

MT12 - TP6 : Résolution approchée d'équations différentielles

Mai/Juin 2023

Il n'est pas toujours possible de calculer explicitement la solution d'une équation différentielle. Le but de ce TP est de présenter différentes méthodes numériques permettant d'approcher la solution de ces équations différentielles.

1 Consignes pour la remise du rapport de TP

Ce TP est à rendre par mail à gildas.bayard@utc.fr, au plus tard

- le 13 juin 2023 pour les étudiant.e.s de la semaine A,
- le 20 juin 2023 pour les étudiant.e.s de la semaine B.

Un rapport est attendu pour chaque binôme.

Votre rapport devrait être rendu en $\text{\LaTeX}$ car c'est un outil tout à fait adapté au rendu de travaux mathématiques, vous pouvez l'installer ou utiliser un éditeur en ligne à l'adresse suivante <https://fr.overleaf.com/>. Cependant, il vous est possible de rendre un document issu d'un logiciel de traitement de texte de votre choix tant que le rendu est propre.

Un document de rendu de TP se doit de comprendre une introduction et une conclusion, un propos clair et relatant la méthode utilisée, les difficultés rencontrées et des figures pour illustrer. Toutes les figures se doivent d'être correctement introduites et légendées. À la lecture de votre rapport, on doit comprendre facilement en le feuilletant :

1. Quel est le problème.
2. Quelle est la méthode utilisée pour le résoudre.
3. Quels sont les résultats obtenus.
4. Pourquoi vous avez confiance en votre résultat.

2 Introduction

Soit $f : y \mapsto f(y)$ une fonction définie de $\mathbb{R}^n$ ($n \geq 1$) dans $\mathbb{R}^n$. Soit $T > 0$ et $y_0 \in \mathbb{R}^n$, on considère le système d'équations différentielles ordinaires (EDO) :

$$\begin{cases} y'(t) = f(y(t)) \text{ pour } t \in [0, T], \\ y(0) = y_0. \end{cases} \quad (1)$$

Par exemple si $n = 2$ avec $y(t) = (y_1(t), y_2(t))^T$ et $f(y(t)) = (y_1(t), -y_2(t))^T$, alors dans ce cas le système (1) s'écrit

$$\begin{cases} y_1'(t) &= y_1(t) \text{ pour } t \in [0, T], \\ y_2'(t) &= -y_2(t) \text{ pour } t \in [0, T], \end{cases}$$

avec $y(0) = y_0 = (y_1(0), y_2(0))^T$ une donnée initiale.

Dans ce TP nous allons considérer différents schémas numériques pour le problème (1). Pour cela, on se donne une subdivision régulière de $[0, T]$: $t_0 = 0 < t_1 < \dots < t_{N-1} < t_N = T$ de pas $\Delta t = T/N$ et on cherche à calculer une solution approchée $(Y^k)_{1 \leq k \leq N}$ (où $Y^k \in \mathbb{R}^n$) du problème (1) où $Y^k \approx y(t_k)$ pour $k = 1, \dots, N$. Dans la suite on va s'intéresser à trois méthodes différentes pour calculer les vecteurs $(Y^k)_{1 \leq k \leq N}$:

(a) **Méthode d'Euler explicite** :

$$Y^{k+1} = Y^k + \Delta t f(Y^k), \quad k = 0, \dots, N-1.$$

(b) **Méthode de Heun** (Runge-Kutta d'ordre 2) :

$$Y^{k+1} = Y^k + \frac{\Delta t}{2} \left(f(Y^k) + f(Y^k + \Delta t f(Y^k)) \right), \quad k = 0, \dots, N-1.$$

(c) **Méthode RK4 « classique »** : pour $k = 0, \dots, N-1$,

$$\begin{cases} k_1 = f(Y^k), \\ k_2 = f(Y^k + \frac{\Delta t}{2} k_1), \\ k_3 = f(Y^k + \frac{\Delta t}{2} k_2), \\ k_4 = f(Y^k + \Delta t k_3), \\ Y^{k+1} = Y^k + \frac{\Delta t}{6} (k_1 + 2k_2 + 2k_3 + k_4). \end{cases}$$

Dans tous les cas, on prend pour Y^0 la condition initiale y_0 .

3 Résolution numérique d'EDO

Afin de tester ces méthodes numériques nous allons étudier différentes EDO pour lesquels il est possible de calculer la solution exacte.

