TP 8 : Simulation d'un système

- ⚠ Vous utiliserez un fichier différent pour chacun des deux exercices.
- Dans ce TP, nous n'aurons pas besoin d'écrire de fonctions (voir au besoin le résumé de cours).

Exercice 1 *SII / Moteur à courant continu.* On alimente un moteur initialement à l'arrêt par une tension constante $U = 10$ V. La vitesse de rotation w du moteur est solution de l'équation différentielle :

$$J\frac{\mathrm{d}w}{\mathrm{d}t} + \left(\frac{K_e K_c}{R} + f\right) w(t) = \frac{K_c}{R}U \quad \text{avec} \quad w(0) = 0$$

où :

- J est l'inertie du moteur, $J = 5.5 \cdot 10^{-5}\ \mathrm{kg}\cdot\mathrm{m}^2$;
- R est la résistance électrique du système, $R = 5.2\ \Omega$;
- K_c et K_e sont des constantes, $K_c = 0.24\ \mathrm{N}\cdot\mathrm{m}\cdot\mathrm{A}^{-1}$ et $K_e = 0.24\ \mathrm{V}\cdot\mathrm{rad}^{-1}\cdot\mathrm{s}$;
- f est un coefficient caractérisant les forces de frottement, $f = 0.05\ \mathrm{N}\cdot\mathrm{m}\cdot\mathrm{s}\cdot\mathrm{rad}^{-1}$.

On notera cette équation différentielle sous la forme :

$$\frac{\mathrm{d}w}{\mathrm{d}t} = aw(t) + b \quad \text{avec} \quad a = -\frac{1}{J}\left(\frac{K_c K_e}{R} + f\right) \quad \text{et} \quad b = \frac{K_c U}{JR}$$

On réalise une simulation avec $N = 100$ étapes de calcul et un pas de temps $\Delta_t = 10^{-4}$ s.

(a) Définir les différentes constantes intervenant dans le problème.

(b) Définir la liste $t = [t_0, \ldots, t_{N-1}]$ avec $t_i = i\Delta_t$ ainsi qu'une liste w de taille N, dont tous les termes sont pour l'instant nuls (et qui contiendra, une fois les calculs effectués, les valeurs $w(t_0), \ldots, w(t_{N-1})$).

(c) À partir de l'équation différentielle, déterminer comment on va définir $w[i+1]$ à partir de $w[i]$.

(d) Écrire la boucle réalisant ce calcul.

(e) Représenter graphiquement w en fonction de t avec :

```python
import matplotlib.pyplot as plt
plt.plot(t,w)
plt.xlabel("t (s)")
plt.ylabel("w (rad/s)")
plt.title("Vitesse de rotation du moteur")
plt.show()
```

(f) Vérifier (numériquement) que la vitesse du moteur tend vers une limite pour t assez grand et que cette limite est $-b/a$. On pourra ajouter les instructions suivantes entre `import...` et `plt.plot(t,w)` :

```python
w0 = -b/a
plt.axhline(y=w0,color='k',linestyle='--')
```

(g) Déterminer (graphiquement) le temps t pour lequel $w \geqslant 0.95 w_0$.

Exercice 2 *Évolution d'une population, modèle de* VERHULST. Le Parc national Kruger est la plus grande réserve animalière d'Afrique du Sud. Il couvre près de 20 000 km^2, est long de 350 km du nord au sud et large de 60 km d'est en ouest, ce qui rend sa taille comparable à celle d'Israël ou à celle du Pays de Galles. En 1905, un petit groupe de 10 éléphants y fut repéré ; en 2004, la population d'éléphants y a atteint 11 670 individus alors que le parc n'est prévu que pour 8 000 éléphants. La contraception animale y est donc utilisée depuis 1995. On donne ci-dessous l'évolution de la population d'éléphants depuis 1905 :

t	1905	1923	1930	1939	1945	1950	1960	1970	1980	1990	2000
Population	10	13	29	450	980	3010	5800	6500	7400	7200	7310

