Feuille de TP Python 07

Résolutions d'EDO, Energie et solutions périodiques

1 Introduction

L'objet de ce TP est de prolonger le travail entrepris sur les résolutions d'équations différentielles et porte notamment sur les systèmes possèdant une intégrale première, c'est à dire pour lesquelles existe une quantité conservée pendant le mouvement.

On reprend le code élaboré dans la dernière partie du TP précédent dans le module `edo.py` dont la dernière version est présente dans l'archive jointe au TP.

La fonction intéressante–implémentant une méthode d'EULER– est `my_odeint`. Sa signature est identique celle de la fonction `odeint` du module `scipy.integrate`.

Vous pouvez exécuter le fichier `edo.py` pour voir une démonstration de résolution de problème de CAUCHY scalaire d'ordre 1 avec graphes comparatifs de la vraie solution, des solutions approchées.

2 Premier exemple : le pendule pesant

2.1 Description du problème et système d'équations

On considère le système mécanique décrit en Fig. 1 composé d'une masse m suspendue à un fil rigide de longueur ℓ. Les variable d'intérêt est θ, l'angle que fait le fil avec la verticale et l'on cherche à décrire l'évolution de θ en fonction du temps t sachant qu'à un instant $t = t_0$ (on peut prendre $t_0 = 0$ pour fixer les idées) on a pour conditions initiales (position et vitesse angulaire initiales) :

$$\theta_{|t=t_0} = \theta_0, \quad \frac{d\theta}{dt}_{|t=t_0} = \omega_0$$

La masse étant soumise à son poids et à la tension du fil rigide, le principe fondamental de la dynamique habilement projeté sur la direction tangentielle au mouvement se traduit en l'équation différentielle du second ordre :

$$\frac{d^2\theta}{dt^2} + \frac{g}{\ell}\sin\theta = 0 \qquad (\star)$$

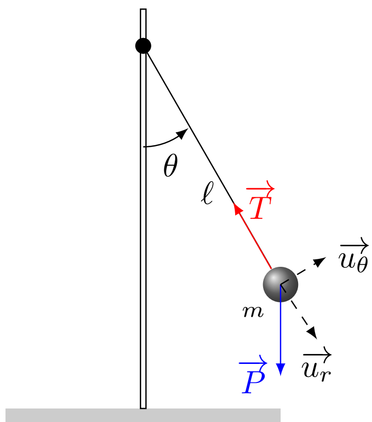

FIGURE 1 – Le pendule pesant

Le changement de variable temporelle $t = \sqrt{\frac{\ell}{g}}.\tau$ permet de toujours se ramener à l'équation « réduite » pour laquelle $\frac{g}{\ell} = 1$.

En posant $\omega = \frac{d\theta}{dt}$, la vitesse angulaire et $X = \begin{pmatrix}\theta\\ \omega\end{pmatrix}, F(X) = F\begin{pmatrix}\theta\\ \omega\end{pmatrix} = \begin{pmatrix}\omega\\ -\frac{g}{\ell}\sin\theta\end{pmatrix}$, on transforme l'équation $(\star)$ du second ordre en une équation vectorielle (un système d'équations scalaires) du premier ordre :

$$\frac{dX}{dt} = F(X),$$

L'équation $(\star)$ mise sous cette forme, avec des conditions initiales adéquates, peut être résolue approximativement par un schéma d'EULER ou tout autre méthode plus sophistiquée..

2.2 Energie conservée, lignes de niveau

Si on pose, pour $X = (\theta, \omega)$ quelconque :

$$E(X) = E(\theta,\omega) = \frac{1}{2}\omega^2 + \frac{g}{\ell}(1-\cos(\theta)) \qquad \text{(E)}$$

On démontre, pour $t \mapsto X(t)$, une solution de $(\star)$ sur un intervalle temporel I, que la fonction $t \mapsto E(X(t))$ est constante sur I. A un facteur constant $m.\ell^2$ près, il s'agit bien entendu de l'énergie mécanique totale de ce système.

