Outils numériques
ChON1.Résolution numérique d'équations différentielles

Méthode d'**Euler** explicite et Fonction **Odeint**

INTRO :
La résolution d'équations différentielles est nécessaire pour prévoir ou interpréter le comportement de divers systèmes. La résolution analytique n'étant pas toujours possible, on procède à une résolution numérique. On décrit ici la méthode d'Euler explicite qui correspond à un schéma numérique (relation de récurrence) relativement simple. Avec la bibliothèque scipy.integrate, on peut utiliser la fonction odeint.
Les applications sont variées :

- Mécanique (oscillations d'amplitude quelconque d'un pendule simple) : équation différentielle non linéaire d'ordre 2
- Electricité (filtrage numérique) : équation différentielle linéaire d'ordre n avec 2nd membre quelconque
- Thermodynamique (équation de la diffusion thermique) : équation différentielle aux dérivées partielles
- Chimie pour les MP : Cinétique chimique (équation différentielle non linéaire d'ordre 1) et couplage des aspects cinétique et thermodynamique d'une réaction chimique (équations différentielles couplées)

Plan du chapitre

A) Méthode d'Euler explicite

1) ED du 1er ordre

a) Principe

On cherche à résoudre une équation différentielle de la forme : $y'(t) = \frac{dy}{dt} = F(y, t)$ connaissant la condition initiale $y(t_0) = y_0$.

Le but de la résolution numérique d'une équation différentielle est d'obtenir une approximation de la solution exacte. On tente en général d'approcher la fonction solution exacte $y(t)$ en n points répartis sur l'intervalle $[t_0, t_f]$, donc en échantillonnant le temps.

En pratique, il faut transformer l'équation différentielle en une relation de récurrence permettant de calculer $y_{i+1} = y(t_{i+1})$ à partir de $y_i = t(t_i)$ et des paramètres du problème.

Schéma / Méthode d'Euler explicite : La dérivée est approximée par un taux de variation entre les instants t_i et t_{i+1}, et toutes les autres grandeurs sont prises à l'instant t_i.

La solution approchée de l'équation différentielle est donnée comme une « suite » (t_i, y_i) :

On considère une subdivision régulière de $[t_0, t_f]$:

$$t_{i+1} = t_i + dt \text{ avec le pas } dt = \frac{t_f - t_0}{n-1} \text{ et } i \in [\![0, n-1]\!]$$

On initialise la liste :

$$y_0 = y(t_0)$$

Les autres termes sont définis par le schéma d'Euler explicite :

$$\boldsymbol{\frac{y_{i+1} - y_i}{dt} \approx F(y_i, t_i)}$$

D'où la relation de récurrence :

$$\boldsymbol{y_{i+1} \approx y_i + dt \cdot F(y_i, t_i)}$$

Rq : - *Cette relation correspond à un DL à l'ordre 1 en dt.*

- Cela revient à approximer la fonction $y(t)$ pour $t \in [t_i, t_{i+1}]$ par la tangente à la courbe en t_i.

- Il existe des schémas numériques plus astucieux, plus précis et/ou plus rapides, d'où l'intérêt d'utiliser des fonctions issues de bibliothèques, cf § B.

b) Exemple pour une ED1 linéaire

♦ La tension aux bornes d'un condensateur d'un circuit RC série est régie par l'équation différentielle :

$$\frac{du}{dt} + \frac{1}{\tau}u = \frac{1}{\tau}e \qquad \text{avec} \qquad \begin{cases} \tau = RC \\ e(t) \text{ un forçage quelconque mais connu} \end{cases}$$

♦ Exemple d'implémentation avec une tension d'entrée $e(t)$ sinusoïdale :

```
1  import numpy as np
2  import matplotlib.pyplot as plt
3  
4  # Paramètres du circuit et condition initiale en USI
5  R = 1e3
6  C = 1e-6
7  tau = R*C
8  u0 = 0
9  # Amplitude et fréquence de la fém du GBF en USI
10 E0 = 2
11 f = 1e3
12 # Nombre d'itérations, instant final et pas de temps en USI
13 n = 10000
14 tf = 10*tau
15 dt = tf/(n-1)
16 
17 # Listes des instants et tensions d'entrée associées
18 t = [i*dt for i in range(n)]
19 e = [E0*np.cos(2*np.pi*f*ti) for ti in t]
20 # Création d'une liste pour les tensions du condensateur
21 u = [0 for i in range(n)]
22 
23 # Initialisation
24 u[0] = u0
25 # Résolution par la méthode d'Euler
26 for i in range(n-1):
27     u[i+1] =
28 
29 # Représentation graphique
30 …
31 plt.xlabel("t (s)")
32 plt.ylabel("u (V)")
33 plt.title("Euler ED1 tension d'un condensateur RC série entrée sinuoïdale")
34 plt.grid()
35 …
36 
37 # Affichage des différentes durées caractéristiques du problème
38 print("tau en s:",tau)
39 print("T en s:",1/f)
40 print("dt en s:",dt)
```

➲ Analyse du code proposé :

i) Commenter les lignes 18 à 21. Proposer une autre ligne de code pour la ligne 18 avec **np.linspace**.

ii) Compléter les lignes 27, 30 et 35.

