目次

第1章

科学技術計算のための Python への入門

Getting Started with Scientific Python

Python は、ここ数年主流となってきている。いまや、Python は工学およびコンピュータサイエンスにおける多くの学部カリキュラムの一部となっている。優れた書籍および対話型のオンラインチュートリアルを見つけることは簡単である。特に、Python は、Django や CherryPy などのフレームワークを用いたウェブプログラミングの分野で十分な地位を確立しており、これらはアクセス数の多い多くのサイトの基盤となっているプラットフォームである。

ウェブプログラミングを越えて、線形代数から機械学習の可視化にいたる多くの科学分野全体にわたって、サードパーティによる拡張機能のリストが今も増え続けている。これらのアプリケーションにおいて、Python はメソッドの交換や、Fortran や C で書かれる典型的なコアルーチンに対してのデータを引き渡しを簡単にするための、いわばグルー（糊付け）としての役割を果たすソフトウェアである。科学技術計算のための Python は、ほぼ 20 年にわたり、産官学で基本的なものであり続けている。たとえば、米航空宇宙局（NASA）のジェット推進研究所は、宇宙船の軌道の計画と可視化のために Fortran/C++ ライブラリのインタフェースとして科学技術計算のための Python を使用している。また、ローレンス・リバモア国立研究所は、いくつかの定型テキスト処理や、ビッグデータの先進的な可視化（VISIT など[1]）など、多種多様な計算タスクに科学技術計算のための Python を利用している。シェルリサーチ、ボーイング、インダストリアル・ライト & マジック、ソニー・エンタテインメント、および P&G は、日々のデータ処理や分析に科学技術計算のための Python を使用している。Python は、このように十分に確立され、様々な分野で拡張し続けている。

Python は正式なソフトウェア開発のトレーニングを受けていない科学者やエンジニア向きの言語である。Python は、何の障害もなく、プロトタイプや設計、シミュレーション、およびテストを行うために利用される。これは Python が、本質的な容易さや、反復型開発サイクル、既存コードの相互運用性、信頼性の高いオープンソースコードの巨大な基盤へのアクセス、および階層的に区画化された設計理念を提供するためである。Python の生産性は、ユーザのワークフロー（実行時間とプログラミング時間など）に強く影響されることが知られている[2]。このため、

J. Unpingco, *Python for Probability, Statistics, and Machine Learning*