

L2 - Langage de programmation - Equations différentielles

Jean-Paul CHEHAB

LAMFA UMR CNRS 7352, Univ. Picardie Jules Verne

Confinement, an 1

Plan

- 1 Pourquoi s'intéresser aux équations différentielles ?
 - Types d'EDOs
 - Equation différentielles d'ordre 1
 - Systèmes différentiels du premier ordre

Plan

- 1 Pourquoi s'intéresser aux équations différentielles ?
 - Types d'EDOs
 - Equation différentielles d'ordre 1
 - Systèmes différentiels du premier ordre
- 2 Les EDOs avec Scilab
 - Le solveur ode
 - Représentation des solutions
 - Systèmes différentiels du premier ordre
 - Représentation des solutions

Plan

1 Pourquoi s'intéresser aux équations différentielles ?

- Types d'EDOs
- Equation différentielles d'ordre 1
- Systèmes différentiels du premier ordre

2 Les EDOs avec Scilab

- Le solveur ode
- Représentation des solutions
- Systèmes différentiels du premier ordre
- Représentation des solutions

3 Applications

- Exemple 1 : Le pendule Oscillant
- Exemple 2 : Equation de Van der Pol
- Exemple 3 : Dynamique des populations

Plan

1 Pourquoi s'intéresser aux équations différentielles ?

- Types d'EDOs
- Equation différentielles d'ordre 1
- Systèmes différentiels du premier ordre

2 Les EDOs avec Scilab

- Le solveur ode
- Représentation des solutions
- Systèmes différentiels du premier ordre
- Représentation des solutions

3 Applications

- Exemple 1 : Le pendule Oscillant
- Exemple 2 : Equation de Van der Pol
- Exemple 3 : Dynamique des populations

4 Ca se complique !

- Le système de Lorenz
- Système de Rössler
- Petit bestiaire

Plan

1 Pourquoi s'intéresser aux équations différentielles ?

- Types d'EDOs
- Equation différentielles d'ordre 1
- Systèmes différentiels du premier ordre

2 Les EDOs avec Scilab

- Le solveur ode
- Représentation des solutions
- Systèmes différentiels du premier ordre
- Représentation des solutions

3 Applications

- Exemple 1 : Le pendule Oscillant
- Exemple 2 : Equation de Van der Pol
- Exemple 3 : Dynamique des populations

4 Ca se complique !

- Le système de Lorenz
- Système de Rössler
- Petit bestiaire

5 Do It Yourself (DIY)

Représenter des phénomènes naturels et de l'activité humaine

- les équations de la mécanique classique, de la cinétique chimique, de la météorologie
- certaines équations en sciences du vivant : biologie, médecine, écologie
- mathématiques de l'économie, de la finance, des comportements humains (mouvements de foule)

Une immense richesse mathématique

- Depuis Henri Poincaré qui a fait le constat d'échec sur la possibilité de résoudre explicitement les EDOs, pratiquement tous domaines des mathématiques sont utilisées pour étudier les systèmes dynamiques : analyse, algèbre, topologie, géométrie, théorie des nombres, probabilités.
- Si on ne peut pas exhiber les solutions, il faut les approcher pour les simuler : analyse numérique

Les équations différentielles ordinaires (EDO)

De manière la plus générale une équation différentielle s'écrit comme

$$f(y, y', y'', \dots, y^{(p)}, x) = 0, \quad (1)$$

où $y^{(k)}$, $k = 0, \dots, p$ est la dérivée k ème de l'inconnue y par rapport à la variable indépendante x .

Definition

On dit que l'équation différentielle est d'ordre p si p est le plus grand ordre de dérivation en présence.

L'équation différentielle (2) peut se réécrire sous la forme d'un système différentiel après avoir effectué le changement de variables

$$y_1 = y', \quad y_2 = y^{(2)}, \dots, \quad y_p = y^{(p)},$$

$$\begin{cases} f(y, y_1, y_2, y_3, \dots, y_p, x) = 0, \\ \frac{dy_1}{dx} = y_2, \\ \vdots \\ \frac{dy_{p-1}}{dx} = y_p \end{cases} \quad (2)$$

De manière générale on peut écrire

$$a_p y^{(p)} + a_{p-1} y^{(p-1)} + \dots + a_1 y' + a_0(t) = b(t, y, \dots)$$

Ici les fonctions a_i peuvent dépendre du temps t et aussi de la solution y et de ses dérivées $y', \dots, y^{(p-1)}$.

