Lycée Pierre-Gilles de Gennes 2024-2025

TP4 Python

Équations différentielles

L'objet de ce TP est de mettre en œuvre la méthode d'Euler sur différents exemples d'équations différentielles.
Ce TP prendra 2 séances.

1 Modèle de Gompertz

On considère le problème de Cauchy suivant sur $\mathbb{R}_+$:

$$(G) \qquad y' = ay\ln\left(\frac{\kappa}{y}\right) \quad \text{et} \quad y(t_0) = y_0$$

où a, κ sont des réels strictement positifs et $y_0 > 0$.
On rappelle [1] que l'unique solution y de (G) est donnée par :

$$\forall t \geq 0, \quad y(t) = \kappa e^{\ln\left(\frac{y_0}{\kappa}\right)e^{-at}}.$$

On se propose de déterminer numériquement une valeur approchée de y sur un segment $[0, T]$ $(T > 0)$ par la méthode d'Euler.
Dans toute cette partie, on prendra : `a` $= 0.036$ et `kappa` $= 760$, $(t_0, y_0) = (0, 16)$, $T = 200$.

1.1 Commande `odeint`

La commande `odeint` du module `scipy.integrate` permet de résoudre de manière approchée des équations différentielles. Pour résoudre une équation différentielle

$$y'(t) = F(y(t), t)$$

avec condition initiale $y(t_0) = y_0$ le script est le suivant

```python
import scipy.integrate as integr
def F(y,t):
        ''' fonction (y,t) --> F(y,t)
        '''
        return #a compléter

Abs = # liste de valeurs de t commençant par t_0
Y = integr.odeint(F,y_0,Abs)
plt.plot(Abs,Y)
plt.show()
```

Travail demandé

1. Compléter le script ci-dessus pour résoudre l'équation de Gompertz.
2. Comparer avec la représentation graphique de la solution exacte.

1. Voir l'exercice 7 du TD4 et son corrigé.

1.2 Méthode d'Euler

Soit $n \in \mathbb{N}^*$. On pose pour tout $k \in [\![0, n-1]\!]$:

$$y_{k+1} = y_k + \frac{T}{n} a y_k \ln\left(\frac{\kappa}{y_k}\right).$$

Travail demandé

1. Écrire une fonction `Euler` prenant en entrées y_0, T et n et renvoyant la suite $(y_k)_{k \in [\![0,n]\!]}$ obtenue par la méthode d'Euler.
2. Tracer la suite de points $\left(\left(k\frac{T}{n}, y_k\right)\right)_{k \in [\![0,n]\!]}$ avec $n = 5, 10, 50$ puis 500.

   On fera apparaître sur le même graphique la solution exacte et la solution obtenue avec `odeint`
3. **Erreur de la méthode d'Euler :** l'erreur commise par la méthode avec n subdivisions sur $[0, T]$ est définie par :

   $$e_n(T) = \max\left\{ |y_k - y\left(k\frac{T}{n}\right)| \ ; \ k \in [\![0, n]\!] \right\}.$$

   Il s'agit de l'écart entre la solution approchée et la solution exacte.

   (a) Écrire une fonction `Erreur` prenant en entrées `T`, `y0` et `n` et renvoyant la valeur de $e_n(T)$

   (b) Tracer le graphique de l'erreur en fonction du nombre de subdivisions $n \in [\![1, 100]\!]$. Que constate-on ?

2 Pendule pesant

On considère le système mécanique décrit en figure 1 composé d'une masse m suspendue à un fil rigide de longueur ℓ. La variable d'intérêt est l'angle θ que fait le fil avec la verticale et l'on cherche à décrire son évolution en fonction du temps t sachant qu'à un instant $t = t_0$ (on peut prendre $t_0 = 0$ pour fixer les idées) on a pour conditions initiales (position et vitesse angulaire initiales) :

$$\theta(t_0) = \theta_0 \quad , \quad \frac{d\theta}{dt}(t_0) = \omega_0.$$

La masse étant soumise à son poids et à la tension du fil rigide, le principe fondamental de la dynamique habilement projeté sur la direction tangentielle au mouvement se traduit en l'équation différentielle du second ordre :

$$\frac{d^2\theta}{dt^2} + \frac{g}{\ell}\sin\theta = 0 \qquad (\star)$$

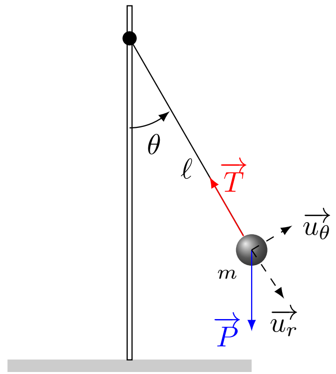

FIGURE 1 – Le pendule pesant

En posant

$$\omega = \frac{d\theta}{dt} \text{ (la vitesse angulaire)} \quad \text{et} \quad X = \begin{pmatrix} \theta \\ \omega \end{pmatrix} \quad ; \quad F(X) = \begin{pmatrix} \omega \\ -\frac{g}{\ell}\sin\theta \end{pmatrix},$$

on transforme l'équation $(\star)$ du second ordre en une équation vectorielle (un système d'équations scalaires) du premier ordre :

$$\frac{dX}{dt} = F(X).$$

Pour les simulations numériques, on prendra : $T = 8$, $t_0 = 0$, $(\theta_0, \omega_0) = (0, 8)$, $g = 9.80665$ et $\ell = 0.3$.

