Rendez-vous Spatiale

1 Dérivé de Gateau $J'(u, \delta u)$

— Déterminer la dérivé de Gateau de la fonction $x(u)$. En déduire la dérivé de Gateau de J.

$$\begin{aligned}
x'(u, \delta u) &= \lim_{\lambda \to 0^+} \frac{x(u + \lambda \delta u) - f(u)}{\lambda} \\
&= \lim_{\lambda \to 0^+} \frac{Ax(u + \lambda \delta u, t) + B(u(t) + \lambda \delta u) - Ax(u, t) - Bu(t)}{\lambda} \\
&= \lim_{\lambda \to 0^+} \frac{Ax(u + \lambda \delta u, t) + B\lambda \delta u - Ax(u, t)}{\lambda} \\
&= B\delta u + \lim_{\lambda \to 0^+} \frac{Ax(u + \lambda \delta u, t) - Ax(u, t)}{\lambda} \\
&= B\delta u + Ax'(u, \delta u)
\end{aligned}$$

On en déduit la dérivé de Gateau de J.

$$\begin{aligned}
J'(u, \delta u) &= \lim_{\lambda \to 0^+} \frac{J(u + \lambda \delta u) - J(u)}{\lambda} \\
&= \lim_{\lambda \to 0^+} \overbrace{\frac{\int_0^T \frac{\varepsilon}{2}(u(s) + \lambda \delta u)^2 ds - \int_0^T \frac{\varepsilon}{2} u(s)^2 ds}{\lambda}}^{C1} + \overbrace{\frac{\frac{1}{2}x(u + \lambda \delta u, T)^2 - \frac{1}{2}x(u, T)^2}{\lambda}}^{C2}
\end{aligned}$$

On va maintenant calculer C1 et C2.

$$\begin{aligned}
C1 &= \lim_{\lambda \to 0^+} \frac{\int_0^T \frac{\varepsilon}{2}[(u(s) + \lambda \delta u)^2 - u(s)^2] ds}{\lambda} \\
&= \lim_{\lambda \to 0^+} \frac{\int_0^T \frac{\varepsilon}{2}[2\lambda u(s)\delta u + (\lambda \delta u)^2] ds}{\lambda} \\
&= \lim_{\lambda \to 0^+} \int_0^T \frac{\varepsilon}{2}[2u(s)\delta u + \lambda \delta u^2] ds
\end{aligned}$$

on a $\lim_{\lambda \to 0^+} \lambda \delta u^2 = 0$, donc

$$C1 = \int_0^T \varepsilon u(s) \delta u ds$$

Au tour de C2.

$$\begin{aligned}
C2 &= \lim_{\lambda \to 0^+} \frac{1}{2} \frac{x(u + \lambda \delta u, T)^2 - x(u, T)^2}{\lambda} \\
&= \lim_{\lambda \to 0^+} \frac{1}{2} \frac{x(u + \lambda \delta u, T) - x(u, T)}{\lambda} (x(u + \lambda \delta u, T) + x(u, T)) \\
&= \lim_{\lambda \to 0^+} \frac{1}{2} x'(u, \delta u)(T)(x(u + \lambda \delta u, T) + x(u, T))
\end{aligned}$$

on a $\lim_{\lambda \to 0^+} x(u + \lambda \delta u, T) = x(u, T)$, donc

$$\begin{aligned} C2 =& \frac{1}{2} x'(u, \delta u)(T) 2x(u, T) \\ =& x'(u, \delta u)(T) x(u, T) \end{aligned}$$

$$\begin{aligned} J'(u, \delta u) =& C1 + C2 \\ =& \int_0^T \varepsilon u(s) \delta u ds + x'(u, \delta u)(T) x(u, T) \end{aligned}$$

— Déterminer la fonction $g(t)$
On a besoin de Taylor.

$$J(u + \lambda \delta u) = J(u) + \lambda(\nabla J(u).\delta u) + o(||\delta u||)$$

On repart de la dérivé de Gateau.

$$\begin{aligned} J'(u, \delta u) &= \lim_{\lambda \to 0^+} \frac{J(u + \lambda \delta u) - J(u)}{\lambda} \\ &= \lim_{\lambda \to 0^+} \frac{J(u) + \lambda(\nabla J(u).\delta u) + o(||\delta u||) - J(u)}{\lambda} \\ &= \lim_{\lambda \to 0^+} (\nabla J(u).\delta u) + o(||\delta u||) \\ &= \nabla J(u).\delta u + (||\delta u|| \\ &= \int_0^T \nabla J(u(t)) \delta u dt \end{aligned}$$

