Étude du pendule oscillant

L'objectif de cette étude est la résolution de l'équation différentielle du mouvement d'un pendule oscillant par des méthodes numériques. On souhaite mettre en évidence l'impact du pas de discrétisation sur la qualité des résultats et sur la durée du calcul. On comparera les résultats obtenus avec ceux des fonctions déjà implémentées dans les bibliothèques numériques de Python.

Remarques :
- certaines constantes sont tabulées dans scipy

from scipy import constants g = constants.g

1. Présentation du problème physique

On considère un pendule oscillant de longueur $L = 1m$ et de masse $m = 1$kg dans le champ de pesanteur terrestre $\vec{g} = -g\overrightarrow{u_Z}$. Celui-ci est repéré par la position angulaire $\theta(t)$angle fait par le pendule avec la verticale descendante. Dans un premier temps, on négligera les frottements.
La mise en équation par la deuxième loi de Newton conduit à :

$$\ddot{\theta}(t) = -\frac{g}{L}\sin\big(\theta(t)\big)$$

2. Résolution de l'équation dans le cas linéaire :

Afin de comparer les résultats obtenus par ces méthodes numériques avec une solution exacte connue, on se place dans le cas des petits angles ($\theta(t) \leq 1/10$ alors sin $(\theta) \approx \theta$ à mieux que 1 % près).

2.1. Solution exacte

- Déterminer la solution exacte de l'équation dans le cas linéaire. Donner sa période.
- Tracer l'évolution temporelle de $\theta(t)$.

Vous justifierez notamment le choix de la durée de l'étude, les conditions initiales et le pas d'intégration.

2.2. Par la méthode d'Euler

- Réaliser la discrétisation du problème ci-dessus qui permette la résolution approchée de l'équation différentielle par la méthode d'Euler.
- Implémenter en python un programme qui permette de résoudre l'équation différentielle du mouvement par la méthode d'Euler et ainsi de tracer l'évolution temporelle de $\theta(t)$.

Vous justifierez notamment le choix de la durée de l'étude, les conditions initiales et le pas d'intégration.

2.3. Grâce à des fonctions numériques de la bibliothèques scipy

- En utilisant la fonction déjà implémentée « odeint » de la bibliothèque scipy.integrate de Python, écrire un programme qui permette de résoudre à nouveau l'équation différentielle du mouvement du pendule simple et de déterminer l'évolution temporelle de $\theta(t)$.

3. **Analyse des résultats obtenus dans le cas linéaire**

3.1. Qualité du résultat

- Dans un premier temps, la comparaison de ces deux méthodes avec la solution exacte se fera de manière graphique en superposant, sur un même graphique, les courbes obtenues.
- Dans un second temps, on affichera l’erreur en fonction du temps entre la solution exacte et la solution calculée par intégration numérique.
- On notera ε_N l’erreur numérique, qui sera le maximum de la valeur absolue de la différence entre la solution exacte et la solution calculée par intégration numérique.
- Dresser un tableau de la valeur de ε_N en fonction du pas d’intégration et de la méthode utilisée. Conclure.

3.2. Durée du calcul

Le module `time` peut aussi tester le temps nécessaire à la mise en œuvre de chacune des méthodes. Pour cela, il faut :

- enregistrer l’instant de l’horloge au début de l’algorithme (en seconde)

```
init=time.clock()
```

- enregistrer l’instant de l’horloge à la fin de l’algorithme

```
fin=time.clock()
```

- afficher la durée écoulée

```
print(fin-init)
```

- Dresser un tableau de la durée de calcul en fonction du pas d’intégration et de la méthode utilisée. Conclure.

4. **Analyse des résultats obtenus dans le cas non-linéaire**

On se place maintenant dans le cas où la résolution exacte n'est plus possible (cas des grands angles). Des résultats expérimentaux vous sont donc fournis dans le fichier annexe pendule.txt (les conditions expérimentales sont précisées dans l'en-tête du fichier). Importer ces données afin de pouvoir les exploiter pour la comparaison avec des résultats numériques.
Comparer les résultats obtenus par les deux méthodes de résolution numérique et les résultats expérimentaux. A nouveau, vous explorerez l'influence du pas d'intégration sur la qualité des résultats et le temps de calcul.