**TP N°5 : Equations différentielles**

**But de TP :**

Etudier deux méthodes de résolution des équations différentielles ; **Euler** et **Runge-Kutta**,

Implémenter sous Matlab ces deux méthodes et comparer les solutions obtenues avec la solution exacte.

1. **Méthode d’Euler**

Soit $I=[a,b]$ un intervalle fermé de , et $f $est une fonction donnée $f:I×R\rightarrow R$ et $y$ est une fonction différentielle $R\rightarrow R$.

On appelle *équation différentielle* de premier ordre, la relation :

$$\frac{dy(t)}{dt}=f\left(t,y\left(t\right)\right) $$

On dit que $y$ est la solution de cette équation différentielle sur $[a,b]$, si $y $vérifie l’équation au dessus pour tout $t\in [a,b]$.

La méthode d’Euler est une méthode d’analyse numérique fournit des approximations $y\_{i}$ de $y(t\_{i})$ pour $i=1,2…N$, N un nombre entier. Son principe consiste à substituer $\frac{dy(t)}{dt}$ par l’expression $\frac{y\left(t+h\right)-y(t)}{h}$.

L’algorithme d’Euler s’écrit alors :

$\left\{\begin{array}{c}y\_{0}=y\left(a\right) \\y\_{i+1}=y\_{i}+hf\left(t\_{i},y\_{i}\right)\end{array}\right. \begin{matrix}donné\\i=1,2…N\end{matrix}$ Avec $h=\frac{b-a}{N}$ , $t\_{i}=a+i h$.

1. **Méthode de Runge-Kutta**

La méthode de *Runge-Kutta* permet ainsi de calculer des approximations $y\_{i}$ de $y\left(t\_{i}\right)$ en suivant l’algorithme suivant :

$\left\{\begin{array}{c}\begin{matrix}y\_{0}=y\left(a\right) \\K1=hf\left(t\_{i},y\_{i}\right) \end{matrix} \\\begin{matrix}K2=hf\left(t\_{i}+\frac{1}{2}h,y\_{i}++\frac{1}{2}K1\right)\\y\_{i+1}=y\_{i}+K2 \end{matrix}\end{array}\right. $ Avec $h=\frac{b-a}{N}$ , $t\_{i}=a+i h$.

**Exercice**

1. On considère une fonction $f=-2ty$ définit sous un intervalle [0,1], calculer à la main une approximation de la solution $y$ par la méthode d’Euler puis par la méthode de Runge Kutta, on prenant N=10 et . y0=1.
2. Programmer deux fonctions *Euler.m* et *RungeKutta.m* qui calcule l’approximation de $y$, avec les paramètres suivants.

|  |  |
| --- | --- |
| **Les entrées** | **Les sorties** |
| $a$et***b***les bornes de l’intervalle***[a, b]***,La fonction ***f*****N** le nombre des intervalles $y\_{0}$ la valeur de la condition initiale | ***y****:* la solution approximée $t$ : vecteur de *t* discrétisé |

1. A l’aide de la fonction *Euler.m* et *RungeKutta.m* calculer une valeur approximée de y, avec N=10 et N=100.
2. considérant la solution exacte $y\_{exacte}=e^{-t^{2}}$, tracer sous Matlab la solution exacte et celle approximée avec Euler et avec Runge-Kutta, que remarquez vous avec N=10 et N=100.