

## Thème 2 : Mouvements et interactions

### Partie 2. Relier les forces appliquées à un système à son mouvement

#### CHAP 14-ACT NUM Lois de Kepler

##### Objectif :

- Exploiter, à l'aide d'un langage de programmation, des données astronomiques ou satellitaires pour tester la deuxième et la troisième loi de Kepler.

#### A/ Vérification de la 3<sup>ème</sup> loi de Kepler

##### Introduction :

La validité de la troisième loi de Kepler est d'abord testée en exploitant un tableau relatif aux orbites et période de révolution des planètes au tour du Soleil. Ces données sont récupérées librement sur le site de l'observatoire de Paris.

#### 1) Récupération des données dans python

##### 1. Obtention du tableau des données

Sur la page de l'Observatoire de Paris « Peser l'Univers avec la troisième loi de Kepler », on trouve le tableau suivant relatif aux orbites et période de révolution des planètes au tour du Soleil :

| Planète | $a$    | $T_{\text{sid}}$ | $i$ | $e$   | $T^2/a^3$                        |
|---------|--------|------------------|-----|-------|----------------------------------|
|         | UA     | an               | deg |       | an <sup>2</sup> /UA <sup>3</sup> |
| Mercure | 0.3871 | 0.2408           | 7.0 | 0.206 | 0.9996                           |
| Vénus   | 0.7233 | 0.6152           | 3.4 | 0.007 | 1.0002                           |
| Terre   | 1.0000 | 1.0000           | --  | 0.017 | 1                                |
| Mars    | 1.5237 | 1.8808           | 1.8 | 0.093 | 1.0000                           |
| Jupiter | 5.2026 | 11.862           | 1.3 | 0.048 | 0.9992                           |
| Saturne | 9.5547 | 29.457           | 2.5 | 0.056 | 0.9948                           |
| Uranus  | 19.218 | 84.020           | 0.8 | 0.046 | 0.9946                           |
| Neptune | 30.109 | 164.77           | 1.8 | 0.009 | 0.9946                           |

Les valeurs ont été reportées dans un tableur intitulé [3loiKepler.csv](#). Le fichier doit être enregistré dans le même répertoire que le fichier python [3loiKepler.py](#)

##### 2. Récupération des données dans Python

- Compléter le script suivant afin de récupérer les données correspondant aux valeurs de  $a$  et  $T$ .

```
#importation des bibliothèques
import csv
import numpy as np
import matplotlib.pyplot as plt
source = open( 'NOMduFICHER.csv', 'r') # code à adapter

a=[] # création des listes
T=[]

reader = csv.reader(source,delimiter=',')
liste_aT = list( zip(*reader) ) #renvoie une liste de tuples: (a,T)

#Le script suivant permet de récupérer les données des colonnes 2 et 3 #Code à adapter
Grandeur en colonne 2= [ float( liste_aT[1][i].replace(',','.') ) for i in range(2,
len(liste_aT[1])) ]
Grandeur en colonne 3= [ float( liste_aT[2][i].replace(',','.') ) for i in range(2,
len(liste_aT[2])) ]

print("a(ua)", a) # imprime les valeurs
print("T(an)", T)

aa=np.array(a) # transforme les listes en array/tableau manipulables par numpy
tt=np.array(T)
```

```

a3=          # code à compléter pour calculer a^3
T2=          # code à compléter pour calculer T^2

# modélisation pour a^3 en fonction de T^2 #code à adapter
coeff_model=np.polyfit(Grandeur en x,Grandeur en y,1) # renvoie un array avec
les coefficients du polynôme du 1er degré
a3mod= coeff_model[0]*T2 # calcul des valeurs modélisées de a^3 d'après les
coeff du polynôme

# représentation graphique a^3 en fonction de T^2 + modèle #code à adapter
plt.grid()
plt.plot(T2,a3,'CouleurMarqueur')
plt.plot(T2,Grandeur Modélisée,'b-',label="modèle: a3="+format(coeff_model[0])+" x
T2")
plt.xlabel("Grandeur en x(unité)")
plt.ylabel("Grandeur en y(unité)")
plt.legend()
plt.title("Titre du graphique")
plt.show()

```

## 2) Exploitation

### 1. Calcul

- Compléter le programme pour:
  - Faire calculer les valeurs de  $a^3$  et  $T^2$

### 2. Modélisation

- Compléter le programme pour :
  - Modéliser la courbe  $a^3 = f(T^2)$  par un polynôme du 1<sup>er</sup> degré d'équation :  $a^3 = k \times T^2$

### 3. Représentation graphique

- Compléter le programme pour:
  - Représenter le nuage de points ( $a^3$ ,  $T^2$ )
  - Superposer la courbe modélisée + équation

### 4. Vérification de la 3ème loi de Kepler

- Conclure :