♦ Pertinence des résultats

Un schéma numérique ne propose qu'une résolution numérique approchée de l'équation différentielle, il faut donc être critique sur la pertinence de ce résultat. **Pour que la méthode d'Euler puisse conduire à un résultat correct, le pas de temps `dt` doit être très petit devant tous les temps caractéristiques du système étudié.** Pour l'exemple traité, il faut donc `dt ≪ tau et dt ≪ 1/f`.

➲ Tester l'influence du pas de temps `dt` via le nombre `n` d'itérations sur la qualité de la résolution.

2) ED d'ordre $p > 1$

a) Principe

♦ Pour résoudre une ED d'ordre 2 avec la méthode d'Euler, il faut transformer l'ED d'ordre 2 en un système différentiel de 2 équations d'ordre 1 en introduisant une deuxième fonction inconnue égale à la dérivée première.

On cherche à résoudre une équation différentielle du 2e ordre de la forme : $y''(t) = \frac{d^2y}{dt^2} = F(y, y', t)$ connaissant les conditions initiales $y(t_0) = y_0$ et $y'(t_0) = y'_0$.

La solution approchée de l'équation différentielle est donnée comme une « suite » (t_i, y_i) :

On considère une subdivision régulière de $[t_0, t_f]$:

$$t_{i+1} = t_i + dt \text{ avec le pas } dt = \frac{t_f - t_0}{n-1} \text{ et } i \in [\![0, n-1]\!]$$

On initialise les listes :

$$y_0 = y(t_0) \text{ et } y'_0 = y'(t_0)$$

Et les autres termes sont définis par récurrence :

$$\begin{cases} \boldsymbol{y_{i+1} \approx y_i + dt \cdot y'_i} \\ \boldsymbol{y'_{i+1} \approx y'_i + dt \cdot F(y_i, y_i', t_i)} \end{cases}$$

NB : On peut généraliser cette démarche pour une ED d'ordre $p > 1$ quelconque.

b) Exemple pour une ED2 linéaire

Considérons un oscillateur masse-ressort horizontal amorti par une force de frottement fluide, dont l'équation du mouvement s'écrit sous forme canonique :

$$\ddot{x} + \frac{\omega_0}{Q}\dot{x} + {\omega_0}^2 x = {\omega_0}^2 x_{éq}$$

Pour ce problème, on introduit x la liste des positions et v la liste des vitesses.

➲ Déterminer les relations de récurrence donnant x_{i+1} et v_{i+1} à implémenter dans la boucle **for** de la méthode d'Euler.

B) Fonction odeint de la bibliothèque scipy.integrate

1) Principe d'utilisation pour une ED d'ordre 1

On peut utiliser la méthode de résolution d'équations différentielles contenue dans la bibliothèque **scipy.integrate** : **odeint**. En pratique, cette méthode est plus rapide à coder et plus fiable que la méthode d'Euler.

La fonction **odeint** impose d'écrire l'équation différentielle sous la forme :

$$y'(t) = \frac{dy}{dt} = F(y, t)$$

Il faut donc coder la fonction $F(y, t)$ avant d'utiliser la fonction **odeint**.

Les arguments de **odeint** sont :

- la fonction F (dont les arguments sont y et t à donner dans cet ordre lors de sa définition),
- la condition initiale $y(t_0) = y_0$,
- `time` le tableau des `n` instants (on cherche à résoudre l'équation différentielle en ces points).

➲ Exercice classique :

Application à l'exemple du circuit RC :

$$\frac{\mathrm{d}u}{\mathrm{d}t} + \frac{1}{\tau}u = \frac{1}{\tau}e \qquad \text{avec} \qquad \begin{cases} \tau = RC \\ e(t) \text{ un forçage quelconque mais connu} \end{cases}$$

i) Donner l'expression de la fonction $F(y, t)$ correspondant à cet exemple pour une tension d'entrée sinusoïdale.

ii) Coder la résolution de l'équation avec la fonction **odeint** et tracer la courbe $u(t)$.

2) Cas d'une ED d'ordre 2

On peut utiliser la fonction odeint pour résoudre une ED d'ordre $p = 2$.

Les arguments de **odeint** sont :

- la fonction F dont les arguments sont y et t avec y un tableau contenant $p = 2$ éléments, la fonction F retourne un tableau contenant $p = 2$ éléments relatif au système différentiel, cf A.2.a ;
- les conditions initiales : tableau de $p = 2$ éléments ;
- `time` le tableau des `n` instants.

Pour une ED d'ordre $p = 2$, la fonction **odeint** renvoie un tableau contenant `n` tableaux à $p = 2$ éléments.

Pour plus de détails, il est bien sûr conseillé de se référer à la documentation !

C) Applications de la méthode d'Euler

1) Au filtrage numérique ***: cf TP et TD d'électronique numérique***

2) A la résolution d'équations différentielles couplées ***: pour les MP cf cours et TD de thermochimie***

3) A la résolution de l'équation de diffusion – Méthode des différences finies ***: cf cours et TP sur les transferts thermiques***