# TP11, méthodes d'integration numérique

As usual, regardez dans le dossier idoine sur GitHub.

Le but du TP est de produire un simulateur à Ncorps complet qui va intégrer les équations du mouvement de chacun des N corps à l'aide de la méthode de d'Euler ou de la méthode Verlet.

Partie I

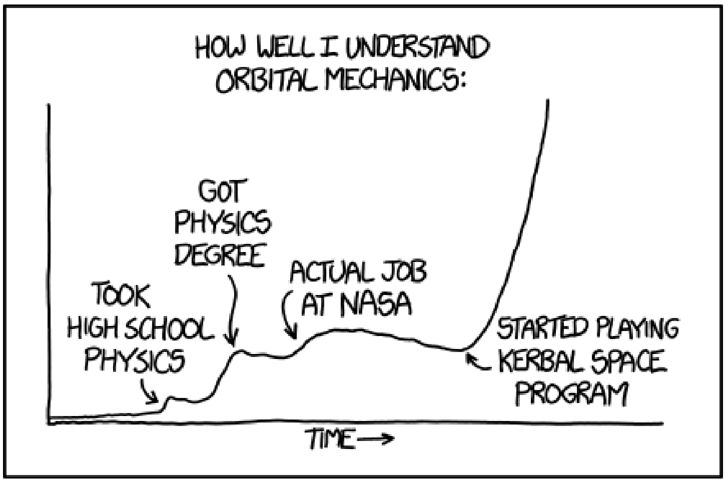
## Fabrication d'un simulateur à N corps : préparatifs

On s'intéresse donc à la dynamique d'un système de N corps en interaction gravitationnelle. Dans la suite, les corps considérés sont assimilés à des points matériels $P_j$ de masses $m_j$ où $j \in [\![ 0\,; N-1 ]\!]$ avec $N \geq 2$ un entier positif donné. Le mouvement de ces points est étudié dans un référentiel galiléen muni d'une base cartésienne orthonormée $(\overrightarrow{e^{(0)}}, \overrightarrow{e^{(1)}}, \overrightarrow{e^{(2)}})$. L'interaction entre deux corps $j$ et $k$ est modélisée par la force gravitationnelle. L'action exercée par le corps $k$ sur le corps $j$ est décrite par la force

$$\overrightarrow{F_{k/j}} = G\,\frac{m_j m_k}{r_{jk}^{\,3}}\,\overrightarrow{P_jP_k} = G\,\frac{m_j m_k}{\left(\displaystyle\sum_{\ell=0}^{2}\left(x_k^{(\ell)} - x_j^{(\ell)}\right)^2\right)^{3/2}}\sum_{\ell=0}^{2}\left(x_k^{(\ell)} - x_j^{(\ell)}\right)\overrightarrow{e^{(\ell)}}$$

où $x_k^{(\ell)}$ est la $\ell^{\text{e}}$ coordonnée spatiale associée à la particule $k$. Pour simplifier, on prendra $G = 1$ dans toute la suite.

1. Écrire une fonction `force2(m1,p1,m2,p2)` qui prend en arguments deux flottants `m1` et `m2` représentant respectivement les masses des deux particules en interaction ainsi que deux listes `p1` et `p2` de trois éléments représentant les positions de chaque particule par rapport à l'origine du référentiel. La fonction doit renvoyer une liste de trois flottants représentative de la force que la particule 2 exerce sur la particule 1 dans la base cartésienne (ne pas oublier qu'on prend $G = 1$).
2. Écrire une fonction `forceN(j,m,pos)` qui prend en paramètres l'indice `j` d'un corps, la liste `m` des masses des N corps du système étudié ainsi que la liste `pos` de leurs positions. La fonction doit renvoyer $\overrightarrow{F_j}$, la force exercée par tous les autres corps sur le corps `j` sous la forme d'une liste de ses trois composantes cartésiennes.




xkcd.com

```
To be fair, my job at NASA was working on robots
and didn't actually involve any orbital mechanics.
The small positive slope over that period is
because it turns out that if you hang around at NASA,
you get in a lot of conversations about space.
```

