

SIMULAÇÕES DE MOVIMENTO BALÍSTICO COM RESISTÊNCIA DO AR

Rafael Zacarias Palierini¹, Eliane de Fátima Chinaglia²
^{1,2} Centro Universitário FEI
 rafael.palierini.contato@gmail.com; echinaglia@fei.edu.br

Resumo: Há uma necessidade na mudança do ensino de engenharia, na qual necessita de um aluno mais ativo em seus estudos. Analisar simulações e comparar com dados experimentais, são ótimas opções para tornar o estudante protagonista de seus estudos. Neste projeto, foram desenvolvidas simulações do movimento balístico de uma partícula no *Interactive Physics* e *Python* e os resultados comparados aos dados obtidos com o *Tracker* para auxiliar os alunos na compreensão conceitual do tópico.

1. Introdução

Há muito tempo a formação em engenharia é desenvolvida apenas em sua parte técnica. Porém, vem sendo trabalhado uma mudança baseada no desenvolvimento de competências do aluno. Para isso, é essencial tornar o aluno protagonista de seu próprio aprendizado, necessitando dessa forma uma substituição das salas tradicionais por novos ambientes que permitam a aplicação de aprendizagens ativas [1 - 3].

É importante que o aluno desenvolva de forma mais sistemática no laboratório os conteúdos vistos nas aulas de teoria. Portanto, é interessante o desenvolvimento de simulações que complementem e estejam diretamente interligadas às aulas teóricas.

Durante os estudos de cinemática, é esperado que o aluno aprenda a trabalhar com o movimento balístico. Porém, é comum os alunos trabalharem o movimento balístico sem atrito do ar.

No movimento balístico, o atrito do ar reduz a altura e o alcance [4]. Quanto maior for a resistência do ar, menor será a altura e o alcance [5].

Difícilmente a resistência do ar é considerada no movimento balístico, devido à dificuldade de calculá-la.

Considerando as dificuldades do aluno foi planejado o desenvolvimento de simulações que fossem capazes de auxiliar os alunos durante os estudos no movimento balístico com uma força de atrito proporcional à velocidade do objeto.

2. Metodologia

Para realizar as simulações, foi necessário primeiro revisar os conteúdos de movimento balístico sem atrito do ar e de movimento balístico com atrito do ar. Segundo Halliday [4], a equação da trajetória do projétil é da forma de uma equação de segundo grau, formando uma parábola, como é demonstrado na figura 1.

Porém, a trajetória real não é uma parábola perfeita e ela possui algumas deformidades como pode ser visto na figura 2 considerando a força de atrito proporcional à velocidade do projétil, com base nas equações desenvolvidas por Freire [5].

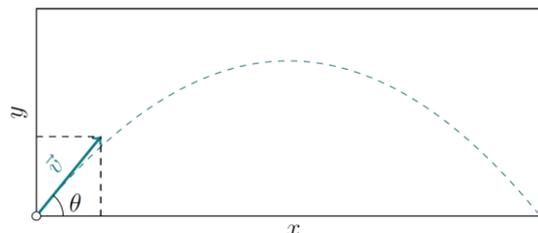


Figura 1 – Trajetória do movimento balístico sem resistência do ar

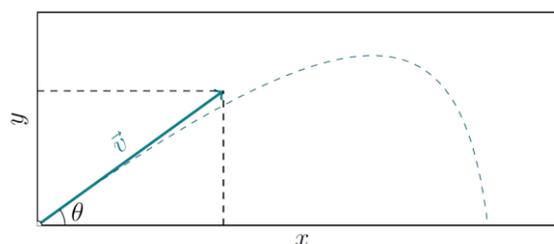


Figura 2 – Trajetória do movimento balístico com resistência do ar

Em seguida, foi construído uma simulação no *Interactive Physics* para comparar com futuros resultados experimentais que serão extraídos pelo *Tracker*. A simulação no *Interactive Physics* permite o usuário simular o lançamento de 3 esferas diferentes simultaneamente de cima de uma das mesas do laboratório de física do Centro Universitário FEI, variando o ângulo de lançamento e velocidade inicial como demonstrado na figura 3.

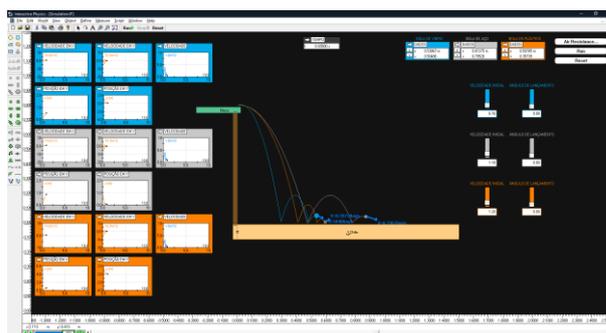


Figura 3 – Simulação construída no *Interactive Physics*

Na simulação do *Interactive Physics* é possível medir o tempo de duração da simulação, as componentes da posição $x(t)$ e $y(t)$, a magnitude da velocidade $v(t)$ e as componentes $v_x(t)$ e $v_y(t)$.

Na etapa seguinte foram realizados os lançamentos de 3 esferas, que foram capturadas por um celular (iPhone 8 Plus) e analisadas pelo *Tracker*. As 3 esferas foram lançadas de uma altura de 0,8m e um ângulo de

lançamento igual a 0° e possuem propriedades diferentes como mostra a tabela I.