On distingue deux cas particuliers

Les EDOs autonomes

Ce sont les équations pour lesquelles les fonctions a_i et b ne dépendent pas du temps t . Lorsque $b(t) = 0$ on parlera d'équations homogènes

EDOs linéaires

Les équations différentielles linéaires s'écrivent sous la forme

$$a_p(t) y^{(p)} + a_{p-1}(t) y^{(p-1)} + \dots + a_1(t) y' + a_0(t) = b(t).$$

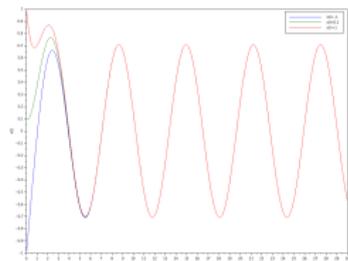
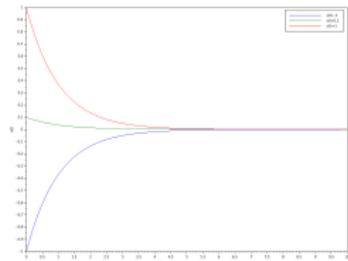
Lorsque les fonctions a_i sont constantes, on parle d'EDO à coefficients constants. Elles sont homogènes lorsque de plus $b(t) = 0$.

Equations linéaires

Equations

- $\frac{dx}{dt} = ax \Rightarrow x(t) = e^{at} x_0$
- $\frac{dx}{dt} = ax + b(t)$
Variation de la constante :
sol=sol. homogène + sol.
particulière
 $x(t) = e^{at} x_0 + \int_0^t e^{(t-s)a} b(s) ds$

Illustration



Equations Non linéaires

On ne sait pas les résoudre en général (on sait juste intégrer une dérivée !)
Voici quelques techniques (parmi de très très nombreuses)

Techniques

- changement de variables
- équations à variables séparées. Elles sont de la forme

$$x' = f(t)g(x) \Rightarrow \frac{x'(t)}{g(x(t))} = f(t)$$

On intègre séparément chaque membre.

Cas particuliers

Equations de Bernoulli. Elles sont de la forme $x'(t) + a(t)x(t) = b(t)x^m(t)$ On pose $z(t) = \frac{1}{x^{m-1}}$, $z(t)$ est solution de $\frac{1}{1-m}z'(t) + a(t)z(t) = b(t)$. On trouve

$$x(t) = x_0 \exp\left(-\int_{t_0}^t a(s)ds\right) \left(1 + (1-m)x_0^{m-1} \int_{t_0}^t b(s) \left[\exp\left(-\int_{t_0}^s a(\xi)d\xi\right)\right]^{m-1} ds\right)^{\frac{1}{1-m}}$$

Ils correspondent au cas $f : \mathbb{R}^n \rightarrow \mathbb{R}^n$. La façon la plus simple de les aborder est de considérer d'abord le cas linéaire à coefficients constants

$$\begin{cases} \frac{dx(t)}{dt} = Ax & t \in]0, T[, \\ x(0) = x_0 \end{cases}$$

Ici A est une matrice $n \times n$.

Theorem

La solution est $x(t) = e^{tA} x_0$ où $e^{tA} = \sum_{k=0}^{+\infty} \frac{(tA)^k}{k!}$.

On admet que la série converge normalement. On peut dériver par rapport à t et on vérifie que (t) est solution du système. Le comportement des solutions dépend du signe de la partie réelles des valeurs propres.