On souhaite comparer l'évolution de cette population avec le modèle théorique de VERHULST. Dans ce modèle, la fonction $N(t)$ donnant l'effectif de la population à l'instant t est solution de l'équation différentielle :

$$\frac{\mathrm{d}N}{\mathrm{d}t} = r\left(1 - \frac{N(t)}{M}\right)N(t)$$

avec la condition initiale $N(1905) = 10$. Dans cette équation, M et r sont des paramètres, on prendra $M = 7300$ et $r = 0.14$.

On veut réaliser une simulation de ce modèle pour le comparer aux valeurs mesurées, on prendra $\Delta_t = 1$ (le temps est mesuré en années), $t_0 = 1905$ et $n = 100$.

(a) Définir les différentes constantes intervenant dans le problème.

(b) Définir les listes $t = [t_0, \dots, t_n]$ avec $t_i = t_0 + i\Delta_t$ et N la liste de taille $n+1$ dont tous les termes sont pour l'instant égaux à 10 (c'est à dire $N(1905)$).

(c) À partir de l'équation différentielle, déterminer comment on va définir $N[i+1]$ à partir de $N[i]$.

(d) Écrire la boucle réalisant ce calcul.

(e) Représenter graphiquement N en fonction de t avec :

```python
import matplotlib.pyplot as plt
plt.plot(t,N,'b-')
plt.show()
```

(f) Définir les listes **t_mes** et **N_mes** correspondant aux effectifs mesurés puis représenter simultanément les valeurs mesurées et les valeurs calculées avec :

```python
plt.plot(t_mes,N_mes,'r.')
plt.legend(["Modèle","Mesures"],loc='lower right')
plt.title("Évolution de la population")
plt.xlabel("t (années)")
plt.ylabel("N")
plt.show()
```

Remarque. Pour que les valeurs obtenues par la simulation soient pertinentes, il faut que le pas de temps Δ_t soit choisi judicieusement. Disons simplement que Δ_t doit être « petit » devant le temps de réaction du système. Ceci a pour conséquence qu'il faut déjà avoir une certaine connaissance du système avant de le simuler. □

Corrections

Ex 1.

```python
import matplotlib.pyplot as plt

# Constantes du problème

U = 10.0
J = 5.5e-5
R = 5.2
Kc = 0.24
Ke = 0.24
f = 0.05
a = -1.0/J*(Kc*Ke/R+f)
b = Kc*U/(J*R)

# Paramètres de la simulation

N = 100
Delta_t = 1e-4

t = [i*Delta_t for i in range(0,N)]
w = [0]*N
```

L'équation différentielle permet d'écrire (sachant que $t[i+1]-t[i]$ est « petit ») :

