積分など科学技術計算に必要なさまざまなアルゴリズムを提供する SciPy ライブラリー, そして高品質なグラフ描画ライブラリーである Matplotlib を基盤としています (図 1.1)。これら基盤ライブラリーに加えて、機械学習のための scikit-learn, 高度な統計分析のための Statsmodels, データ分析のための pandas, 数式処理のための SymPy, など数多くのライブラリーが Python を用いた科学技術計算を支えています。

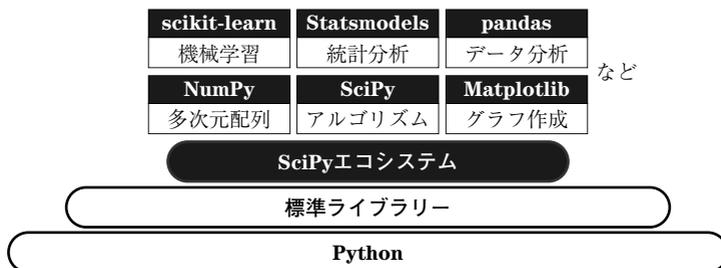


図 1.1 SciPy エコシステム

汎用性の高い科学技術計算ライブラリーに加え、物理学や化学, 生物学などの学問分野に, より特化した専門的なライブラリーも開発されてきています。これらライブラリー利用者の多くは学生や研究者であり, 必ずしもプログラミングに熟達した人ではありません。しかし, 1.1 節で述べたように Python は学習しやすいプログラミング言語であり, プログラマーでなくても Python エコシステムの恩恵を容易に享受できます。本書でも化学, 特に量子化学計算用のライブラリーである Psi4 を用いて計算を行っていきます。Python の基本的な文法知識を前提としていますが, 本書を読み進めるのに Python に熟達している必要はまったくありません。

1.3 第1章のまとめ

本章では Python とはなにかについて学びました。Python がプログラミング初心者でも学びやすい特徴を有していることをまず紹介し、Python 誕生の話から現在用いられている Python のバージョンが Python3 系であることを説明しました。さらに科学技術計算における Python の強みとして SciPy エコシステムを紹介し、その周辺にさまざまな学問分野に特化した専門的なライブラリーが整備されつつある現状を示しました。

本書ではこれから Python を用いて化学、特に量子化学計算を扱っていきます。実際に Python で量子化学計算を行うに先立ち、第2章ではコンピューターで化学を扱う方法を整理します。また、量子化学計算でできることについても説明し、本書で学んでいく内容を概観します。

引用・参考文献

- 1) Python@—the language of today and tomorrow : <https://pythoninstitute.org/about-python> (2024年1月現在)
- 2) Python 2.7.0 Release : <https://www.python.org/download/releases/2.7/> (2024年1月現在)
- 3) Python 3.0 Release : <https://www.python.org/download/releases/3.0/> (2024年1月現在)
- 4) Sunsetting Python 2 : <https://www.python.org/doc/sunset-python-2/> (2024年1月現在)
- 5) Scientific computing tools for Python : <https://svn.scipy.org/about.html> (2024年1月現在)

索 引

【あ】	シュレーディンガー方程式	45	電子配置	194
アイソデミック反応	初期分子軌道	46	電子密度解析	139
安定構造	振動数計算	92	【と】	
【か】	【す】		トリプルゼータ基底	50
ガウス型軌道	スケール因子	99	【な】	
化学情報学	スピン多重度	37	内部座標表現	27
仮想環境	スプリットバレンス基底	50	【は】	
【き】	スレーター型軌道	49	パッケージマネージャ	14
基底関数	【せ】		ハートリー・フォック近似	46
基底関数重なり誤差	制限開殻法	209	針金モデル	23
基底関数系	制限法	209	汎関数	48
求核置換反応	静電ポテンシャル	146	半経験的分子軌道法	8
球棒モデル	静電力	179	反応エンタルピー	105
局所最適解	制約付き最適化	82	反応機構	157
【く】	遷移状態	157	【ひ】	
空間充填モデル	線形表記法	29	非経験的分子軌道法	8
【け】	全体最適解	70	非制限法	209
計算化学	【そ】		【ふ】	
結合クラスター法	双極子モーメント	144	フォーマット済みチェックポ イントファイル	117
ケモインフォーマティクス	素反応	157	フロンティア軌道	126
原子価軌道	【た】		分極関数	50
原始 GTO	ダブルゼータ基底	50	分散関数	51
【こ】	【ち】		分散力	179
交換・相関汎関数	チェックポイントファイル	117	分散力補正	49
交換反発力	中間体	157	分子軌道	114
構造最適化計算	長距離補正	49	分子軌道法	8, 45
固有反応座標	超分子法	169	【へ・ほ】	
【さ】	調和振動子近似	99	ヘッセ行列	89
最小エネルギー経路	【て】		棒モデル	23
最小基底系	ディールス・アルダー反応	128	【み】	
【し】	電荷移動力	179	密度汎関数法	8
自己無撞着場	電子相関	47	ミネソタ汎関数	49
	電子相関エネルギー	47		
	電子相関理論	48		