Dans Scilab est implantée une fonction de résolution numérique d'EDO.
Considérons le problème de Cauchy

$$\begin{cases} \frac{dx(t)}{dt} = f(x(t), t) & t \in]0, T[, \\ x(0) = x_0 \end{cases}$$

Le solveur d'EDO est `ode` et il s'utilise basiquement comme suit :

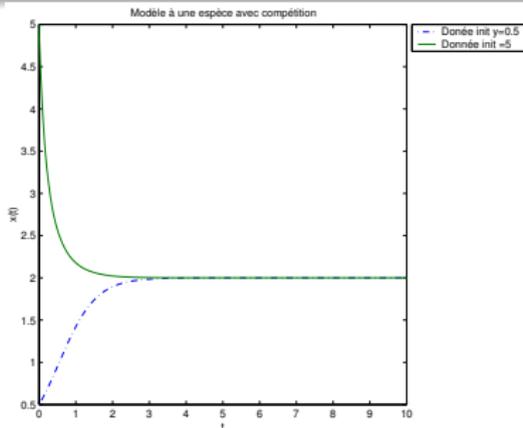
- On définit d'abord une fonction second membre ou source
function [f]= source(t,u)
- On définit le suite des temps discrets
t=linspace(t0,T,n) ou bien
t=t0:dt:T, t0 et dt ayant préalablement été initialisés.
- On calcule les approximations successives de la solution aux temps discrets dans U
[U]=ode(u0,t,source)

IMPORTANT : on ne peut approcher la solution qu'à des temps prédéfinis.

En projection : représenter les courbes $(t, x_i(t))$, $t \in [t_0, T]$ pour $i = 1, \dots, n$

L'équation : $x'(t) = a(x(t) - bx(t))$

```
t=linspace(0,1,100);  
function [f]=popucompet(t,u)  
f=u*(2-u);  
endfunction  
[U]=ode([0,1],t,popucompet);  
plot(t,U)
```



Cas linéaire $X' = Ax$

Formellement

En scilab la commande `expm` permet de calculer l'exponentielle d'une matrice.

```
dt=0.1; ExA=expm(A);
for k=1:N Xnew=ExA*Xnew;
X=[X Xnew];
end
```

On ne le fait jamais : cela coûte trop en temps de calcul !

Avec ode

```
function dxdt=f(t,x)
dxdt=A*x;
endfunction
A=[[-2 1];[ 1 -2]];
// t=linspace(0,100,1000)';
X0=[-1 ; 0.1];
X=ode(X0,0,t,f);
```

Portrait de phase : représenter les courbes $(x_i(t), x_j(t))$, $t \in [t_0, T]$ pour $i, j = 1, \dots, n, i \neq j$

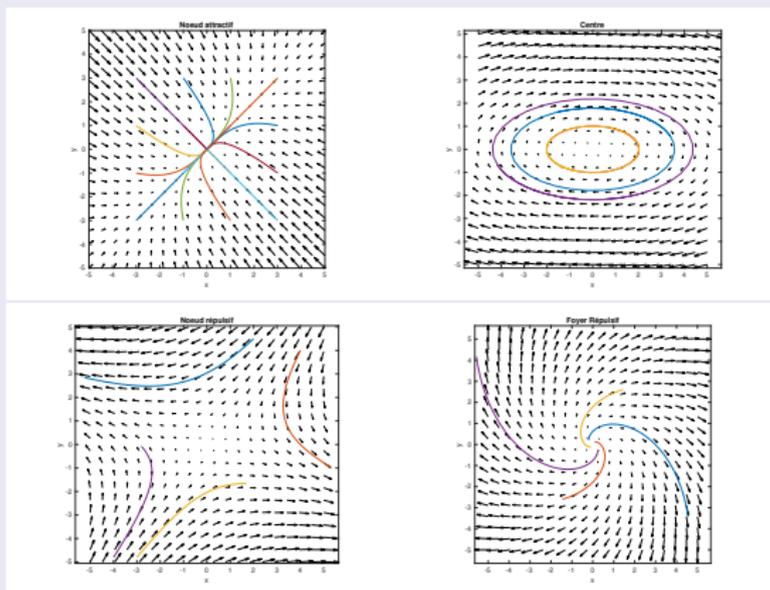
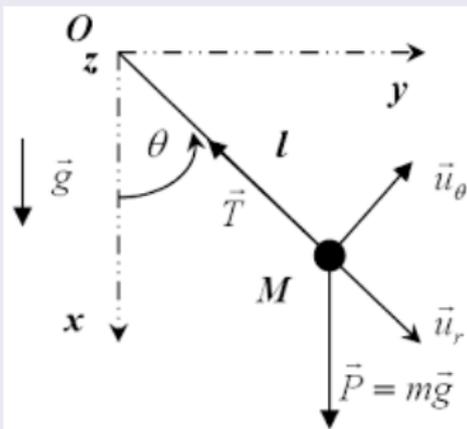


