

物理学の学習のためのプログラマブルな シミュレータと環境の提案

東京工業大学 情報理工学院 数理・計算科学系
18B04657 木内康介
指導教員 増原英彦教授

1 はじめに

高等学校における物理学の授業において、実験は重要である。Holubova [1] は、実験室での作業は理論的な概念を検証する最も重要な方法であり、生徒は実験を通してどのような現象が起きるかを確認することができると述べている。

しかし実際は、生徒全員が実験を経験しているわけではない。力学分野において最も基本的な「運動の法則」に関する実験経験は、2014 年の調査時点で 60% にしか満たない [2]。理由としては、実験用の装置の準備や測定が難しいことや、実験を行うのに時間を要することが考えられる。

そこで実験の代替として近年利用されているのが、物理実験のシミュレータである。シミュレータを用いることで、実験と同様の学習効果を得ることができる [3]。

しかし、実際に生徒が解く問題の解答過程と既存のシミュレータの間には、確認できる情報にギャップがある。高等学校で扱う物理の問題では図 1 のように方程式の計算を要求される。一方、例えば PhET [4] は、図 2 のように速度や質量、位置のような物理量を数値で与えることで指定された運動が描画される。そのため、可視化された運動がどのような方程式によるものなのか確認することができない。

そこで本研究では、物理系を定義できるシミュレータである SimSym (Simulation with Symbols) を提案する。学習者は SimSym で系内の物体をそのパラメータとともに定義し、その物体の運動を表す方程式を立式する。SimSym は、次元の異なる物理量の和などの不正な方程式を警告するなど、学習者が正しい物理系を作成するための補助を行う。シミュレーションを実行すると、定義した物理系に基づいて数値計算がなされ、物体の運動が可視化される。[TODO: 教師役の話]

(問題) 静止している質量 m の物体に大きさ F の力をかけ続ける。 t 秒後の速さを求めよ。

(解答) 物体の加速度を a とする。運動方程式より $ma = F$
 $\therefore a = \frac{F}{m}$ 。等加速度運動の公式より $v = at = \frac{F}{m}t$
答え: $\frac{F}{m}t$

図 1 方程式の計算例

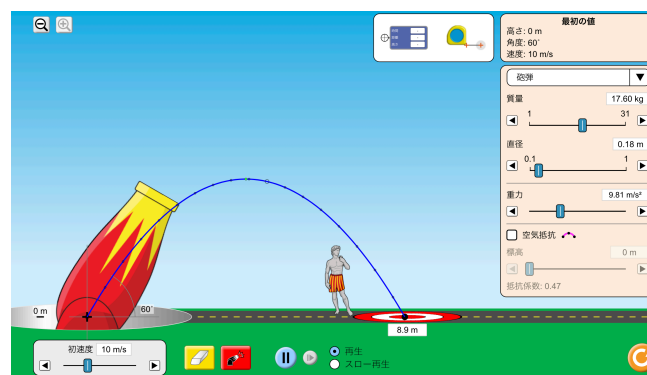


図 2 PhET のシミュレーション例

既存のシミュレータは教師役が物理系を全て定義し、学習者はパラメータを指定するだけである。[TODO: SimSym による学習効果のはなし]

なお、SimSym は大部分が未実装であるが、2 節で説明するアイデアは 3 節に記す方針で実装が可能であると考えている。

2 SimSym

学習者は、SimSym で以下の操作によって物理系を定義し、シミュレーションを実行することができる:

- 物体の作成:** 学習者が物体名を入力すると、SimSym はその物体に紐付いた物理量に対応する次元付き変数 ($m_A[M]$, $x_A[L]$ など) を生成する。
- 方程式の立式:** 学習者は、1. で生成された変数を使って方程式を立式する。この際、次元付き変数は自由に追加することができる。なお、時刻を表す変数 $t[T]$ や重力加速度 $g[L/T^2]$ などの物理定数は用意されている。また、不正な次元の方程式は警告される。
- シミュレーションの実行:** 学習者が観測したい変数を指定すると、時刻 t が変化しながら各方程式が計算され、指定した変数に対応した物理量が可視化される。なお、シミュレーションの実行のために必要な変数には初期値を与える必要がある。

[TODO: Pane の話]