3.1 Programmation des méthodes

1. Calculer la solution exacte y de (1) dans le cas suivant :

$$f(y) = -50y, \quad y_0 = 10.$$

2. Définir une fonction scilab `sol` donnant l'expression de la solution exacte de (1). Écrire ensuite trois fonctions `EulerExpl`, `Heun` et `RK4` qui, partant de Y_0 , T et N , calculent les solutions approchées $(Y^k)_{0 \leq k \leq N}$ pour chacune des 3 méthodes. Par exemple, dans le cas de la fonction `EulerExpl` aidez-vous du script suivant :

```

1 function y=f(x)
2     y=-50*x
3 endfunction
4
5 function Y=EulerExpl(Y0,T,N,f)
6     dt=T/N;
7     Y=zeros(length(Y0),N)
8     Y(:,1)=...; // remplir
9     for i=1:N-1
10         Y(:,i+1)=...; // remplir
11     end
12 endfunction

```

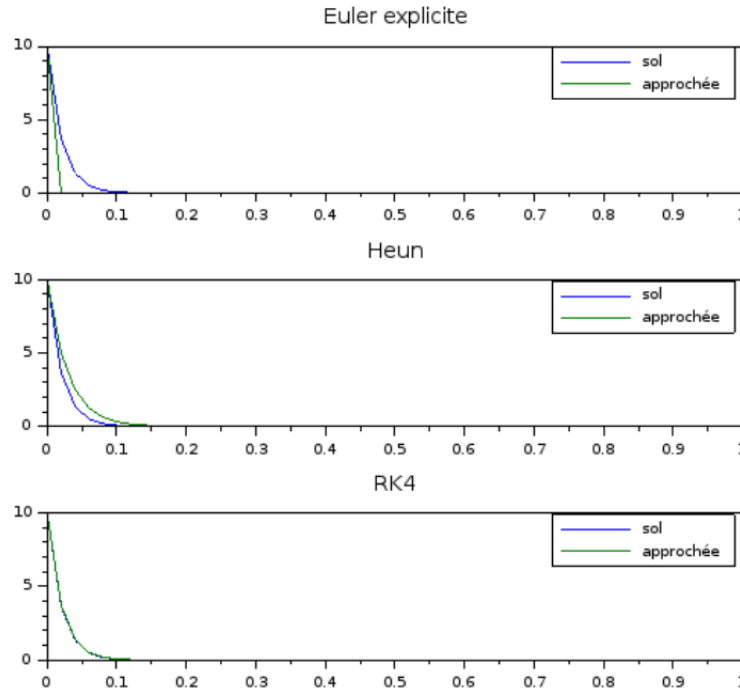


FIGURE 1 – Illustration des méthodes numériques (a), (b) et (c).

Pour $T = 1$ et $N = 50$ et en utilisant la fonction `subplot` de `Scilab`, tracer sur une même figure trois graphiques représentant la solution approchée obtenue par les trois méthodes et la solution exacte (comme sur la Figure 1).

3.2 Vitesse de convergence

Pour les trois méthodes numériques proposées il est possible de démontrer que lorsque $\Delta t \rightarrow 0$ alors les solutions approchées convergent vers la solution exacte de l'EDO. On s'intéresse alors à la vitesse de convergence de chacune des méthodes. Pour ce faire on considère le problème (1) toujours avec $f(y) = -50y$ et $y_0 = 10$. Pour un schéma donné et une valeur de N (on a alors $\Delta t = T/N$), on définit l'erreur

$$E = \max_{0 \leq k \leq N} |Y^k - y(t_k)|.$$

3. Écrire un script `Scilab` qui pour $N = 30, 60, 120, 240, 480$ et 960 permet de calculer les erreurs associées aux méthodes (a), (b) et (c). Les résultats seront stockés dans trois vecteurs `ErreurEulerExpl`, `ErreurHeun` et `ErreurRK4`.
4. Mettre en évidence graphiquement les vitesses de convergence de chacun des schémas en utilisant le script suivant :

```

1 plot([30,60,120,240,480,960], log(ErreurEulerExpl), 'o-');
2 plot([30,60,120,240,480,960], log(ErreurHeun), 'r');
3 plot([30,60,120,240,480,960], log(ErreurRK4), 'black');
4 legend('Euler Explicite', 'Heun', 'RK4')
5 title("Vitesse de convergence")

```

5. Quelle méthode semble la plus efficace ?

3.3 Le cas d'un système de deux EDO découplées

On considère à présent le système d'EDO suivant

$$\begin{cases} y_1'(t) &= -50y_1(t), \quad \text{pour } t \in [0, T], \\ y_2'(t) &= 3y_2(t), \quad \text{pour } t \in [0, T], \\ y(0) &= (10, 2)^\top. \end{cases} \quad (2)$$

Le système d'EDO (2) est dit découplé car on peut résoudre de manière indépendante l'équation vérifiée par y_1 et celle vérifiée par y_2 .