Tabela I – Propriedades das esferas

Material	Diâmetro (mm)	Massa (g)
Vidro	$17,00 \pm 0,05$	$7,82 \pm 0,01$
Aço	$12,35 \pm 0,05$	$7,95 \pm 0,01$
Plástico	$12,25 \pm 0,05$	$0,88 \pm 0,01$

A simulação em Python utiliza as equações vistas no artigo de Freire [5] e foi construída utilizando as bibliotecas *Numpy* e *Tkinter* [6]. A simulação pode ser vista na figura 4.

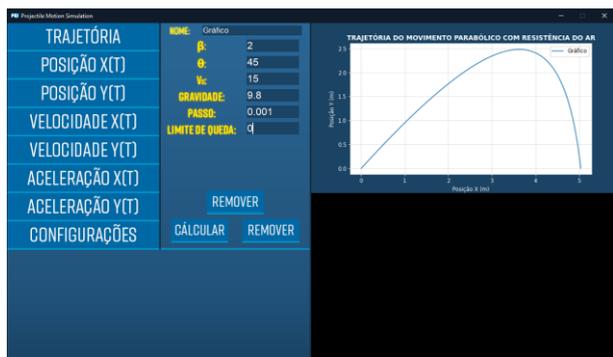


Figura 4 – Simulação construída em Python

3. Resultados

Com os dados de lançamento obtidos no *Tracker*, foi medida a distância horizontal (alcance) e a velocidade inicial para as 3 esferas, como mostra a tabela II.

Tabela II – Resultados do *Tracker* para as 3 esferas

Resultados do <i>Tracker</i> de cada esfera			
	Vidro	Aço	Plástico
Distância horizontal (m)	0,295	0,582	0,414
$v_{inicial}$ (m/s)	0,76	1,18	1,20

Utilizando os dados da tabela II como parâmetros iniciais em ambas simulações, foi possível comparar os resultados experimentais com os resultados simulados, que podem ser observados na tabela III e na tabela IV.

Tabela III – Resultados do *Interactive Physics*

Resultados do <i>Interactive Physics</i>			
	Vidro	Aço	Plástico
Distância horizontal (m)	0,298	0,480	0,412
$v_{inicial}$ (m/s)	0,76	1,18	1,20

Tabela IV – Resultados da simulação em *Python*

Resultados da simulação em <i>Python</i>			
	Vidro	Aço	Plástico
Distância horizontal (m)	0,298	0,477	0,414
$v_{inicial}$ (m/s)	0,76	1,18	1,20
β (1/s)	0,21000	0,00001	1,12500

Na simulação em Python, mesmo com um β bem próximo de 0 (quase sem resistência do ar) a distância

horizontal da esfera de aço não chegou próxima do resultado esperado. A razão pela qual a simulação da esfera de aço não ter alcançado a distância horizontal esperada foi devido a qualidade do contraste da imagem durante a gravação do tracking, o que dificulta a análise experimental [6].

4. Conclusões

Como o software em Python está em desenvolvimento, ainda existem funções a serem implementadas. Contudo, foi possível extrair bons resultados de ambas simulações.

É possível observar que os resultados do *Interactive Physics* e da simulação em *Python* ficaram bem próximos entre si, incluindo o da esfera de aço. Utilizando como base a distância horizontal das esferas de vidro e plástico obtida experimentalmente, foi possível avaliar o valor do parâmetro de atrito com o ar β utilizando a simulação em Python.

Como o *Interactive Physics* é um software pago e não recebe atualizações desde 2005, o aluno acaba tendo dificuldades para ter acesso ao software. Assim, o foco na continuação deste projeto deve ser o desenvolvimento da simulação em *Python*, já que o software é em código *open source*.

5. Referências

- [1] MEI/CNI, ABENGE. **Diretrizes para o Curso de Engenharia, 2018.** Disponível em: <http://www.abenge.org.br/documentos/PropostaDC_NABENGEMEI_CNI.pdf>. Acesso em: 22 set. 2020.
- [2] RUGARCIA, A. et al. **The Future of Engineering Education: Part 1. A vision for a new century.** Chemical Engineering Education, v. 34, p. 16–25, dez. 2000.
- [3] WANKAT, Phillip C.; BULLARD, Lisa G. **The Future of Engineering Education - Revisited.** Chemical Engineering Education, v. 50, p. 19–28, jan. 2016.
- [4] HALLIDAY, David; RESNICK, Robert; WALKER, Jearl. **Fundamentos de Física, Volume 1: Mecânica.** 10. ed. Rio de Janeiro: LTC - Livros Técnicos e Científicos Editora Ltda, 2016. v. 1.
- [5] FREIRE, Wilson Hugo C. et al. **Lançamento oblíquo com resistência do ar: Uma análise qualitativa.** Revista Brasileira de Ensino de Física, SciELO, v. 30, 2016. Disponível em: <http://www.scielo.br/scielo.php?script=sci_arttext&pid=S1806-11172008000300013&nrm=iso>. Acesso em: 22 set. 2020.
- [6] PALIERINI, Rafael Zacarias. **Elaboração De Simulações Associadas A Novas Experiências No Laboratório De Física 1.** 2020. Disponível em: <<https://bit.ly/33U2ze3>>. Acesso em: 22 set. 2020

Agradecimentos

Ao Centro Universitário FEI pela disponibilidade de suas instalações e equipamentos e apoio financeiro.

¹ Aluno de iniciação didática do Centro Universitário FEI. Projeto com vigência de 01/20 a 01/21.