

TD 22 - Résolution d'équations différentielles du 2nd Ordre - II

Informatique
MPSI-PCSI - Lycée Thiers

Exercice 1.

Énoncé

Correction

Exercice 1.

On souhaite résoudre le problème de Cauchy suivant sur l'intervalle $[0, 1]$:

$$\begin{cases} \ddot{y} = \sin^2(x) \\ \dot{y}(0) = 0 \\ y(0) = 0 \end{cases}$$

On se donne un entier $N=100$ et la subdivision régulière de $(x_n)_{0 \leq n \leq 100}$ de $[0, 1]$ qu'on se donne en python par `X = np.linspace(0,1,101)`. On souhaite résoudre ce problème en constituant le tableau `Y` contenant les valeurs $y(x_0), y(x_1), \dots, y(x_N)$.

1. Première méthode.

- 1.1 Ecrire l'équation différentielle à l'ordre 2 : $\ddot{y} = \sin^2(x)$ comme un système de deux équations différentielles à l'ordre 1.
- 1.2 Résoudre le problème de Cauchy en donnant le code qui permet d'obtenir le tableau `Y`. (On pourra utiliser la fonction `odeint()` de `scipy.integrate`).

Exercice 1.

2. Deuxième méthode

- 2.1 On part d'un tableau vide : `Y=np.empty(N+1)`. Quelle valeur devra prendre `Y[0]` dans la solution apportée ?
- 2.2 Quelle valeur devra prendre `Y[1]` ?
- 2.3 En appliquant le schéma d'Euler explicite à l'ordre 2 :

$$\ddot{y}(x_k) \approx \frac{y(x_{k+1}) - 2y(x_k) + y(x_{k-1}))}{(1/N)^2}$$

exprimer pour tout $k \in [[1, N-1]]$, une valeur approchée obtenue par cette approximation de `Y[k+1]` en fonction de `N`, `Y[k]`, `Y[k-1]` et de `np.sin(X[k])`.

- 2.4 Donner le code permettant de constituer le tableau `Y` des valeurs prises par `y` aux points de la subdivision régulière $(x_n)_{0 \leq n \leq 100}$ de $[0, 1]$.
3. Tracer dans une même fenêtre les deux graphiques obtenus par les deux méthodes.

Exercice 1. Corrigé

1. 1.1 Soit $Y(x) = \begin{pmatrix} y(x) \\ \dot{y}(x) \end{pmatrix}$; le système est équivalent à :

$$\frac{d}{dx} Y(x) = \frac{d}{dx} \begin{pmatrix} y(x) \\ \dot{y}(x) \end{pmatrix} = \begin{pmatrix} \dot{y}(x) \\ \sin^2(x) \end{pmatrix} \quad \text{avec} \quad Y(0) = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

Exercice 1. Corrigé

1. 1.1 Soit $Y(x) = \begin{pmatrix} y(x) \\ \dot{y}(x) \end{pmatrix}$; le système est équivalent à :

$$\frac{d}{dx} Y(x) = \frac{d}{dx} \begin{pmatrix} y(x) \\ \dot{y}(x) \end{pmatrix} = \begin{pmatrix} \dot{y}(x) \\ \sin^2(x) \end{pmatrix} \quad \text{avec} \quad Y(0) = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

1.2 Résolution à l'aide de odeint :

```
import numpy as np
from math import sin

f = lambda Y,x : np.array([Y[1], sin(x)**2])
y0 = np.array([0,0])
X = np.linspace(0,1,101)
y = integrate.odeint(f,y0,X)
Y1 = y[:,0]      # récupérer la première colonne
```

Exercice 1 : Correction.

Résolution par la méthode d'Euler :

```
N = 100
X = np.linspace(0,1,N+1)
Y = np.empty(N+1)
Z = np.empty(N+1)
Y[0] = 0
Z[0] = 0
dx = 1/N
for k in range(N):
    Y[k+1] = Y[k] + dx * Z[k]
    Z[k+1] = Z[k] + dx * sin(X[k])**2

import matplotlib.pyplot as plt
plt.figure(1)
plt.clf()
plt.subplot(2,1,1)
plt.plot(X,Y)
```

Exercice 1.2. Corrigé

2. 2.1 Valeur de $Y[0]$:

```
Y = np.empty(N+1)
Y[0] = 0
```

2.2 Valeur de $Y[1]$: (en appliquant le schéma d'Euler d'ordre 1) :

```
Y[1] = 0
```

en effet : $Y[1] - Y[0] \approx \frac{1}{N} y'(0)$

2.3. Obtention de $Y[k+1]$:

En appliquant le schéma d'Euler explicite à l'ordre 2 :

$$Y[k+1] - 2*Y[k] + Y[k-1] = dx**2 * \sin(X[k])**2$$

ce qui permet d'exprimer $Y[k+1]$ en fonction de $Y[k]$, $Y[k-1]$, $X[k]$ et dx :

```
Y[k+1] = np.sin(X[k])**2/N**2 + 2 * Y[k] - Y[k-1]
```

car ici $dx = 1/N$, et multiplier par dx^2 revient à diviser par N^2 .

2.4. Obtention de Y :

```
for k in range(1,N):
    Y[k+1] = np.sin(X[k])**2/N**2 + 2 * Y[k] - Y[k-1]
```


Exercice 1 : Correction.

2)

```
Y1 = np.empty(N+1)
dx = 1/N
Y1[0] = 0
Y1[1] = 0
for k in range(1,N):
    Y1[k+1] = dx**2*sin(X[k])**2 + 2*Y1[k]-Y1[k-1]
plt.subplot(2,1,2)
plt.plot(X,Y1)
```

