

TP2 - Étude de la trajectoire de Jupiter autour du Soleil

Février 2025

1 Contexte et objectif

Jupiter, la plus grande planète du Système solaire, décrit une trajectoire complexe sous l'influence prédominante de la force gravitationnelle exercée par le Soleil. L'objectif de ce TP est de simuler la trajectoire de Jupiter en utilisant la méthode d'intégration d'Euler et de comparer cette simulation avec les données réelles fournies par l'outil **Horizons** (de la NASA).

Le contenu du TP est accessible à cette adresse : <https://github.com/davidAlgis/TP-Jupiter-trajectory/archive/refs/heads/main.zip>.

Le projet est divisé en deux scripts python :

- `horizons_jupiter.py` : code de récupération des données et d'affichage, celui-ci a déjà été préparé pour vous, **pas besoin d'y toucher** par contre assurez vous qu'il est toujours dans le même dossier que l'autre script `euler_integration.py`.
- `euler_integration.py` : squelette du code dans lequel vous devez implémenter la fonction `euler_method` qui vous permettra de simuler la trajectoire de Jupiter.

Pour la simulation, les données issues de la base **Horizons**¹ sont utilisées :

- Période d'observation : du **1^{ère} janvier 2010** au **13 février 2025**.
- Pas de temps : **1 jour**.
- Positions et vitesses héliocentriques de Jupiter en unités astronomiques (UA)² et UA/jour respectivement.

On rappelle que la trajectoire de Jupiter est modélisée à partir de la deuxième loi de Newton, en considérant l'équation différentielle suivante :

$$\frac{dv}{dt} = -\mu \frac{\mathbf{r}}{\|\mathbf{r}\|^3}, \quad (1)$$

où \mathbf{r} est le vecteur position de Jupiter par rapport au Soleil et μ est le paramètre gravitationnel du Soleil :

$$\mu = 0.00029591220828559104 \quad [\text{UA}^3/\text{jour}^2]. \quad (2)$$

¹accessible gratuitement à cette adresse <https://ssd.jpl.nasa.gov/horizons/app.html> et automatiquement récupérées dans notre programme.

²On utilise des unités astronomiques parce que les distances sont gigantesques 1 UA = 149 597 870.7 km

2 Avant de commencer

1. Pour lancer la simulation vous devez dans un premier temps installer les *packages* python nécessaires aux scripts. Pour cela, ouvrez une console et lancez la commande suivante :

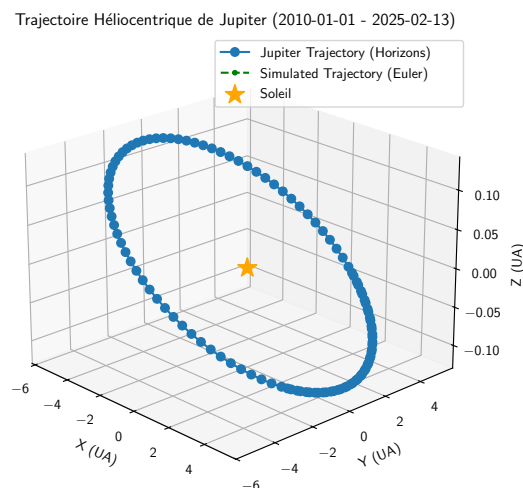
```
pip install -r [chemin-vers-requirements.txt]
```

où [chemin-vers-requirements.txt] est le chemin vers le fichier `requirements.txt`, par exemple chez moi c'est `C:/Users/david/Downloads/TP2/requirements.txt`. Cette commande va lancer le téléchargement et l'installation des *packages* nécessaire pour le programme.

2. Le point d'entrée du programme est le script `euler_integration.py`. Lancez-le soit depuis Sublime Text avec `Ctrl+B` ou `Ctrl+Shift+B` en exécutant `Launch TP2` ou encore manuellement depuis la console avec :

```
python euler_integration.py
```

3. Au premier lancement vous devriez obtenir la Figure 2. La console devrait afficher : *Erreur relative moyenne de la simulation Euler par rapport aux données Horizons : 100.0000%*



3 Exercice

1. Rappelez le principe de la méthode d'Euler.
2. Présentez en quelques lignes la façon dont vous appliquez la méthode d'Euler à l'équation différentielle 1.
3. Complétez la fonction `euler_method` dans le fichier `euler_integration.py`. En utilisant la méthode d'Euler, vous devez remplir les tableaux `sim_x`, `sim_y`, `sim_z`, `sim_vx`, `sim_vy`, `sim_vz`. Vous aurez réussi quand l'erreur relative sera inférieure à 5%.
4. Modifier la variable `dt` (définie en dessous de la fonction `euler_method`) et tester avec différentes valeurs d'entiers (par exemple 2, 5 et 10). Que constatez vous ?

Bonne chance et n'hésitez pas à poser des questions en cas de difficulté !