Partie II

## Fabrication d'un simulateur à N corps : Intégrateurs

Intégrer une équation différentielle du second ordre $\ddot{y} = f(y)$ revient, par la méthode d'Euler, à intégrer deux équations du premier ordre :

$$\begin{cases} \dfrac{\mathrm{d}y}{\mathrm{d}t} = z \\ \dfrac{\mathrm{d}z}{\mathrm{d}t} = f(y) \end{cases}$$

Posons $J_n = [\![\, 0\,;\, n \,]\!]$ avec $n$ le nombre de pas d'intégration d$t$ que l'on veut effectuer après avoir pris connaissance des conditions initiales. En notant $z = \dot{y}$ la dérivée temporelle de la fonction $y$ recherchée, le schéma d'intégration va permettre de définir deux suites $(y_i)_{i\in J_n}$ et $(z_i)_{i\in J_n}$ définies par les relations de récurrence suivantes

$$y_{i+1} = y_i + z_i\,\mathrm{d}t \qquad \text{et} \qquad z_{i+1} = z_i + f_i\mathrm{d}t$$

en ayant pris le soin de poser $f_i = f(y_i)$.

En se souvenant que l'application de la relation fondamentale de la dynamique à la particule $j$ s'écrit, en utilisant les notation introduites précédemment

$$\frac{\mathrm{d}^2\overrightarrow{OP_j}}{\mathrm{d}t^2} = \frac{\overrightarrow{F_j}}{m_j}$$

on va pouvoir utiliser le schéma d'Euler pour intégrer les équations du mouvement.

3. Écrire une fonction `etats_suivants_euler(m,pos,vit,dt)` qui prend en paramètres la liste `m` des masses des N corps du système étudié, la liste de leurs positions à l'instant $t_i$, la liste des vitesses au même instant et le pas d'intégration `dt`. La fonction doit renvoyer (et non modifier !) la liste des positions et la liste des vitesses des N corps à l'instant $t_{i+1} = t_i + \mathrm{d}t$ calculées à l'aide du schéma d'Euler [1].

Le schéma de Verlet proposé par le physicien français Loup Verlet en 1967 est un schéma d'intégration d'une équation de la forme $\ddot{y} = f(y)$ qui donne un résultat bien meilleur que la méthode d'Euler. Comme dans la méthode d'Euler, on cherche à trouver deux suites $(y_i)_{i\in J_n}$ et $(z_i)_{i\in J_n}$ définies par les relations de récurrence suivantes

$$y_{i+1} = y_i + z_i\,\mathrm{d}t + f_i\,\frac{\mathrm{d}t^2}{2} \qquad \text{et} \qquad z_{i+1} = z_i + \left(\frac{f_i + f_{i+1}}{2}\right)\mathrm{d}t$$

en ayant pris le soin de poser $f_i = f(y_i)$ et $f_{i+1} = f(y_{i+1})$.

4. Écrire une fonction `positions_suivantes_verlet(m,pos,vit,dt)` qui prend en paramètres la liste `m` des masses des N corps du système étudié, la liste de leurs positions à l'instant $t_i$, la liste des vitesses au même instant et le pas d'intégration `dt`. La fonction doit renvoyer (et non modifier !) la liste des positions des N corps à l'instant $t_{i+1} = t_i + \mathrm{d}t$ calculées à l'aide du schéma de Verlet.
5. Écrire une fonction `etats_suivants_verlet(m,pos,vit,dt)` qui prend les mêmes paramètres que la fonction précédente mais doit renvoyer (et non modifier !) la liste des positions et la liste des vitesses des N corps à l'instant $t_{i+1}$ en utilisant le schéma de Verlet. Attention, certains tests vont contrôler la vitesse d'exécution pour s'assurer que l'algorithme global est au plus quadratique en N.

1. N'oubliez pas de diviser la force par la masse correspondante !

Partie III

## Fabrication d'un simulateur à N corps : simulation complète

6. À l'aide des listes `masses`, `positions` et `vitesses` prédéfinies dans le fichier à remplir, exécuter votre algorithme de $t = 0$ jusqu'à $t = 10$ par pas de temps `dt_euler=1e-6` pour Euler et `dt_verlet=1e-4` pour Verlet afin de produire deux graphiques (un pour chaque méthode d'intégration[2]) permettant de visualiser l'évolution des différentes particules dans le plan $(Oxy)$ sur l'intervalle de temps considéré. Ces graphiques seront enregistrés dans le répertoire courant sous le nom `integration_euler_VotreNom.png` et `integration_verlet_VotreNom.png` (où bien sûr vous remplacerez `VotreNom` par votre propre nom). Votre nom devra aussi figurer dans le titre des graphiques.