La trajectoire de la courbe $t \mapsto X(t) = (\theta(t), \omega(t))$ est donc contenue dans une courbe de niveau de la fonction E, la courbe de niveau correspondant au niveau initial $E_0 = E(\theta_0, \omega_0)$.

La formule de l'énergie est réglée de sorte que cette énergie soit toujours positive et que la position d'équilibre stable $(\theta_*, \omega_*) = (0,0)$ soit d'énergie $E(0,0) = 0$.

La position d'équilibre instable est $(\theta^*, \omega^*) = (\pi, 0)$ et l'énergie de cette position est $E^* = 2\frac{g}{\ell}$.

2.3 Détermination de la période et non isochronie

Lorsque l'énergie à l'instant initial est non nulle mais inférieure à $E^* = 2\frac{g}{\ell}$, le pendule admet une période (ce qui sert à faire des pendules comtoises) qui dépend de θ_0 et ω_0, position et vitesse à l'instant initial. En fait, en fixant qu'à l'instant $t_0 = 0$, le pendule est en position extremale $\theta_0 = \theta_{max} \in]-\pi, +\pi[\setminus\{0\}$ avec $\omega_0 = 0$, son énergie sera automatiquement strictement positive et inférieure à E^* et donc le pendule oscillera entre les deux angles extremaux θ_{max} et $-\theta_{max}$.

Notre but est de déterminer comment la période T du pendule dépend de θ_{max}.

La méthode est la suivante : on dispose d'un calcul approché de la solution (θ, ω) sur un intervalle discrétisé $[t_0, t_N[= [0, t_N[$ de $\theta_0 = \theta_{max} \in]0, +\pi[$, $\omega_0 = 0$. ? On recherche le premier instant t_i, $i \geq 1$ de cet intervalle tel que $\theta(t_{i-1}) \leq \theta_{max}$ et $\theta(t_i) > \theta_{max}$. La valeur $T = \frac{t_{i-1}+t_i}{2}$ est une bonne approximation de la période cherchée.

On cherchera alors à grapher la fonction $\theta_{max} \mapsto T$. On pourra remarquer que cette fonction est strictement croissante, qu'il y a tout lieu de croire que sa limite en 0^+ soit $\sqrt{\frac{\ell}{g}}2\pi$ (approximation aux petits angles) et que sa limite en π^- soit $+\infty$ (lorsque $E = E^*$, la période est « infinie », quoique ça veuille dire).

Le système n'est pas isochrone : sa période dépend des conditions initiales.

2.4 Travail demandé

On utilisera le canevas `td07-pendule.py` pour accélérer le travail. L'entête permet de choisir (en commentant/décommentant) si l'on utilise la fonction `my_odeint` du script `edo.py` sous l'alias `odeint` ou la fonction `odeint` du module `scipy.integrate`.

Ce changement de fonction de résolution est à faire notamment pour observer sur la qualité de la conservation de l'énergie.

1. Après lecture du §2.1, mettre en place une fonction `pendule(theta0,omega0,t,g_sur_ell = 1)` qui retourne, en `ndarray` de shape `(t.shape[0],2)` le couple (θ, ω), solution de $(\star)$ avec condition initiale $\theta_0 =$ `theta0`, $\omega_0 =$ `omega0`, calculé de manière approchée en les instants décrits par le `ndarray t`.
   Le coefficient $\frac{g}{\ell}$ est spécifié en argument nommé avec valeur par défaut.
2. Pour différentes valeurs de conditions initiales et une plage de temps `t` suffisamment longue et finement échantillonée, calculer le couple (θ, ω), puis
   (a) grapher (sur une seul dessin) ces fonctions en fonction de t. On pourra utiliser deux axes d'ordonnées à l'aide de `plt.twinx`.
   (b) grapher la courbe paramétrée $t \mapsto (\theta(t), \omega(t))$ (sur un autre dessin)
   On pourra créer une fonction qui fait ce double travail (NB : utiliser `plt.subplots()` pour initialiser ces graphiques et de ne faire `plt.show()` qu'une fois TOUS les dessins composés.
3. Après lecture du §2.2,
   (a) Définir une fonction `E(theta,omega)` qui retourne $E(\theta, \omega)$. Cette fonction doit être Numpy compatible au sens où, si `theta,omega` sont deux `ndarray` de même forme, `E` retourne un `ndarray` de même forme contenant l'ensemble des valeurs de E calculées sur les couples (θ, ω).
   (b) Compléter les fonctions graphiques précédemment développées pour que :
      i. le graphe de $t \mapsto E(\theta(t), \omega(t))$ apparaisse en parallèle des graphes de θ et ω ;
      ii. des lignes de niveau de E pertinentes apparaissent sur le dessin comportant la trajectoire $t \mapsto (\theta(t), \omega(t))$. On pourra utiliser la fonction `LignesDeNiveau` définie dans `edo.py` après avoir consulté [1] sa `docstring`.
4. Après lecture du §2.3 :
   (a) vérifier que la fonction `Periode(t,x)` écrite ci-dessous calcule bien la période suivant la recette indiquée ;

```
#%% Q.4
def Periode(t,x):
    x0 = x[0]
    i = 1
    while i <t.shape[0] - 1 and x[i] <= x0 :
        i += 1 ;
    return (t[i-1]+t[i])*0.5
```

