TP noté : il devra faire l'objet d'un compte-rendu écrit à réaliser seul ou en binôme, et à rendre au plus tard le vendredi 15 mai 2015. Les programmes réalisés pour répondre à chaque exercice, en language Scilab(ou Matlab) devront être deposés sur DIDEL (section Travaux)

Pour chaque exercice, on demande de rendre un programme principal (sous la forme d'un fichier) ***Ex1.sci****,* ***Ex2.sci****, etc., qui comprendra les définitions des fonctions utilisées ainsi que les instructions nécessaires à l'exécution du programme.*

Le compte-rendu est à rendre : sous forme électronique (document Word, pdf), en le joignant aux programmes.

Equations différentielles

Exercice 1. Mise en oeuvre de schémas de résolution numérique

On considère l'équation différentielle

$$\begin{cases} x' = \sin(tx) \\ x(0) = 1/10 \end{cases} \tag{1}$$

de solution unique sur $[0; 1/2]$.

1) Résoudre (analytiquement) le problème de Cauchy approché

$$\begin{cases} y' = ty \\ y(0) = 1/10 \end{cases} \tag{2}$$

2) Ecrire un script Scilab pour calculer la solution approchée de l'équation avec le schéma d'Euler Explicite. On comparera sur le même graphique la solution par Euler explicite des équations différentielles (**??**) et (**??**) ainsi que la solution exacte de la deuxième équation.

3) Vérifier la rélation d'approximation de (**??**) par (**??**) où $E(t) = y(t) - x(t)$ représente l'erreur.

$$\sup_{0 \leqslant t \leqslant 1/2} |E(t)| \leqslant 5.10^{-6} \tag{3}$$

4) Tester et comparer la solution numérique obtenue avec les schémas d'intégration d'Euler implicite, de Crank-Nicholson et Runge Kutta d'ordre 2 et 4. Illustrer graphiquement.

5) *Jouer* sur les paramètres (dont le pas de temps) ainsi que sur le choix du schéma pour illustrer la convergence plus ou moins rapide. Illustrer graphiquement.

Exercice 2. Vérification de l'ordre de convergence

On veut maintenant mettre en évidence l'ordre de convergence d'un schéma numérique. Pour cela, on va se placer dans un cas où on connait la solution exacte de l'équation differentielle. Pour differentes valeurs du pas de discrétisation $\Delta t = t_i - t_{i-1}$, on calcule à un temps T fixé, la solution exacte U_e et approchée U_a, et on trace l'erreur - la valeur absolue de la difference entre ces deux valeurs - en fonction de t.

```
deff('y=fun(x,t)','y=2*t*sqrt(1.-x.^2)')
deff('y=exacte(t)','y=sin(t.^2)')
y0=0;
T=[0,1];
np=14;
h=zeros(np,1);
E=zeros(h);
nt=1;
for i=1:np
    nt=2*nt;
    h(i)=(T(2)-T(1))/nt;
    y=EulerExplicite(fun,y0,T,nt);
    E(i)=abs(y(2)-exacte(T(2)));
end
xset("window",2)
xbasc()
xtitle('verification de l''ordre de convergence du schema Euler explicite')
plot2d(h,E,2,leg='erreur euler explicite',strf='121',logflag="ll");
plot2d(h,h,[3,2],leg='ordre 1',strf='101',logflag="ll");
```

où `EulerExplicite(f,x0,T,n))` calcule une solution numérique `x` en differentes valeurs de t données dans `T` (`T(1)` étant le temps initial) de l'équation

$$\left\{ \begin{aligned} x'(t) &= \mathtt{fun}(x(t),t) \\ x(\mathtt{T(1)}) &= \mathtt{x0} \end{aligned} \right. \tag{4}$$

1. Quelle est l'équation différentielle résolue ? Sur quel intervalle connait-on sa solution ?
2. A quoi sert l'argument `logflag` dans l'appel à `plot2d` ?
3. Que se passe-t-il si on prend des valeurs de T de plus en plus grand jusqu'à $\sqrt{\pi/2}$?
4. Adapter le script pour vérifiér l'ordre numérique du schéma de Runge Kutta. Commenter la courbe obtenue.