物体の名前を入力し物体作成ボタンを押すと、

以下では具体例を見ていく。図 3 は、SimSym 上で斜方

投射を表現した例である。この例では y 座標に紐付けている方程式が $v_{0y}t - \frac{gt^2}{2}$ となっているため、シミュレーションを再生すると図 4 のような軌道を描く。ここで、速度の次元を持つ gt という方程式は y 座標に紐付けることができない。また、(位置) - (速度) という形になっている $v_{0y}t - gt$ という方程式を作成しようとすると、次元が一致していないため警告され、定義できない。

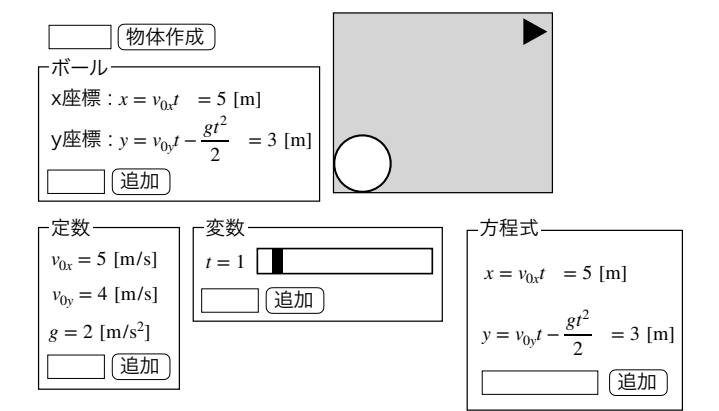


図 3 SimSym 上で斜方投射を表現した例

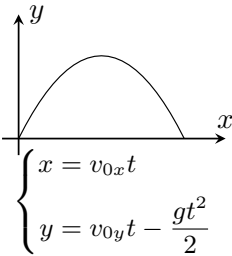


図 4 斜方投射の軌道

3 実装の方針

フロントエンドに lively.next[TODO: 引用]^{*1} を、方程式の処理と数値計算に SymPy [5] を用い、JavaScript で実装する。lively.next は、GUI アプリケーションを作成・実行するための Web プログラミング環境である。学習者が物体や方程式を定義する画面とシミュレーションを表示する画面を lively.next で作成する。SymPy は、記号計算のための Python ライブラリであり、Pyodide ^{*2} を用いることで WebAssembly に変換し、ブラウザで実行する。SimSym は、入力された物理量や方程式を SymPy オブジェクトに変換することで、数値計算を可能にする。[TODO: 具体的にどのように計算し可視化するか]

4 まとめと課題

本研究では、物理系を定義できるシミュレータ SimSym を提案した。学習者は SimSym を用いることで自身が定義した物理系の運動と現実の運動との対応を視覚的に確認

することができる。[TODO: SimSym による学習効果のほなし]。

今後の課題は、SimSym の教育効果の評価である。評価手法として、Hake [6] が導入した normalized gain を用いた実験を検討する。

参考文献

[1] Renata Holubova. The impact of experiments in physics lessons – “why, when, how often?”. *AIP Conference Proceedings*, Vol. 2152, No. 1, p. 030007, 2019.
[2] 林 社一ほか. 大学生に対する高校物理実験および放射線学習の現状調査. *物理教育*, Vol. 63, No. 3, pp. 191–196, 2015.
[3] Fadil Ajredini, et al. Real Experiments versus Phet Simulations for Better High-School Students’ Understanding of Electrostatic Charging. *European Journal Of Physics Education*, Vol. 5, No. 1, p. 59, February 2014.
[4] Katherine K. Perkins, et al. PhET: Interactive Simulations for Teaching and Learning Physics. *The Physics Teacher*, Vol. 44, No. 1, pp. 18–23, January 2006.
[5] Aaron Meurer, et al. SymPy: symbolic computing in Python. *PeerJ Computer Science*, Vol. 3, p. e103, January 2017. Publisher: PeerJ Inc.
[6] Richard Hake. Interactive-engagement versus traditional methods: A six-thousand-student survey of mechanics test data for introductory physics courses. *American Journal of Physics - AMER J PHYS*, Vol. 66, , 01 1998.

^{*1} <https://lively-next.org>
^{*2} <https://pyodide.org/en/stable/>