Figure – Noeud attractif. $\lambda_2 < \lambda_1 < 0$ $\lambda_1 < 0$ $\lambda_2 < 0$ col (instable)

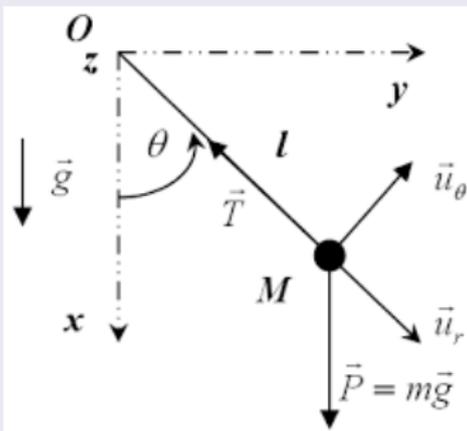
Soit une tige rigide de longueur ℓ dont l'extrémité supérieure est fixée à un plan horizontal. A l'autre extrémité est fixé un poids de masse m , la pesanteur est la seule force extérieure considérée.

Dispositif expérimental



Soit une tige rigide de longueur ℓ dont l'extrémité supérieure est fixée à un plan horizontal. A l'autre extrémité est fixé un poids de masse m , la pesanteur est la seule force extérieure considérée.

Dispositif expérimental

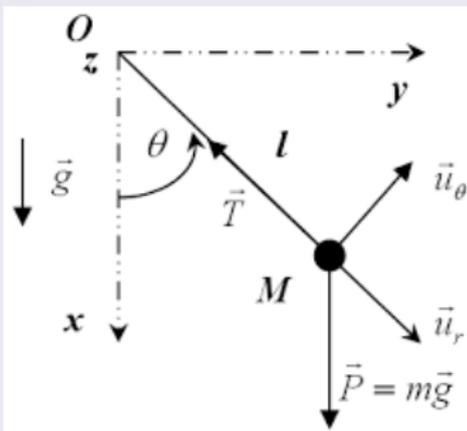


Lois de Newton (RFD)

$$m \frac{d^2 X}{dt^2} = -mg \quad (3)$$

Soit une tige rigide de longueur ℓ dont l'extrémité supérieure est fixée à un plan horizontal. A l'autre extrémité est fixé un poids de masse m , la pesanteur est la seule force extérieure considérée.

Dispositif expérimental



Lorsque les déplacements sont petits θ est petit et, en utilisant l'approximation $\sin \theta \approx \theta$. On obtient alors l'oscillateur harmonique $\theta'' + \alpha^2 \theta = 0$ avec $\alpha^2 = \frac{g}{\ell} > 0$.

Lois de Newton (RFD)

$$m \frac{d^2 X}{dt^2} = -mg \quad (3)$$

On passe en polaires, on projette sur la composante angulaire :

$$\theta'' = -\frac{g}{\ell} \sin(\theta) \quad (4)$$

$$\theta_0 = \theta(0) \quad (5)$$

$$\theta_1 = \theta'(0) \quad (6)$$

On réécrit cette équation d'ordre 2 sous la forme d'un système d'ordre 1 en introduisant l'inconnue auxillière $w = \theta'$. On obtient

$$\begin{cases} w' = -\alpha^2 \sin \theta \\ \theta' = w, \\ \theta(0) = \theta_0, w(0) = \theta_1 \end{cases}$$

Programme

```
t=linspace(0,1,100);  
function [f]=oscill(t,theta)  
f(1)= theta(2)  
f(2)=-sin(theta(1))  
endfunction  
U=ode([0,1],t,oscill);
```

