Mouvement d'un proton ou d'une particule électrisée dans un champ magnétique(suite)

15 juin 2012

Ce document est en quelque sorte la suite de :
http://pstricks.blogspot.fr/2012/05/mouvement-dune-particule-chargee-dans_13.html
Cette fois-ci, la vitesse initiale a une composante verticale.
La force de frottement proportionnelle à v :

$$\overrightarrow{f} = -h\overrightarrow{v}$$

Hypothèse : $\overrightarrow{v}_0$ est dans le plan Oxz et $\overrightarrow{B}$ est vertical.

$$\overrightarrow{B} = \begin{bmatrix} 0 \\ 0 \\ B \end{bmatrix}$$

On pose $\omega = \dfrac{qB}{m}$ et $\tau = \dfrac{m}{h}$.

La relation fondamentale de la dynamique appliquée à la particule donne :

$$\frac{\mathrm{d}\overrightarrow{v}}{\mathrm{d}t} = \frac{q}{m}\overrightarrow{B} \wedge \overrightarrow{v} - \frac{\overrightarrow{v}}{\tau} \tag{1}$$

$$\begin{cases} \ddot{x} &= \omega\dot{y} - \dfrac{\dot{x}}{\tau} \\ \ddot{y} &= -\omega\dot{x} - \dfrac{\dot{y}}{\tau} \\ \ddot{z} &= -\dfrac{\dot{z}}{\tau} \end{cases} \tag{2}$$

Nous avons un système de 3 équations différentielles couplées à 3 variables. La macro `\psplotDiffEqn` de PSTricks-add permet de tracer, par exemple, $y = f(x)$; $z = f(x)$ et $y = f(z)$. Mais elle ne permet pas d'obtenir un tableau de valeurs (x, y, z) pour tracer la trajectoire de la particule.

C'est pourquoi nous allons utiliser la package pst-eqdf, qui est une version simplifiée et adaptée de celle Dominique Rodriguez qui permet de sauver des tableaux postscript et/ou des fichiers de toutes les variables ainsi que de leurs dérivées. Les équations peuvent être écrites en mode algébrique.

```
% x    y    z    x'   y'   z'
% y[0] y[1] y[2] y[3] y[4] y[5]
\def\mvtB{y[3]|y[4]|y[5]|wc*y[4]-y[3]/tau|-wc*y[3]-y[4]/tau|-y[5]/tau}
\def\initCond{0 0 0 vx 0 vz}
```

Les conditions initiales seront précisées dans l'environnement `\begin{pspicture}...\end{pspicture}`. La sauvegarde sous forme de fichiers est facultative, c'est pourquoi on a placé, dans cet exemple, un % devant la ligne correspondante.

```
\pstVerb{/wc 5 def
        /tau 5 def
        /vx 20 def
        /vz 2 def}%
  \psequadiff[% x,y
  tabname=mvtXY,
  method=rk4,
```