2.1 Résolution avec la commande `odeint`

On considère le script suivant (on suppose que toutes les constantes on été introduites avant) :

```
def F(X,t):
    return np.array([X[1],-g/l*np.sin(X[0])])

Abs = np.linspace(t0,T,1000)
X = integr.odeint(F,np.array([theta0,omega0]),Abs)
```

La variable `X` est une matrice à deux colonnes : la première colonne donne la liste des valeurs approchées de $\theta(t)$ pour $t \in$ `Abs` et la deuxième colonne donne la liste des valeurs approchées de $\omega(t)$ pour $t \in$ `Abs`.

Travail demandé

1. Recopier le script ci-dessus.
2. Afficher le graphe de $\theta(t)$ en fonction de $t \in$ `Abs`.

2.2 Méthode d'Euler

L'équation ($\star$) mise sous cette forme, avec des conditions initiales adéquates, peut être résolue approximativement sur un segment $[t_0, T]$ par un schéma d'Euler en considérant, pour tout $n \in \mathbb{N}^*$ la suite $(Y_k)_{[\![0,n]\!]}$ définie par :

$$Y_0 = \begin{pmatrix} \theta_0 \\ \omega_0 \end{pmatrix} \quad \text{et} \quad \forall k \in [\![0, n-1]\!], \quad Y_{k+1} = Y_k + \frac{T - t_0}{n} F(Y_k).$$

Travail demandé

1. Programmer la méthode d'Euler pour le pendule pesant.
2. Tracer la courbe $(t, \theta(t))$ pour $t \in [0, T]$ avec $n = 1000$.
3. Que constate-on ? Est-ce cohérent ?

2.3 Énergie conservée

Si on pose $X = (\theta, \omega)$ alors l'énergie mécanique totale du système est (à un facteur constant $m\ell^2$ près) :

$$E(X) = E(\theta, \omega) = \frac{1}{2}\omega^2 + \frac{g}{\ell}(1 - \cos(\theta)). \tag{E}$$

D'après la loi de conservation de l'énergie, pour $t \mapsto X(t)$ une solution de ($\star$) sur un intervalle temporel I la fonction $t \mapsto E(X(t))$ est constante sur I.
La trajectoire de la courbe $t \mapsto X(t) = (\theta(t), \omega(t))$ est donc contenue dans une courbe de niveau de la fonction E, la courbe de niveau correspondant au niveau initial $E_0 = E(\theta_0, \omega_0)$.
La formule de l'énergie est réglée de sorte que cette énergie soit toujours positive et que la position d'équilibre stable $(\theta_*, \omega_*) = (0, 0)$ soit d'énergie $E(0, 0) = 0$.
La position d'équilibre instable est $(\theta^*, \omega^*) = (\pi, 0)$ et l'énergie de cette position est $E^* = 2\dfrac{g}{\ell}$.

Travail demandé

1. Vérifier que l'énergie $t \mapsto E(X(t))$ est bien constante.
2. Écrire une fonction `energie` prenant en entrées des flottant θ et ω et renvoyant $E(\theta, \omega)$.
3. En reprenant les mêmes valeurs qu'au paragraphe précédent, tracer la courbe $(t, E(\theta(t), \omega(t))$ où $t \mapsto (\theta(t), \omega(t))$ est la valeur approchée obtenue par la méthode d'Euler.
   Que constate-on ? Est-ce cohérent ? Comment l'expliquer ?
4. Même question avec la valeur approchée obtenue par `odeint`.

3 Modèle proies-prédateurs de Lotka-Volterra

Soit a, b, c, d des réels strictement positifs.
On considère le système différentiel ci-dessous où x et y désignent respectivement l'effectif d'une population de proies et l'effectif d'une population de prédateurs en fonction de $t \in \mathbb{R}_+$.
Enfin, $(x_0, y_0) \in \mathbb{R}^2$ désigne les effectifs relevés à un instant $t_0 \in \mathbb{R}_+$.
On admet, pour tout $t_0 \in \mathbb{R}_+$ et tout $(x_0, y_0) \in \mathbb{R}^2$ l'**existence d'un unique couple** (x, y) de fonctions dérivables sur $\mathbb{R}_+$ vérifiant :

$$(S) \qquad \forall t \in \mathbb{R}_+, \quad \left\{ \begin{array}{lcl} x'(t) & = & (a - by(t))x(t) \\ y'(t) & = & (cx(t) - d)y(t) \end{array} \right. \quad \text{et} \quad \left\{ \begin{array}{l} x(t_0) = x_0 \\ y(t_0) = y_0 \end{array} \right.$$

