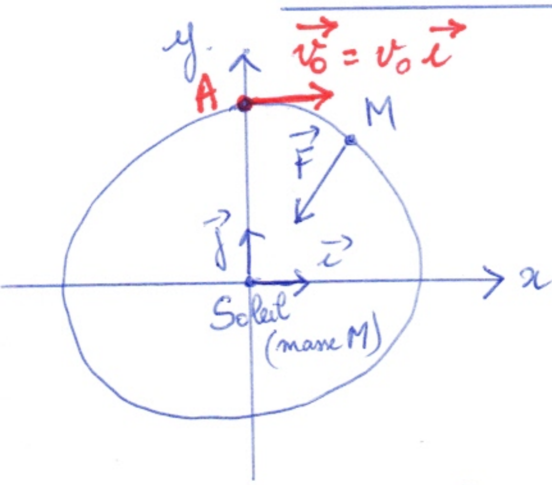
Mouvement d'un satellite avec python



<u>système</u> = { satellite de masse m }

<u>référentiel</u> héliocentrique supposé galiléen

<u>cinématique</u> : $\vec{OM} = x\,\vec{i} + y\,\vec{j}$

$$\vec{v} = \frac{d\vec{OM}}{dt} \qquad \vec{a} = \frac{d\vec{v}}{dt}$$

<u>Bilan des forces</u> : $\vec{F} = -G\,\dfrac{Mm}{r^3}\,\vec{r}$ avec $\vec{r} = \vec{OM} = \begin{pmatrix} x \\ y \end{pmatrix}$

<u>Principe fondamental de la dynamique</u> : $m\,\vec{a} = \vec{F} = -\dfrac{GMm}{r^3}\,\vec{r}$

$$\vec{a} = -G\frac{M}{r^3}\,\vec{r} \quad \text{soit} \quad \boxed{a_x = -\frac{GM}{r^3}\,x} \quad \boxed{a_y = -\frac{GM}{r^3}\,y}$$

On connaît a_x et a_y si on connaît la position initiale du satellite. On veut alors connaître v_x et v_y et x et y !

On forme le tableau python $w = [x, y, v_x, v_y]$

La dérivée par rapport au temps de ce tableau est $\dfrac{dw}{dt} = [v_x, v_y, a_x, a_y]$

On suppose connue une méthode d'intégration numérique : connaissant $v_x(0)$, elle permet de calculer $v_x(t)$. Celle que nous allons utiliser s'appelle scipy.integrate.odeint( )

ode : ordinary differential equation

nommée <u>équadiff</u>

Cette fonction a besoin de 3 paramètres : <u>une fonction</u> qui permet de calculer le tableau $\dfrac{dw}{dt}$ à partir du tableau w, <u>le vecteur w initial</u> (noté w0) et <u>un tableau t</u> contenant tous les instants où on souhaite connaître w.

<u>tableau w initial</u> : $\vec{OM}(t=0) = \vec{OA} = \begin{pmatrix} 0 \\ y0 \end{pmatrix}$ $\quad \vec{v_0} = \vec{v}(t=0) = \begin{pmatrix} v0 \\ 0 \end{pmatrix}$

on aura donc 2 paramètres : y0 et v0 en début de programme qui permettront de faire varier les conditions initiales.

w0 = [0, y0, v0, 0] en python.

tableau t : on utilise le module **numpy**

```
import numpy as np
t = np.linspace(0, tmax, N)
```

contient N valeurs régulièrement espacées dans l'intervalle [0, tmax].

fonction équadiff() : cette fonction permet de calculer le vecteur dérivé $\frac{dw}{dt}$ à partir du vecteur w.

$w = [x, y, v_x, v_y]$

indices : 0 1 2 3

```
def equadiff(w, t):
    vx = w[2]
    vy = w[3]
    C = 1        (valeur du produit GM)
    r = sqrt(w[0]**2 + w[1]**2)        (r = √(x² + y²))
    ax = -C/r**3 * w[0]
    ay = -C/r**3 * w[1]
    return np.array([vx, vy, ax, ay])
```