Pour vous guider, on peut voir dans le papier[3] de Szebehely, V., & Peters, C. F. (1967, Astronomical Journal, 72, 1187), le diagramme suivant qui correspond à la « bonne » résolution des équations demandées. Je vous laisse en déduire quelle est la méthode la plus précise entre Verlet et Euler.

**1188** V. SZEBEHELY AND C. F. PETERS

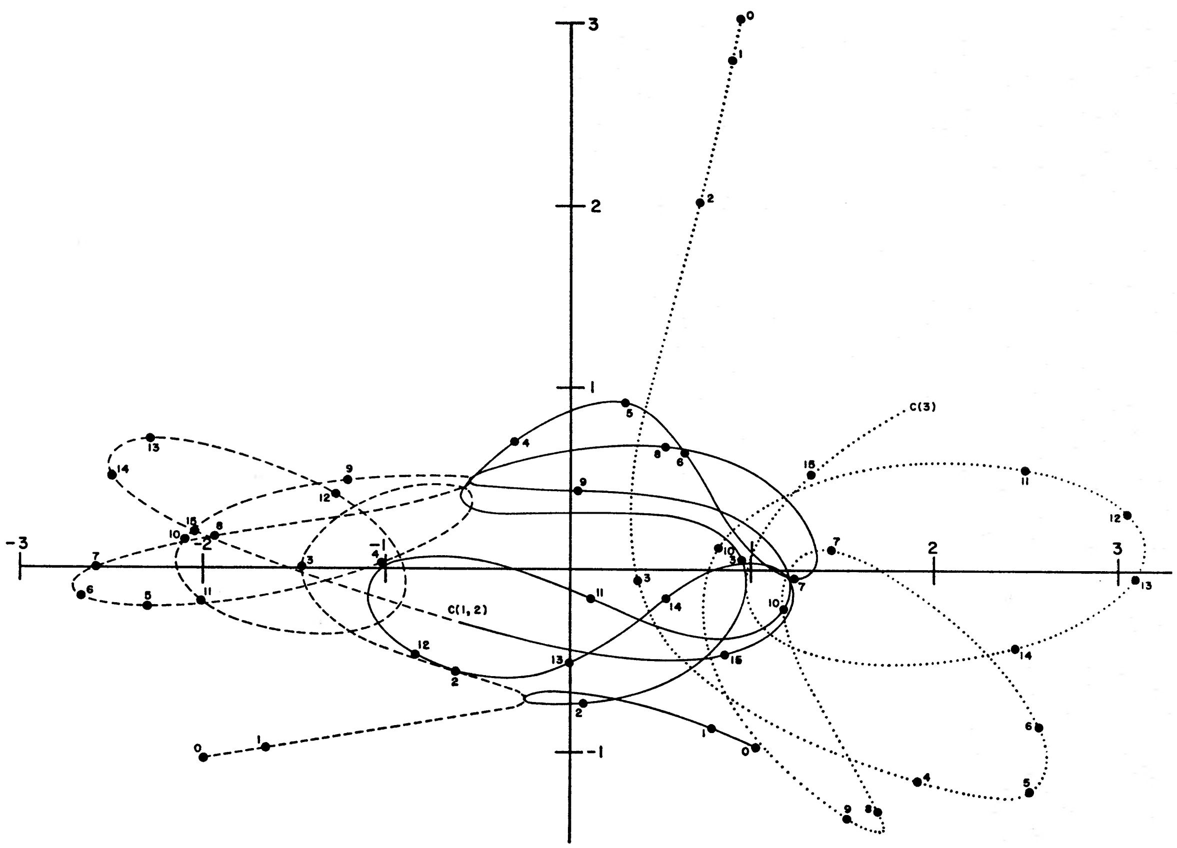



**Fig. 1. Periodic orbit with collision between bodies 1 and 2 at point C(1,2) and with simultaneous zero velocity of body 3 at point C(3).**

2. Pensez à factoriser votre code pour éviter toute duplication : ne faites pas de copier/coller !
3. Voir `http://adsabs.harvard.edu/abs/1967AJ.....72.1187S`