   (b) utiliser la fonction `Periode(t,x)` pour tracer le graphe de la période T en fonction de θ_{max}.
5. Si on a le temps : Ajouter des « frottements », c'est à dire ajouter un terme en $+R.\frac{d\theta}{dt}$ à l'équation $(\star)$ (et donc un paramètre `R = 0.0` à la fonction `pendule`) et observer ce qui se passe au niveau des variations de l'énergie, variation de l'amplitude maximale.

1. via `help(LignesDeNiveau)`

3 Deuxième exemple : le système proies-prédateurs de LOTKA–VOLTERRA

3.1 Description du problème et système d'équations

En Amazonie coexistent des groupes d'Unaus [2] et de Vautours. Même si les vautours sont essentiellement des charognards, il leur arrive de chasser (et manger) des unaus vivants. Nommons V le nombre de vautours et U le nombre d'unaus vivants à un instant t sur un territoire délimité.

On suppose que ces populations sont régies par le système de LOTKA–VOLTERRA :

$$\frac{dU}{dt} = (r_u - k_u.V).U,\ \frac{dV}{dt} = (-r_v + k_v.U).V \tag{$\star$}$$

où t, qui désigne la variable « temporelle », varie dans un intervalle de $\mathbb{R}$, non trivial.

Les nombres réels r_u, r_v (constantes de temps), et k_u, k_v (inverses de « capacités ») sont tous > 0, indépendants de t.

On rappelle que ce système décrit une évolution vue d'un point de vue macroscopique, en particulier, il ne faut surtout pas imaginer que U et V–effectifs de populations– ne prennent que des valeurs entières mais plutôt voir en U et V des valeurs moyennes de ces quantités observables sur un grand nombre d'instances de systèmes réels régis par les mêmes règles stochastiques de probabilités de rencontre.

On pose $U^* = \frac{r_u}{k_u}$, $V^* = \frac{r_h}{k_h}$. Le point (U^*, V^*) est point d'équilibre du système différentiel au sens où la fonction constante sur $\mathbb{R}$, $t \mapsto (U(t), V(t)) = (U^*, V^*)$ est solution de $(\star)$.

Ce système admet d'autres solutions évidentes :

1. le couple (U, V) défini par $\forall t \in \mathbb{R}$, $U(t) = 0$, $V(t) = V(0)e^{-r_h.t}$;
2. le couple (U, V) défini par $\forall t \in \mathbb{R}$, $V(t) = 0$, $U(t) = U(0)e^{+r_u.t}$;
3. le couple constant $t \mapsto (0, 0)$.