Cela nous donne

$$g(t) = \nabla J(u(t))$$

2 Calcul de ∇J

Montrer que :

$$\nabla J(u) = \varepsilon u + B^T p.$$

$$\begin{aligned} \int_0^T \dot{p}(t) x'(u, \delta u, t) dt &= [x'(u, \delta u, t) p(t)]_0^T - \int_0^T p(t) \dot{x}'(u, \delta u, t) dt \\ &= x'(u, \delta u, T) p(T) - x'(u, \delta u, 0) p(0) - \int_0^T p(t) \dot{x}'(u, \delta u, t) dt \\ &= x'(u, \delta u, T) p(T) - \int_0^T p(t) \dot{x}'(u, \delta u, t) dt \end{aligned}$$

L'équation différencielle rétrograde nous donne :

$$\begin{aligned}
\dot{p}(t) &= -{}^t A p(t) \\
\int_0^T \dot{p}(t) x'(u, \delta u, t) dt &= -\int_0^T {}^t A p(t) x'(u, \delta u, t) dt \\
x'(u, \delta u, T) p(T) - \int_0^T p(t) \dot{x}'(u, \delta u, t) dt &= -\int_0^T {}^t A p(t) x'(u, \delta u, t) dt \\
x'(u, \delta u, T) p(T) &= -\int_0^T {}^t A p(t) x'(u, \delta u, t) dt + \int_0^T p(t) \dot{x}'(u, \delta u, t) dt \\
&= \int_0^T -{}^t A p(t) x'(u, \delta u, t) + p(t)[A x'(u, \delta u, t) + B \delta u] dt \\
&= \int_0^T -{}^t A p(t) x'(u, \delta u, t) + p(t) A x'(u, \delta u, t) + p(t) B \delta u dt \\
&= \int_0^T p(t)[(A - A) x'(u, \delta u, t)] + p(t) B \delta u dt \\
&= \int_0^T p(t) B \delta u dt \\
&= \int_0^T B^t p(t) \delta u dt
\end{aligned}$$

On regroupes les résultats précédents (et on utilise la linéaritée des intégrales) :

$$\begin{aligned}
J'(u, \delta u) &= \int_0^T \nabla J(u(t)) \delta u dt \\
\int_0^T \nabla J(u(t)) \delta u dt &= \int_0^T \varepsilon u(t) \delta u dt + x'(u, \delta u)(T) x(u, T) \\
\int_0^T \nabla J(u(t)) \delta u dt &= \int_0^T \varepsilon u(t) \delta u dt + \int_0^T B^t p(t) \delta u dt \\
\int_0^T \nabla J(u(t)) \delta u dt &= \int_0^T \varepsilon u(t) \delta u + B^t p(t) \delta u dt \\
0 &= \int_0^T \nabla J(u(t)) \delta u dt - \int_0^T \varepsilon u(t) \delta u + B^t p(t) \delta u dt \\
0 &= \int_0^T \delta u [\nabla J(u(t)) - (\varepsilon u(t) + B^t p(t))] dt
\end{aligned}$$

Comme on a fait l'identification par magie avec

$$\int_0^T \nabla J(u(t)) \delta u dt = \int_0^T g(t) \delta u dt$$

nous donne

$$g(t) = \nabla J(u(t))$$

et comme

$$\nabla J(u).\delta u + o(||\delta u||) = \nabla J(u).\delta u$$

On admet

$$\nabla J(u) = \varepsilon u + B^T p.$$

3 Résolution numérique

Pour la résolution numérique, nous avons mis en place un programme en Python, en utilisant la bibliothèque Numpy, pour la manipulation de matrices. On utilise la méthode du gradient, dont chaque itération se décompose, dans notre cas, en 4 étapes :

- 1) résoudre : $\begin{cases} \dot{x}(t) = Ax(t) + Bu(t) \\ x(0) = x_0 \end{cases}$
- 2) résoudre : $\begin{cases} \dot{p}(t) = A^t p(t) \\ p(t) = x(t) \end{cases}$
- 3) calculer : $\nabla J(u(t)) = \varepsilon u(t) + B^t p(t)$
- 4) calculer : $\nabla u_{n+1} = u_n - p\nabla J(u_n)$

— Pour résoudre 1, on utilise la méthode d'euler améliorée.
— Pour résoudre 2, on utilise la même méthode, mais, en quelques sortes, dans le sens opposé.
— L'étape 3 ne pose pas de problème
— L'étape 4 nous donne une nouvelle approximation de $u(t)$, à utiliser pour la prochaine itération, dans l'étape 1.