$$\frac{w[i+1]-w[i]}{t[i+1]-t[i]} \simeq aw[i]+b$$

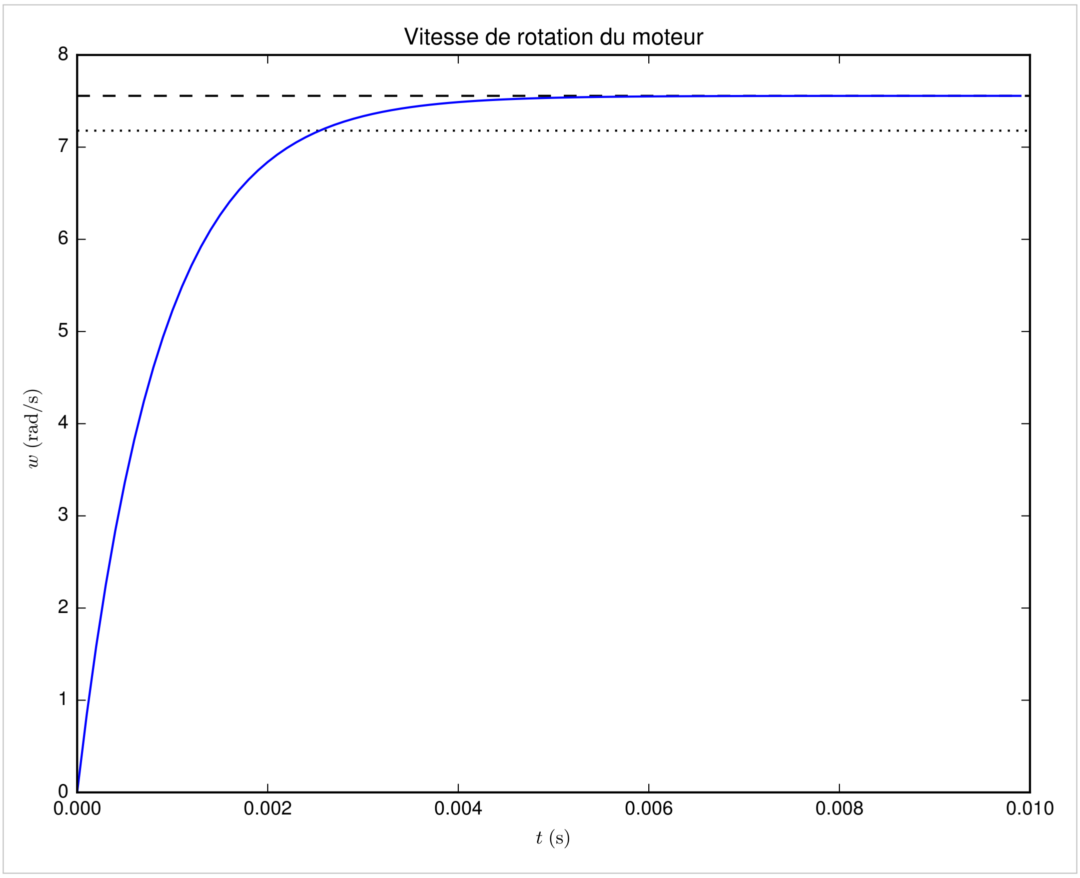
de sorte que l'on définit $w[i+1]$ par :

$$\forall i \in [\![0, N-2]\!],\ w[i+1] = w[i] + (t[i+1]-t[i])(aw[i]+b)$$

```python
for i in range(0,N-1):
    w[i+1]=w[i]+(t[i+1]-t[i])*(a*w[i]+b)
```

Représentation graphique :

```python
w0 = -b/a
plt.axhline(y=w0,color='k',linestyle='--')
plt.axhline(y=0.95*w0,color='k',linestyle=':')
plt.plot(t,w)
plt.xlabel("$t~(\mathrm{s})$")
plt.ylabel("$w~ (\mathrm{rad/s})$")
plt.title("Vitesse de rotation du moteur")
```



On trouve que le temps t pour lequel $w \geqslant 0.95 w_0$ est de l'ordre de $2.6 \cdot 10^{-3}$ s.

Ex 2.

```python
import matplotlib.pyplot as plt

M = 7300
r = 0.14
t0 = 1905
n = 100
Delta_t = 1
t = []
N = []
for i in range(0,n+1):
    t.append(t0+i*Delta_t)
    N.append(10)
for i in range(0,n):
    N[i+1] = N[i]+Delta_t*r*(1-N[i]/M)*N[i]

t_mes = [1905,1923,1930,1939,1945,1950,1960,1970,1980,1990,2000]
N_mes = [10,13,29,450,980,3010,5800,6500,7400,7200,7310]

plt.plot(t,N,'b-')
plt.plot(t_mes,N_mes,'r.')
plt.legend(["Modèle","Mesures"],loc='lower right')
plt.title("Évolution de la population")
plt.xlabel("t (années)")
plt.ylabel("N")
```