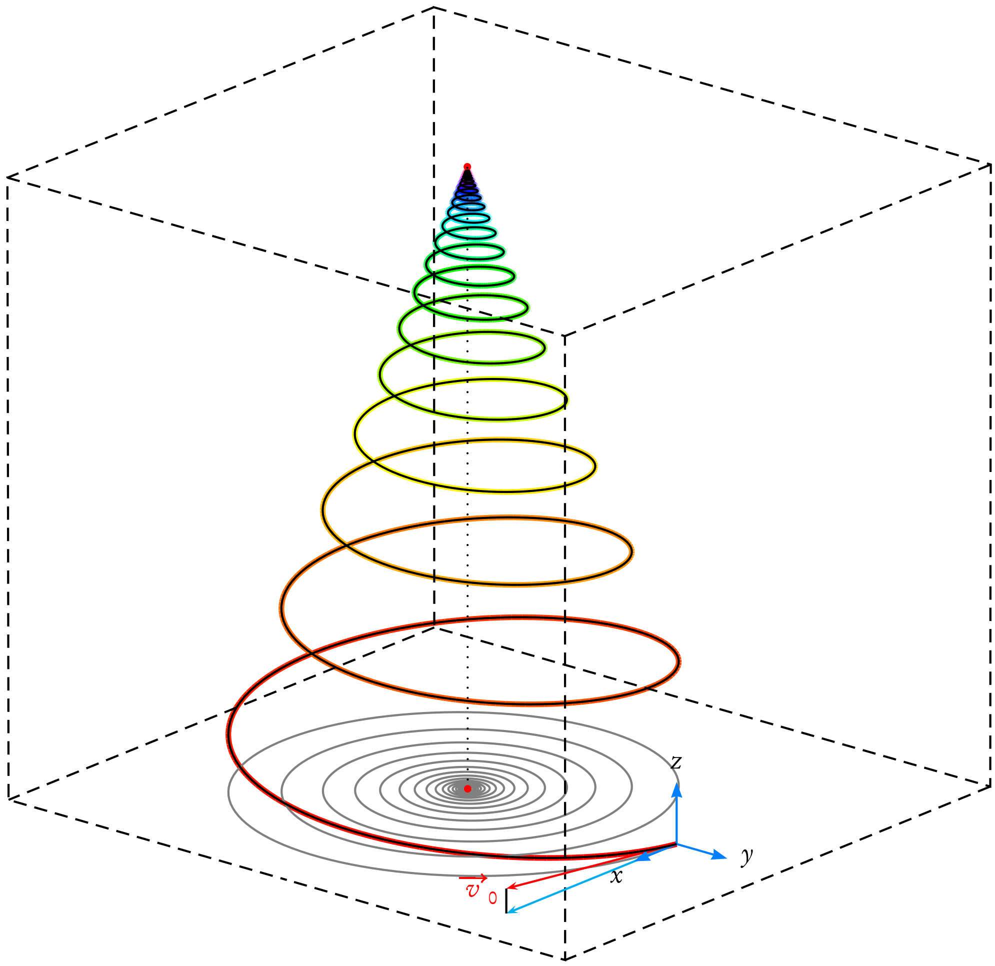
```
  whichabs=0,
  whichord=1,
%  saveData,filename=mvtXY.dat,
  plotpoints=2500,algebraic]{0}{25}{\initCond}{\mvtB}
  \psequadiff[% x,y
  tabname=mvtZX,
  method=rk4,
  whichabs=2,
  whichord=0,
%  saveData,filename=mvtZX.dat,
  plotpoints=2500,algebraic]{0}{25}{\initCond}{\mvtB}
\psset{THETA=40,PHI=20,Dobs=2000,Decran=1750}
\pnodeTroisD(0,0,0){O}
\pstVerb{
 /tabXYZ {
    0 2 mvtXY length 2 sub {/i exch def
     /abscisse mvtXY i get def
     /ordonnee mvtXY i 1 add get def
     /cote mvtZX i get def
     \formules
    }  for
    } def}%
\listplotHSB[unit=1,linewidth=0.075]{tabXYZ}
```

Il y a trois étapes amenant à la construction du tableau de 2500 points (x,y,z) :

1. Le calcul du tableau : tabname=mvtXY qui contient les couples (x,y).
2. Le calcul du tableau : tabname=mvtXZ qui contient les couples (x,z).
3. La concaténation des deux précédents en tabXYZ, pour les triplets (x,y,z).

Le tracé s'effectue avec la commande `\listplotHSB[unit=1,linewidth=0.075]{tabXYZ}`.

`\listplotHSB` fait partie du package pst-plothsb.



Ce système d'équations peut se résoudre algébriquement, ce qui permettra de comparer le tracé obtenu par la méthode numérique qui utilise Rung-Kutta-4 et celui obtenu avec les expressions exactes.

Les solutions de ce système d'équations différentielles sont :

$$\left\{\begin{array}{lcl} x & = & \tau^* v_{0_x}\left[\cos\varphi - \mathrm{e}^{-\frac{t}{\tau}}\cos(\omega t+\varphi)\right] \\ y & = & \tau^* v_{0_x}\left[-\sin\varphi + \mathrm{e}^{-\frac{t}{\tau}}\sin(\omega t+\varphi)\right] \\ z & = & \tau v_{0_z}\left(1-\mathrm{e}^{-\frac{t}{\tau}}\right) \end{array}\right. \tag{3}$$

