

## PROGRAMMER EN PYTHON PROJET N°1 : ÉTUDE DU REBOND D'UNE BALLE DE TENNIS DE TABLE

L'étude de la chute et des rebonds d'une balle sur un sol horizontal immobile permet de mesurer la valeur du champ de pesanteur terrestre et le coefficient de restitution associé aux chocs.

### Présentation

### Expérience et exploitation

Muni d'une balle, on réalise l'expérience en la lâchant d'une hauteur  $h_0$  de l'ordre du mètre. La vitesse  $v_0$  juste avant le premier impact est mesurée. Cette phase du mouvement est l'occasion de tester la conservation de l'énergie mécanique.

La vitesse  $v_1$  juste après l'impact est ensuite mesurée. La vitesse après l'impact avec le sol est inférieure à la vitesse avant l'impact, l'énergie mécanique n'est pas conservée lors de ce choc. Le caractère inélastique des collisions est traduit par l'introduction du coefficient de restitution  $e$  défini par le rapport de l'énergie cinétique du mobile après le choc sur l'énergie cinétique avant le choc :

$$e = \frac{E_{c,après}}{E_{c,avant}}$$

La hauteur de rebond  $h_1$  est ensuite mesurée. Il est alors possible de vérifier que la hauteur de premier rebond  $h_1$  est égale à  $e \cdot h_0$  et que la série des valeurs des hauteurs de rebond successives  $h_n$  est égale à  $e^n \cdot h_0$ .

Remarque : l'étude de la conservation de l'énergie mécanique peut être menée sur le mouvement de la balle entre deux chocs consécutifs.

## Validation avec le logiciel Phyphox

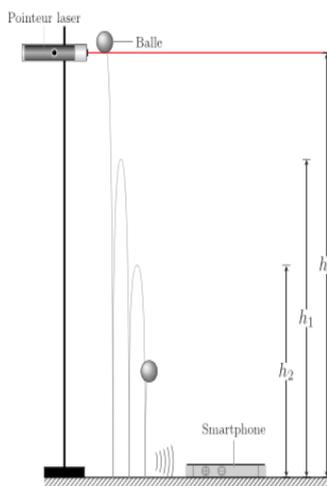
Cette configuration ne nécessite pas de système d'acquisition vidéo : il suffit de poser le smartphone sur le support, après avoir téléchargé le logiciel Phyphox.



Capture d'écran du logiciel

- Fixer un repère horizontal sur une potence et mesurer la hauteur de chute  $h_0$ .
- Placer le smartphone au niveau du sol. Ouvrir l'application Phyphox ou dictaphone d'un smartphone et passer en mode enregistrement.
- Placer la balle en  $h_0$  et la lâcher.

La balle entame alors une série de rebonds dont le son est enregistré par le smartphone.



Il est possible de montrer que les instants de rebonds  $T_n$  sont séparés dans le temps par des durées :

$$T_n = e^n \cdot \left(\frac{2 \cdot h_0}{g}\right)^{\frac{1}{2}}$$

La mesure des durées séparant les instants de chute permet de remonter à tous les paramètres de l'expérience.

Retrouvez éducol sur :



Le smartphone est ici utilisé en mode dictaphone commandé par l'application Phyphox. Cette application réalise une exploitation directe et affiche les différents paramètres de l'expérience après calcul.

Il reste cependant possible de faire réaliser aux élèves une partie de cette exploitation. Phyphox affiche la série des durées séparant deux impacts  $T_n$ . Le tracé de  $T_n = f(n)$  permet de remonter au coefficient de restitution  $e$  et l'intensité de la pesanteur  $g$ .

Le tracé et la détermination des paramètres du modèle peuvent être réalisés grâce au logiciel Python.

## Exemples d'exploitation

### Avec Phyphox

En lâchant une balle depuis une table, les données fournies par Phyphox ont été :

- intervalles entre deux chocs :

$$T_1 = 0,470 \text{ s} ; T_2 = 0,424 \text{ s} ; T_3 = 0,388 \text{ s} ; T_4 = 0,357 \text{ s} ; T_5 = 0,330 \text{ s} ;$$

- hauteurs calculées :

$$h_0 = 33,24 \text{ cm} ; h_1 = 27,09 \text{ cm} ; h_2 = 22,08 \text{ cm} ; h_3 = 18,47 \text{ cm} ; h_4 = 15,65 \text{ cm} ; h_5 = 13,36 \text{ cm} ;$$

- énergie totale dans le système :

$$E_1 = 100 \% ; E_2 = 81,5 \% ; E_3 = 68,2 \% ; E_4 = 57,8 \% ; E_5 = 49,3 \%$$

Le logiciel calcule les coefficients de restitution pour tous les chocs :

$$e_1 = 81,5 \% ; e_2 = 83,6 \% ; e_3 = 84,7 \% ; e_4 = 85,4 \%$$

D'où un coefficient de restitution moyen  $e_{\text{moy}} = 83,8 \%$ .

Retrouvez éduscol sur :



## Avec Python

Le script suivant permet de tracer en échelle logarithmique la durée entre deux rebonds en fonction du numéro du rebond.

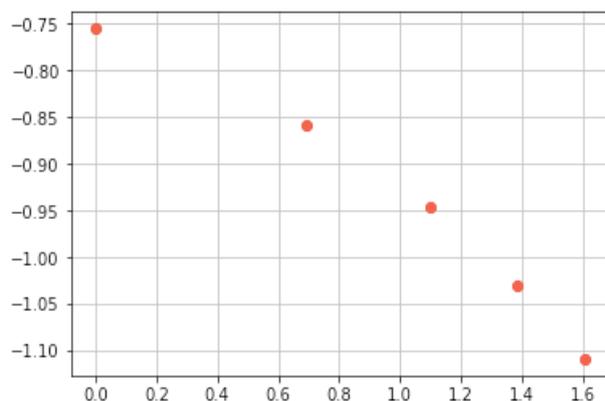
```
import numpy as np
import scipy
import matplotlib.pyplot as plt
```

```
durée=np.array([0.47,0.424,0.388,0.357,0.330])
```

```
num=1+np.arange(5)
```

```
lnum=np.log(num)
ldur=np.log(durée)
```

```
plt.plot(lnum,ldur,"ro")
plt.grid('true')
plt.show()
```



```
np.polyfit(lnum,ldur,1)
```

```
array([-0.21559847, -0.73326012])
```

D'après la théorie, la pente de la fonction est  $\ln(e)$ . Numériquement, le programme renvoie  $\ln(e) = -0,21$ , soit un coefficient de restitution  $e = \exp(-0,21) = 0,79$ .

Le résultat est très proche de celui calculé par le logiciel Phyphox.

D'autre part, l'ordonnée à l'origine permet de remonter à une valeur expérimentale de l'intensité de pesanteur  $g$ .

Retrouvez éducol sur :

