

# Pythonによる物理シミュレーション教材の作成

米村敦<sup>1</sup>・上田学<sup>2</sup>・廣田千明<sup>3</sup>・伊藤桂一<sup>1</sup>

秋田高専電気情報工学科<sup>1</sup>、秋田高専自然科学系<sup>2</sup>、秋田県立大学<sup>3</sup>

## 本研究の目的

近年、iPadやタブレットPCを活用した学習環境が普及してきている。これらの学習教材では、アプリやソフトウェアを用いてノートやテキストと同様の教材として扱うことができるが、ノートやテキストでは出来ないことも実現できる。その1つにプログラミングがある。

本研究は、Pythonを用いた物理シミュレーション教材の開発を行い、学習者が物理現象を理解することと、シミュレータを通してプログラミングの知識を深めることの両者を支援することを目的とする。

## Google Colabの概要

Google Colabとは、Google社が提供している機械学習の教育及び研究用の開発環境である。Googleアカウントがあれば誰でも無料で使うことができ、ブラウザ上で動作するため使用するまでの手順が少ない。また、使用機材の性能によらないため、PCのみならずスマートフォンでも実行が可能である。使用言語はPythonを用い、Jupyter Notebookと似たインターフェースを持つ。

## 教材内容

本研究で作成した教材は「等速・透加速直線運動」、「鉛直投げ上げ」、「斜方投射」、「衝突運動」、「自由落下（反発係数を考慮）」の5つである。これらはすべてSciPyライブラリのodeintという微分を行う関数を用いる。初期値として初速度と初期位置を与えると、任意の時刻での変位と速度の値を得られる。これをmatplotlibの動的グラフを用いて描画することでシミュレーションを行っている。おおまかな構造を以下の図1に示す。



図1 シミュレーションの流れ

作成した物理シミュレーション教材は、物理を学習する上でプログラミングによるシミュレータにしか実現できない要素に着目している。図2の等加速度直線運動シミュレータにおける一定時間間隔での軌跡のプロット、図3の自由落下シミュレータにおける位置エネルギー・運動エネルギーの可視化などが、その代表的な例である。これらのように、プログラミングによるシミュレータによって、得られる視覚的な情報が格段に増加する。また、巻き戻しや停止なども可能であるため、状態の比較が容易にでき、教材は全て最初に作成した「等速・等加速度直線運動」のシミュレータを基に作成した。odeintやmatplotlibの描画部分の構成は大きく変わらないため、メイン実行部の変更や公式の追加によって様々な物理現象をシミュレーションできる。そのため単純な物理シミュレータではなく、利用者自身がプログラミングを学びながら物理シミュレータを作成できる教材の製作が最終的な展望である。

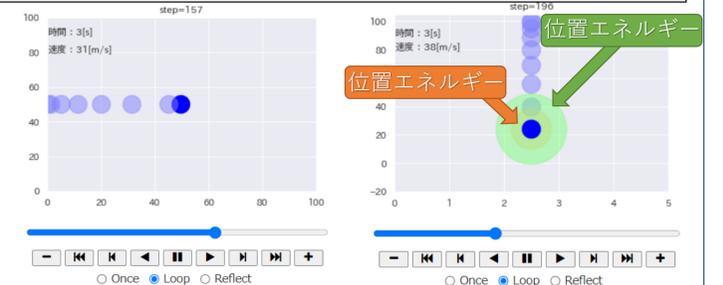


図2 等加速度直線運動 図3 自由落下

## 模擬授業による検証

物理シミュレーション教材を用いることによる効果を検証するため、12月16日（金）に本校機械科の3年生の学生9名に約50分の模擬授業を行った。模擬授業では図3の「自由落下」、図4の「衝突運動」のシミュレータを用い、反発係数の値による物理現象の変化を確かめ、そこから運動量保存則について学ぶ「帰納的授業」を行った。構成を以下に示す。

- (1)導入と反発係数の説明 (5分)
- (2)自由落下による床との衝突によるシミュレーションの練習 (5分)
- (3)2体衝突のシミュレーション (15分)
- (4)運動量保存則の説明 (5分)
- (5)二体衝突の問題演習およびシミュレーションによる確認 (15分)
- (6)授業のまとめとアンケート (5分)

学生達は物理シミュレーション教材を用いて意欲的に学習していた。(6)で行ったアンケート結果の一部を図5に示す。物理シミュレーション教材として好印象だったようである。

今後は教材の更なる改善と共に、プログラミングの学習をメインとした教材の作成を進めていく。



図4 衝突運動



図5 教材に関するアンケート結果