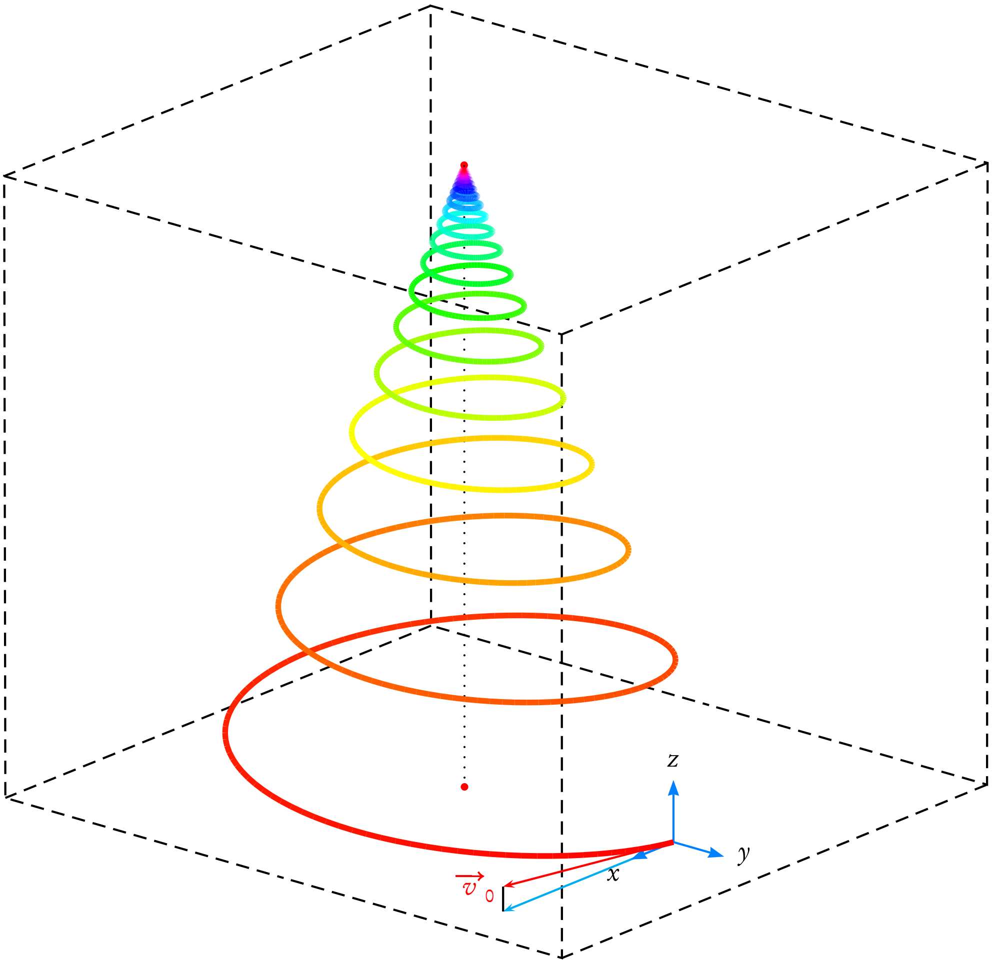
En posant $\dfrac{1}{\tau^{*2}} = \dfrac{1}{\tau^2} + \omega^2$, et $\tan\varphi = \tau\omega$. Si vous souhaitez avoir des précisions sur une méthode de résolution du système, vous pourrez lire le remarquable document de Joël SORNETTE sur ce sujet :

`http://www.joelsornette.fr/physique/PC/ressources/exotypes/exotype40.pdf`

Pour le tracé en 3D de la courbe ainsi paramètrée, j'utilise la macro `\parametricplotIIIDHSB` qui fait partie du package pst-plothsb. On définit algébriquement la fonctions :

```
\def\mvbxyz{TAU*vx*(cos(Phi)-Ex(-t/tau)*cos(wc*t+Phi))|% x(t)
TAU*vx*(-sin(Phi)+Ex(-t/tau)*sin(wc*t+Phi))|% y(t)
 tau*vz*(1-Ex(-t/tau))}% z(t)
\parametricplotIIIDHSB[algebraic,plotpoints=2000,linewidth=0.075]{0}{25}{\mvbxyz}
```

La première courbe était tracée aux couleurs de l'arc-en-ciel et cette dernière en noir d'un trait un peu plus fin. On peut constater l'excellence de la méthode numérique : difficile de distinguer une différence entre les deux tracés.


z
y
x
$\vec{v}_0$