Étude de fonctions, équations différentielles
TP3 – Résolutions approchées d'équations différentielles

Sur cette feuille on va étudier le modèle de Lotka-Volterra d'évolution des populations de deux espèces animales, dont une de prédateurs, et une de proie (par exemples des requins et des sardines). À l'origine, les travaux de Volterra étaient motivées par une observation surprenante des pêcheurs italiens pendant la première guerre mondiale : pendant cette période où il y avait très peu de pêche, le nombre de sardines dans l'océan *diminuait*, mais le nombre de requins *augmentait* par rapport aux populations habituelles.

Notons la populations des requins $r(t)$, et celle des sardines $s(t)$. Le modèle de Lotka-Volterra suppose que les taux d'accroissement de ces espèces sont données par

$$s'(t) = a\,s(t) - c\,s(t)r(t) \qquad \text{et} \qquad r'(t) = -b\,r(t) + d\,s(t)r(t)$$

a, b, c, d étant des constantes positives dépendant des espèces considérées.

1°) Interprétation – Expliquer pourquoi ce système d'équations différentielles modélise la dynamique des populations. Pour celà, regarder d'abord ce qui se passe dans le cas particulier où $r(t) \equiv 0$ (pas de requins, que de sardines) et dans le cas $s(t) \equiv 0$. Pour le cas général, remarquez que la quantité $s(t) \cdot r(t)$ est proportionnelle au nombre de rencontres entres requins et sardines.

Comment est-ce qu'une diminution de la pêche influe sur les paramètres a, b, c et d ?

2°) Résolution numérique – Écrire une fonction MATLAB `Euler.m` mettant en œuvre la méthode d'Euler pour calculer les valeurs approchées de la solution ayant pour condition initiale `s0,r0` sur l'intervalle de temps `[0,Duree]` pour un pas `h` en fonction des paramètres `a,b,c,d`.

3°) Résolution numérique – Faites de même pour la méthode de Runge-Kutta : `RK.m`.

4°) Illustrations – Écrire deux scripts `SolEuler.m` et `SolRK.m` qui tracent sur un même graphique les valeurs approchées de s et de r en fonction du temps obtenues pour les paramètres

`a=2, b=1, c=0.4, d=0.1, Duree=5, h=0.1`

dans chacun de cas donnés par les conditions initiales suivantes

`[s0,r0] = [22,8] , [18,7] , [14,6] , [10,5].`

Écrire un script qui trace sur un seul et même graphique les valeurs approchées de r en fonction de celles de s pour les paramètres et conditions initiales ci-dessus. Autrement dit, le script doit tracer les courbes $(s(t), r(t))$ dans le plan $\mathbb{R}^2$, pour les conditions initiales ci-dessus.

5°) Résolution exacte – Une *intégrale première* du système de Lotka-Volterra est, par définition, une fonction $L(s, r)$ qui reste constante lorsque r et s évoluent. Une telle fonction L a donc la propriété suivante : si $(s(t), r(t))$ est une solution de l'équation différentielle, alors la courbe $(s(t), r(t))$ dans le plan $\mathbb{R}^2$ se balade sur une courbe de niveau de la fonction L.

Vérifier, par un calcul direct, que

$$L(s, r) = a\ln(r) + b\ln(s) - c\,r - d\,s$$

est une intégrale première.

Écrire ensuite un script qui trace sur un seul et méme graphique plusieurs courbes de niveau de cette fonction $L(s, r)$. (Indication : la commande `contour` pourrait être utile ici.)

6°) Comparaison – Ce dessin est-il cohérent avec les graphiques donné par le deuxième script de la question précédente ? Écrire un script qui trace sur un même graphique les valeurs approchées de s en fonction de celles de r et leurs valeurs exactes pour les paramètres ci-dessus et la condition initiale `[14,6]`.

7°) Et la pêche dans tout ça ? – Montrer par un calcul sur une feuille de papier que la fonction L a un maximum à $\left(\frac{b}{d}, \frac{a}{c}\right)$. En déduire que $(s(t), r(t)) = \left(\frac{b}{d}, \frac{a}{c}\right)$ est une solution constante de l'équation différentielle. Selon cette solution, comment est-ce que le nombre de sardines et requins change lors d'une diminution de la pêche ?

"Rappels"

Runge-Kutta – La méthode de Runge-Kutta classique (dite d'ordre 4) pour obtenir des valeurs approchées d'une solution d'une équation d'ordre 1 et **de rang p (*i.e.* d'inconnue $x : \mathbb{R} \to \mathbb{R}^p$, c'est a dire un vecteur de p fonctions)**

$$\begin{cases} x'(t) = f(t, x(t)) & \forall t \in [t_0, t_0 + T] \\ y(t_0) = x_0 \in \mathbb{R}^p \end{cases}$$

consiste à calculer y_{n+1} au temps $t_{n+1} = t_n + h$ à partie de x_n au temps t_n par

$$\begin{aligned} k_1 &= f(t_n, x_n) \\ k_2 &= f(t_n, x_n + (h/2)k_1) \\ k_3 &= f(t_n, x_n + (h/2)k_2) \\ k_4 &= f(t_n, x_n + hk_3) \\ x_{n+1} &= x_n + (h/6)(k_1 + 2(k_2 + k_3) + k_4) \end{aligned}$$