## B/ Vérification de la 2<sup>ème</sup> loi de Kepler

### Introduction :

La validité de la 2<sup>ème</sup> loi de Kepler est testée en exploitant les coordonnées cartésiennes spatiotemporelles relatives au mouvement de Mercure dans le référentiel héliocentrique. Les données sont récupérées sur le site de l'Institut de Mécanique Céleste et de Calcul des Ephémérides : <http://www.imcce.fr>

### 1) Récupération des données dans python

#### 1. Obtention du tableau des données

Allez sur la page d'accueil de l'IMCCE. Onglet Service « Ephémérides en ligne »

Ouvrir le portail des formulaires « SSP » (Planètes du Système Solaire) et choisir « éphémérides de position »

Compléter le formulaire pour obtenir l'éphémérides de  **Mercure**  dans le référentiel **héliocentrique**, en coordonnées **cartésiennes** (rectangulaires) par rapport au plan de l'**écliptique** sur une période de **88 jours** à compter de la date d'aujourd'hui et par **pas de 1 jour**. Les données obtenues peuvent être exportées au **format CSV** (séparateur ;).

- Vous disposez de données ainsi obtenues dans un fichier intitulé [mercure.csv](#)  
Rq : Par souci de simplification, n'ont été conservés que les 3 premières colonnes (t,x,y,z)  
Attention : Le fichier doit être enregistré dans le même répertoire que le fichier python [2loiKepler.py](#)

#### 2. Récupération des données dans Python

- Modifier le script du programme python fourni afin de récupérer les données correspondant aux valeurs (t,x,y,z) pour la planète Mercure.

## 2) Exploitation

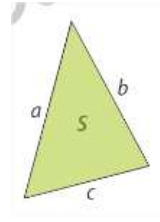
### 1. Aide au calcul

#### Formule de Héron d'Alexandrie

L'aire  $S$  d'un triangle quelconque dont les côtés ont pour longueurs  $a$ ,  $b$  et  $c$  s'obtient par :

$$S = \sqrt{p(p-a)(p-b)(p-c)}$$

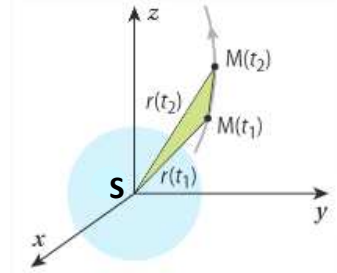
où  $p$  est le demi-périmètre du triangle.



#### Aire balayée

Soit  $S$  le centre du Soleil et  $M$  une planète du système solaire.

- L'aire  $S$  balayée par le segment  $[SM]$  entre les instants  $t_1$  et  $t_2$  est environ l'aire du triangle  $SM(t_1)M(t_2)$
- L'aire balayée par unité de temps est  $s = \frac{S}{t_2 - t_1}$   
On la nomme aussi *vitesse aréolaire*.



#### Rayon Soleil-Planète

Soit  $M(x,y,z)$  la position de la planète, à la date  $t$ , dans le référentiel héliocentrique .

Le rayon Soleil-Planète :

$$r(t) = \sqrt{x^2 + y^2 + z^2}$$

#### Distance parcourue par la planète

Si  $M(x_1, y_1, z_1)$  est la position de la planète à la date  $t_1$  et  $M(x_2, y_2, z_2)$  la position de la planète à la date  $t_2$ , la distance parcourue par la planète entre les instants  $t_1$  et  $t_2$  est obtenue par le théorème de Pythagore à trois dimensions:

$$M(t_1)M(t_2) = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2}$$

#### Vitesse de la planète

Le vitesse de la planète à la date  $t$  se calcule simplement comme :

$$v(t) = \frac{M(t_1)M(t_2)}{t_2 - t_1}$$

### 2. Programmation Python

Le programme Python déduit des valeurs  $(t, x, y, z)$ , les positions successives  $M(x, y, z)$  de la planète.

- Compléter le script du programme python fourni pour qu'il calcule à chaque position:
  - **La distance** entre la position de la planète et sa position suivante, dans la liste **d** ;
  - **La vitesse** du satellite à cette position, dans la liste **v** ;
  - **L'aire balayée** entre cette position et la position suivante, en utilisant la formule de Héron, dans la liste **S** ;
  - **La vitesse aréolaire** à cette position, dans la liste **s**.

### 3. Représentation graphique

- Exécuter le programme [2loiKepler.py](#). On obtient les courbes donnant :
  - l'évolution du rayon  $r$  de l'orbite de la planète au cours du temps
  - l'évolution de la vitesse linéaire  $v$  de la planète au cours du temps
  - l'évolution de la vitesse aréolaire  $s$  de la planète au cours
- Commenter l'allure des courbes  $r(t)$ ,  $v(t)$  et  $s(t)$

### 4. Vérification de la 2ème loi de Kepler

- En quoi les courbes obtenues montrent-elles que la 2<sup>ème</sup> loi de Kepler est vérifiée ?