On itère la méthode du gradient un nombre fixé de fois (pas de condition d'arrêt).

4 Résultat

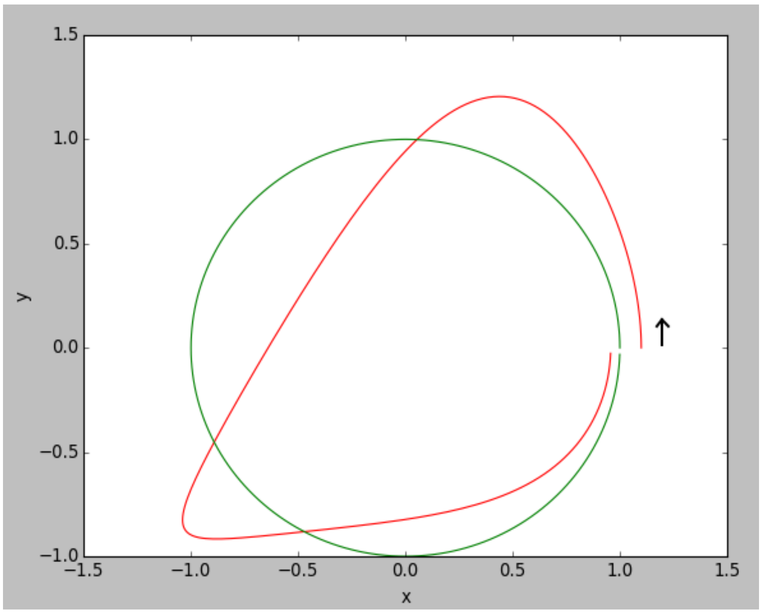

FIGURE 1 – Fusée en rouge – ISS en vert

Donnée pour la $Figure 1$
$T = 1$
$pas(méthode d'Euler) = 0.005$
$Nombre d'itération : 500$
$x_1(0) = 0.1$
$x_2(0) = 0$

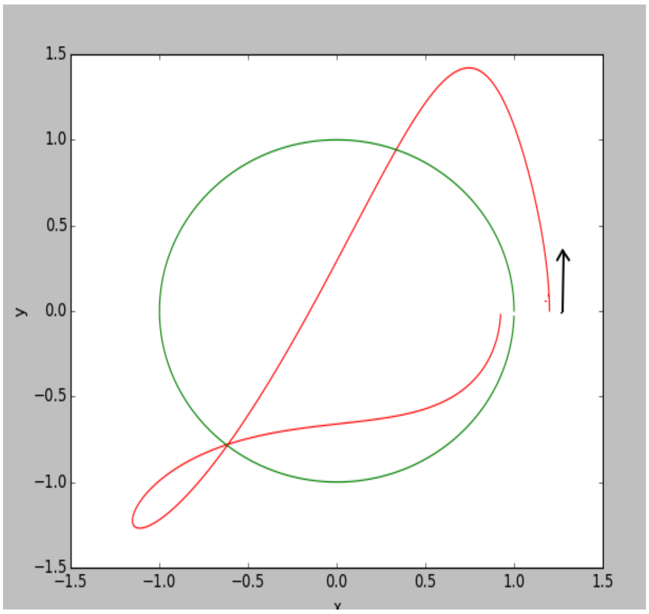

FIGURE 2 – Fusée en rouge – ISS en vert

Donnée pour la $Figure 2$
$T = 1$
$pas(méthoded'Euler) = 0.005$
$Nombred'itération : 5000$
$x_1(0) = 0.2$
$x_2(0) = 0$

5 programme

Pour tester la résolution avec notre programme, ouvrez ui2.py avec un éditeur de texte, choisissez les paramètres que vous voulez (l.34-40 du programme), puis lancer $python3$ $ui2.py$ pour executer le programme.

Un graphe apparaitra alors, c'est la trajectoire la fusée (en rouge) et celle de l'ISS (en vert), dans le référentiel de la terre. En fermant cette fenètre, vous aurez accès a une interface vous permettant de visualiser le parcours des deux objets en fonction du temps :
Utilisez le slider en bas de la fenetre pour faire avancer ou reculer le temps. En rouge, la fusée, en vert, l'ISS.
La résolution du problème est effectuée par le code aux lignes 103-136 de $ui2.py$.

Dépendances :
* Tkinter
* numpy
* matplotlib

pour tout installer : sudo apt-get install python3-tk python3-numpy python3-matplotlib