On note $\varphi : \mathbb{R}_+^* \times \mathbb{R}_+^* \to \mathbb{R}$ définie par :

$$\forall (x, y) \in \left(\mathbb{R}_+^*\right)^2, \quad \varphi(x, y) = cx - d\ln(x) + by - a\ln(y).$$

On rappelle[2] que si x_0 et y_0 sont strictement positifs alors l'unique couple solution (x, y) vérifie la propriété (P_1) suivante :

$$(P_1) \qquad \forall t \geq 0, \quad \varphi(x(t), y(t)) = \varphi(x_0, y_0).$$

On admet également que les solutions sont périodiques et on note ρ la période. Ainsi, (x, y) vérifie la propriété (P_2) suivante :

$$(P_2) \qquad \forall t \geq 0, \quad (x(t + \rho), y(t + \rho)) = (x(t), y(t)).$$

On se propose de déterminer numériquement une valeur approchée d'un couple solution (x, y) sur un segment $[0, T]$ $(T > 0)$ par les méthodes d'Euler et de Heun.

Dans la suite, on prendra $a = b = c = 1$ **et** $d = 2$**.**

3.1 Méthode d'Euler

Pour tout $n \in \mathbb{N}^*$, à partir de la condition initiale $(x_0, y_0) \in (\mathbb{R}_+^*)^2$, on construit deux suites $(x_k)_{k \in [\![0,n]\!]}$ et $(y_k)_{k \in [\![0,n]\!]}$ définies par :

$$\forall k \in [\![0, n-1]\!], \quad \left\{ \begin{array}{lcl} x_{k+1} & = & x_k + \dfrac{T}{n} x_k(a - by_k) \\ \\ y_{k+1} & = & y_k + \dfrac{T}{n} y_k(cx_k - d) \end{array} \right.$$

Travail demandé

1. Écrire une fonction `Euler_syst` prenant en entrées x_0, y_0, T et n et renvoyant les suites $(x_k)_{k \in [\![0,n]\!]}$ et $(y_k)_{k \in [\![0,n]\!]}$ obtenues par la méthode d'Euler.

2. Voir le DS2.

2. Tracer les suites de points $\left(\left(k\frac{T}{n}, x_k\right)\right)_{k\in[\![0,n]\!]}$, $\left(\left(k\frac{T}{n}, y_k\right)\right)_{k\in[\![0,n]\!]}$ pour $x_0 = 1$, $y_0 = 0.5$, $T = 20$ et $n = 1000$.
3. (a) Tracer la suite de points $\left(\left(k\frac{T}{n}, \varphi(x_k, y_k)\right)\right)_{k\in[\![0,n]\!]}$.
   (b) Comparer le résultat obtenu avec la propriété (P_1) attendue en théorie.
4. (a) Tracer la représentation graphique de (y_k) en fonction de (x_k).
   (b) Comparer le résultat obtenu avec la propriété (P_2) attendue en théorie.

3.2 Méthode de Heun

Pour tout $n \in \mathbb{N}^*$, à partir de la condition initiale $(x_0, y_0) \in (\mathbb{R}_+^*)^2$, on construit deux suites $(x_k)_{k\in[\![0,n]\!]}$ et $(y_k)_{k\in[\![0,n]\!]}$ définies pour tout $k \in [\![0, n-1]\!]$ par :

$$\left\{\begin{array}{rcl} x^*_{k+1} & = & x_k + \dfrac{T}{n}x_k(a - by_k) \\ \\ y^*_{k+1} & = & y_k + \dfrac{T}{n}y_k(cx_k - d) \end{array}\right. \quad \text{et} \quad \left\{\begin{array}{rcl} x_{k+1} & = & x_k + \dfrac{T}{2n}(x_k(a - by_k) + x^*_{k+1}(a - by^*_{k+1})) \\ \\ y_{k+1} & = & y_k + \dfrac{T}{2n}(y_k(cx_k - d) + y^*_{k+1}(cx^*_{k+1} - d)) \end{array}\right.$$

Travail demandé

1. Écrire une fonction `Heun_syst` prenant en entrées x_0, y_0, T et n et renvoyant les suites $(x_k)_{k\in[\![0,n]\!]}$ et $(y_k)_{k\in[\![0,n]\!]}$ obtenues par la méthode de Heun.
2. Tracer les suites de points $\left(\left(k\frac{T}{n}, x_k\right)\right)_{k\in[\![0,n]\!]}$, $\left(\left(k\frac{T}{n}, y_k\right)\right)_{k\in[\![0,n]\!]}$ pour $x_0 = 1$, $y_0 = 0.5$, $T = 20$ et $n = 1000$.
3. (a) Tracer la suite de points $\left(\left(k\frac{T}{n}, \varphi(x_k, y_k)\right)\right)_{k\in[\![0,n]\!]}$.
   Indication : pour mieux se rendre compte, on pourra ajouter la commande suivante juste avant `plt.show()` :
   ```
   plt.ylim([2.19, 2.2])
   ```
   (b) Comparer le résultat obtenu avec la propriété (P_1) attendue en théorie.
4. (a) Tracer la représentation graphique de (y_k) en fonction de (x_k).
   (b) Comparer le résultat obtenu avec la propriété (P_2) attendue en théorie.