Il y a un principe d'unicité (et on l'admet ici !) affirmant qu'étant données deux solutions (U_1, V_1), (U_2, V_2), deux instants t_1, t_2 tels que

$$U_1(t_1) = U_2(t_2),\ V_1(t_1) = V_2(t_2)$$

alors, pour tout τ,

$$U_1(t_1 + \tau) = U_2(t_2 + \tau),\ V_1(t_1 + \tau) = V_2(t_2 + \tau).$$

On en déduit, avec la connaissance des solutions évidentes précédemment décrites, que la trajectoire d'une solution (U, V) telle que, à un certain instant t_0, $U(t_0), V(t_0) > 0$, ne coupe jamais les axes de coordonnées. De même, une telle solution, sauf à être la solution constante (U^*, V^*) , ne passe jamais par le point (U^*, V^*).

Notons enfin que le changement de variable (fonctions inconnues, c'est à dire les variables « dépendantes » dans le langage de la modélisation) :

$$u = \frac{U}{U^*},\ v = \frac{V}{V^*}$$

permet de toujours se ramener à l'équation « réduite » :

$$\frac{du}{dt} = +r_u(1 - v).u,\ \frac{dv}{dt} = -r_v.(1 - u).v. \tag{$\star\star$}$$

Ce système admet pour point d'équilibre le point $(u^*, v^*) = (1, 1)$, correspondant à (U^*, V^*) via le changement de variable dépendantes.

En posant $X = \begin{pmatrix} u \\ v \end{pmatrix}, F(X) = F\begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} +r_u(1 - v).u \\ -r_v.(1 - u).v \end{pmatrix}$, on réécrit le système d'équations scalaires $(\star\star)$ en une équation vectorielle du premier ordre :

$$\frac{dX}{dt} = F(X).$$

L'équation $(\star\star)$ mise sous cette forme, avec des conditions initiales adéquates, peut être résolue approximativement par un schéma d'EULER ou tout autre méthode plus sophistiquée..

3.2 Energie conservée, lignes de niveau

Si on pose, pour $X = (u, v)$ quelconque :

$$E(X) = E(u, v) = r_v(u - 1 - \ln u) + r_u(v - 1 - \ln v) \tag{E}$$

On démontre, pour $t \mapsto X(t)$, une solution de $(\star\star)$ sur un intervalle temporel I, que la fonction $t \mapsto E(X(t))$ est constante sur I. Par analogie avec la mécanique et ses lois de conservation, on nomme E l'« énergie » de ce système.

La trajectoire de la courbe $t \mapsto X(t) = (u(t), v(t))$ est donc contenue dans une courbe de niveau de la fonction E, la courbe de niveau correspondant au niveau initial $E_0 = E(u_0, v_0)$.

La formule de l'énergie est réglée (constante) de sorte que cette énergie soit toujours positive et que la position d'équilibre $(u^*, v^*) = (1, 1)$ soit d'énergie $E(1, 1) = 0$.

Noter que lorsque (u, v) s'approche d'un axe de coordonnées (*i.e.* soit $u \to 0$, soit $v \to 0$), $E(u, v) \to +\infty$.

2. *Choloepus hoffmanni*, un genre d'Aï, le pluriel est vérifié.

3.3 Détermination de la période

Lorsque l'énergie à l'instant initial est non nulle, une solution du système admet une période qui dépend de u_0 et v_0, populations à l'instant initial.

En fait, en fixant qu'à l'instant $t_0 = 0$, la population (normalisée) de vautours est $v(t_0) = v^* = 1$ et la population d'unaus est $u(t_0) > u^* = 1$, la population d'unaus est maximale $u_{max} = u_0$ et ne peut que décroitre ensuite vers une valeur u_{min}, population d'unaus minimale pour laquelle la population de vautours vaut de nouveau $v = v^* = 1$ après être passée par une phase de croissance puis de décroissance. Arrivée à la valeur minimale u_{min}, la population d'unaus reprendra sa croissance jusqu'à la valeur maximale u_{max} et ainsi de suite. La durée d'un aller-retour de u_{max} à u_{max} est la période T de la solution.

La population d'unaus oscillera donc entre deux extrêmes u_{max} et u_{min}, l'autre valeur de u vérifiant $E(u, v^*) = E(u_{max}, v^*)$.

Notre but est de déterminer comment la période T du pendule dépend de $u_{\max}$.