- Montrer que le système (2) peut se mettre sous la forme (1) avec f définie de $\mathbb{R}^2$ dans $\mathbb{R}^2$. Donner l'expression de la solution exacte du système (2).
- Pour $T = 0.5$ et $N = 50$ et en utilisant la fonction `subplot` de `Scilab`, tracer sur une même figure trois graphiques représentant la solution approchée obtenue par la méthode (a), (b) et (c) et la solution exacte (comme dans le cas de la Figure 1).

3.4 Le modèle proies-prédateurs, un système non linéaire

Dans le cas des EDO considérés jusqu'à présent l'utilisation d'une méthode numérique n'est pas vraiment utile car il est possible de calculer « à la main » la solution exacte. Les exemples précédents permettent juste d'illustrer les méthodes numériques et fournissent des cas tests utiles. Considérons à présent un système d'EDO où la solution exacte n'est pas calculable à la main.

On veut calculer les solutions approchées du système différentiel d'ordre 1 non linéaire :

$$\begin{cases} x_1'(t) &= 3x_1(t) - x_1(t)x_2(t), \\ x_2'(t) &= -2x_2(t) + x_1(t)x_2(t), \end{cases} \quad (3)$$

Ici x_1 représente le nombre de proies (par exemple des lapins) et x_2 le nombre de prédateurs (par exemple des renards). Le modèle (3) est un modèle de type proies-prédateurs, appelé aussi modèle de Lotka-Volterra. Ce type de système est utilisé pour décrire la dynamique de systèmes biologiques dans lesquels une proie et son prédateur interagissent.

On peut interpréter les équations du système (3) de la manière suivante : pour la première équation en absence de prédateurs ($x_2(t) = 0$) alors les lapins se reproduisent avec un taux d'accroissement $a = 3$. Cependant les lapins sont mangés proportionnellement à la présence de prédateurs, phénomène modélisé par le terme $-x_2(t)$. De la même manière, pour la deuxième équation, en l'absence de lapins ($x_1(t) = 0$) les prédateurs ne peuvent se nourrir et meurent, phénomène représenté par le terme $-2x_2(t)$ où 2 représente le taux de mortalité. Par contre, en présence de proies, la population de renards s'accroît proportionnellement, ce qui explique le terme $x_1(t)$. Schématiquement, on peut résumer la situation sous la forme :

$$\begin{aligned} x_1'(t) &= \underbrace{3x_1(t)}_{\text{croissance}} + \underbrace{(-x_1(t)x_2(t))}_{\text{proies consommées}}, \\ x_2'(t) &= \underbrace{-2x_2(t)}_{\text{mortalité prédateurs}} + \underbrace{x_1(t)x_2(t)}_{\text{croissance}}. \end{aligned}$$

- Montrer que le système (3) peut se mettre sous la forme (1) avec f définie de $\mathbb{R}^2$ dans $\mathbb{R}^2$. Définir en `Scilab` la fonction f en vous aidant du script suivant :

```
1 function ans=f(X)
2     ans=[3*X(1)...;...]
3 endfunction
```

9. Pour $N = 5000$, calculer la solution approchée du système (3) pour la méthode d'Euler explicite, de Heun et RK4 sur l'intervalle $[0, 20]$ avec $x_1(0) = 1$ et $x_2(0) = 2$. Représenter graphiquement $x_2(t)$ en fonction de $x_1(t)$, pour ce faire on utilisera la fonction `comet` de Scilab.
10. Commenter les résultats obtenus.

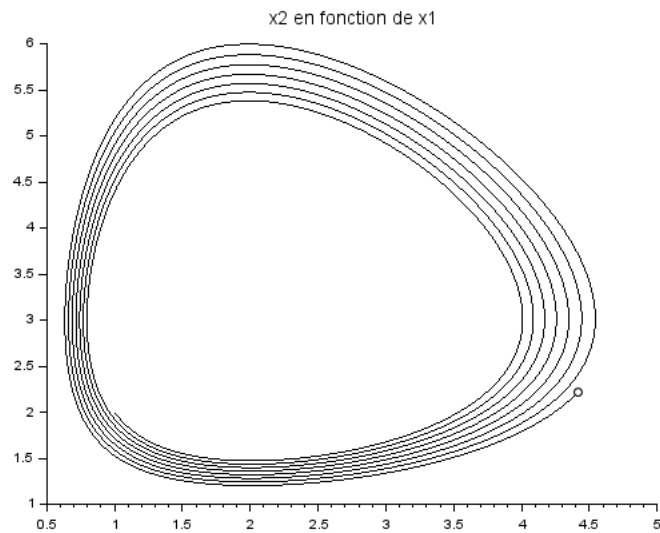


FIGURE 2 – $x_2(t)$ en fonction de $x_1(t)$ par la méthode d'Euler explicite.