<p>【め】</p> <p>メラー・プレセット法 48</p> <p>【ゆ】</p> <p>誘起力 179 有効内殻ポテンシャル 210</p>	<p>【り】</p> <p>量子化学計算 7, 45</p> <p>【れ】</p> <p>励起エネルギー 194 励起状態 193</p>	<p>連続誘電体モデル 210</p> <p>【ろ】</p> <p>ロンドン分散力 179</p> <p>【わ】</p> <p>歪曲・相互作用モデル 175</p>
◇		
<p>【A】</p> <p>ab initio 分子軌道法 8 Avogadro 30</p> <p>【B】</p> <p>BSSE 170</p> <p>【C】</p> <p>CCSD(T) 法 48 chk ファイル 117 CIS 194 conda 13 CPK モデル 23 CP 法 170 cube ファイル 117</p> <p>【D】</p> <p>DFT 8 Diels-Alder 反応 128 DZ 基底 50</p> <p>【E】</p> <p>ECP 210</p> <p>【F】</p> <p>fchk ファイル 117</p> <p>【G】</p> <p>GAMESS 9 Gaussian 9 Gaussian cube ファイル 117 Google Colaboratory 15 Grimme の補正法 49</p>	<p>GTO 49</p> <p>【H】</p> <p>HF 方程式 46 HOMO 126</p> <p>【I】</p> <p>IRC 165</p> <p>【J】</p> <p>Jagaur 10 Jupyter ノートブック 15</p> <p>【L】</p> <p>LCAO 49 Löwdin 電子密度解析 139 LUMO 126</p> <p>【M】</p> <p>Mayer 結合次数 150 MBIS 法 139 MOL ファイル 25 MP2 法 48 MP4 法 48 MP 法 48 Mulliken 電子密度解析 139</p> <p>【O】</p> <p>Open Babel 35</p> <p>【P】</p> <p>PCM 210 PGTO 50 Psi4 11</p>	<p>PsiAPI 形式 18 Psithon 形式 17 py3Dmol 35</p> <p>【R】</p> <p>RDKit 36</p> <p>【S】</p> <p>SAPT 180 SCF 46 SCF 計算 46 SciPy エコシステム 4 SDF 26 SMILES 記法 29 S_N2 反応 160 Spartan 10 STO 49</p> <p>【T】</p> <p>TD-DFT 法 194 TD-SCF 194 TZ 基底 50</p> <p>【w】</p> <p>Wiberg 結合次数 150</p> <p>【X】</p> <p>xtb 10 XYZ ファイル 24 XYZ 形式 24</p> <p>【Z】</p> <p>Z-マトリックス 27</p>

— 著者略歴 —

公益財団法人微生物化学研究会主席研究員

2006年 東京大学薬学部卒業, 2008年 同大大学院薬学系研究科修士課程修了, 2015年 ETH Zürich 博士課程修了 (Dr. sc. 取得)。日本学術振興会特別研究員 (PD) などを経て 2023年より現職。

Python で動かして始める量子化学計算

Beginning Computational Chemistry with Python

© Hidetoshi Noda 2024

2024年3月27日 初版第1刷発行

★

検印省略

著者 野田秀俊
発行者 株式会社 コロナ社
代表者 牛来真也
印刷所 壮光舎印刷株式会社
製本所 株式会社 グリーン

112-0011 東京都文京区千石 4-46-10

発行所 株式会社 コロナ社

CORONA PUBLISHING CO., LTD.

Tokyo Japan

振替00140-8-14844・電話(03)3941-3131(代)

ホームページ <https://www.coronasha.co.jp>

ISBN 978-4-339-06668-5 C3043 Printed in Japan

(田中)



JCCOPY < 出版者著作権管理機構 委託出版物 >

本書の無断複製は著作権法上での例外を除き禁じられています。複製される場合は、そのつど事前に、出版者著作権管理機構（電話 03-5244-5088, FAX 03-5244-5089, e-mail: info@jcopy.or.jp）の許諾を得てください。

本書のコピー、スキャン、デジタル化等の無断複製・転載は著作権法上での例外を除き禁じられています。購入者以外の第三者による本書の電子データ化及び電子書籍化は、いかなる場合も認めていません。落丁・乱丁はお取替えいたします。