La méthode est la suivante : on dispose d'un calcul approché de la solution (u, v) sur un intervalle discrétisé $[t_0, t_N[= [0, t_N[$ avec $u_0 = u_{max} \in]1, +\infty[$, $v_0 = 1$. On recherche le premier instant t_i, $i \geq 1$ de cet intervalle tel que $u(t_{i-1}) \leq u_{max}$ et $u(t_i) > u_{max}$. La valeur $T = \frac{t_{i-1}+t_i}{2}$ est une bonne approximation de la période cherchée.

On cherchera alors à grapher la fonction $u_{max} \mapsto T$. Est-elle croissante ? A-t-elle d'autres propriétés ? Voilà des questions auxquelles le graphe cherché peut apporter des éléments de réponse.

3.4 Travail demandé

On utilisera le canevas `td07-LV.py` pour accélérer le travail. L'entête permet de choisir (en commentant/décommentant) si l'on utilise la fonction `my_odeint` du script `edo.py` sous l'alias `odeint` ou la fonction `odeint` du module `scipy.integrate`.

Ce changement de fonction de résolution est à faire notamment pour observer sur la qualité de la conservation de l'énergie.

1. Après lecture du §3.1, mettre en place une fonction `LV(u0,v0,t,ru = 1.0, rv = 1.0)` qui retourne, en `ndarray` de shape `(t.shape[0],2)` le couple (u, v), solution de $(\star\star)$ avec condition initiale $u_0 =$ `u0`, $v_0 =$ `v0`, calculé de manière approchée en les instants décrits par le `ndarray t`.

   Les coefficients r_u, r_v sont spécifiés en argument nommés avec valeurs par défaut.
2. Pour différentes valeurs de conditions initiales et une plage de temps `t` suffisamment longue et finement échantillonée, calculer le couple (u, v), puis
   (a) grapher (sur une seul dessin) ces fonctions en fonction de t.
   (b) grapher la courbe paramétrée $t \mapsto (u(t), v(t))$ (sur un autre dessin)

   On pourra créer une fonction qui fait ce double travail (NB : utiliser `plt.subplots()` pour initialiser ces graphiques et de ne faire `plt.show()` qu'une fois TOUS les dessins composés.
3. Après lecture du §3.2,
   (a) Définir une fonction `E(u,v,ru = 1.0, rv = 1.0)` qui retourne $E(u, v)$. Cette fonction doit être Numpy compatible au sens où, si `u,v` sont deux `ndarray` de même forme, `E` retourne un `ndarray` de même forme contenant l'ensemble des valeurs de E calculées sur les couples (u, v).
   (b) Compléter les fonctions graphiques précédemment développées pour que :
      i. le graphe de $t \mapsto E(u(t), v(t))$ apparaisse en parallèle des graphes de u et v ; On pourra utiliser deux axes d'ordonnées à l'aide de `plt.twinx`.
      ii. des lignes de niveau de E pertinentes apparaissent sur le dessin comportant la trajectoire $t \mapsto (u(t), v(t))$. On pourra utiliser la fonction `LignesDeNiveau` définie dans `edo.py` après avoir consulté [3] sa `docstring`.
4. Après lecture du §3.3 :
   (a) vérifier que la fonction `Periode(t,x)` écrite ci-dessous calcule bien la période suivant la recette indiquée ;

```
#%% Q.4
def Periode(t,x):
    x0 = x[0]
    i = 1
    while i <t.shape[0] - 1 and x[i] <= x0 :
        i += 1 ;
    return (t[i-1]+t[i])*0.5
```

   (b) utiliser la fonction `Periode(t,x)` pour tracer le graphe de la période T en fonction de u_0 (on fixe $v_0 = v^* = 1$ et on fait varier u_0 sur $[1, +\infty[$).
5. Si on a le temps : Ajouter des Wapitis, des animaux qui, périodiquement reviennent, font s'envoler une partie des vautours et contribuent avec un terme en $-w(t).v(t)$ à la deuxième équation du système $(\star\star)$. Observer ce qui se passe au niveau des variations de l'énergie ?

   On pourra prendre $w(t) = \cos^2(t)$, histoire d'avoir une fonction périodique positive.

3. via `help(LignesDeNiveau)`