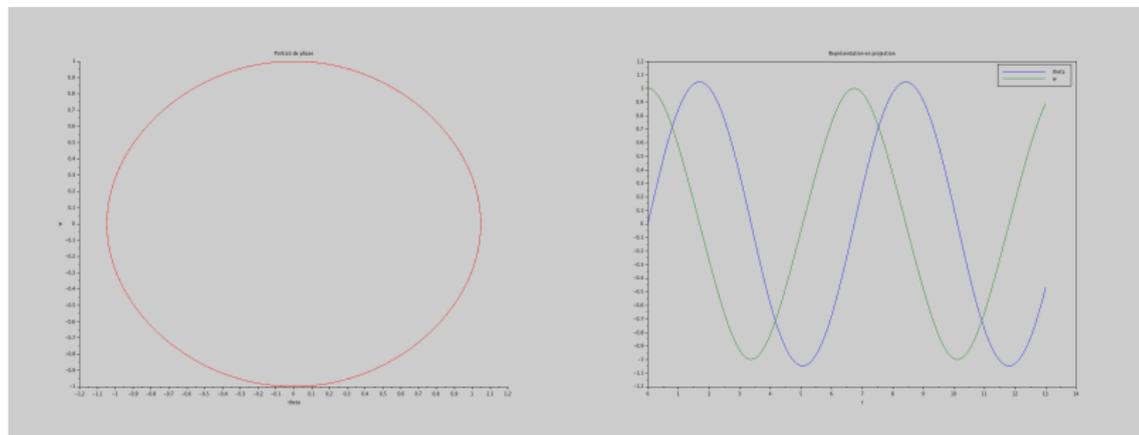


Figure – Oscillateur : représentation en portrait de phase (gauche) et en projection (droite)

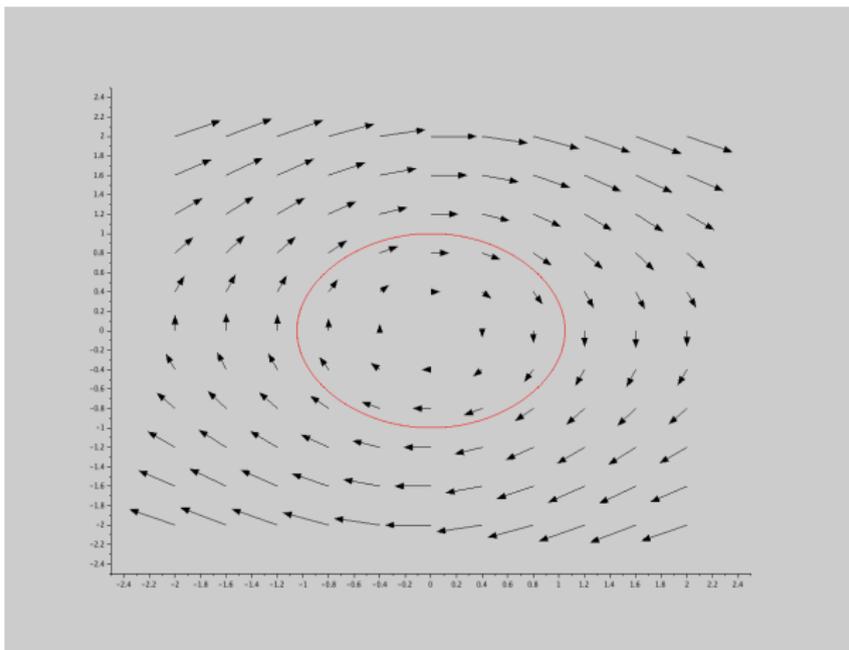


Figure – Oscillateur : représentation et en portrait de phase avec champ de vecteur

Simulation

Est-ce conforme à ce que l'on peut observer ?

Simulation

Est-ce conforme à ce que l'on peut observer ?

Non ! le mouvement est périodique, en particulier le pendule n'atteint pas sa position d'équilibre : à la verticale, pour $\theta = 0$.

Simulation

Est-ce conforme à ce que l'on peut observer ?

Non ! le mouvement est périodique, en particulier le pendule n'atteint pas sa position d'équilibre : à la verticale, pour $\theta = 0$.

On corrige le modèle : on prend en compte la force de frottement avec l'air, elle est proportionnelle à la vitesse $\frac{dX}{dt}$ et est dirigée dans le sens contraire du mouvement

$$m \frac{d^2 X}{dt^2} = -mg - \kappa \frac{dX}{dt}$$

avec $\kappa > 0$

Programme

```
t=linspace(0,1,100);  
function [f]=oscill(t,theta)  
f(1)= theta(2)  
f(2)=-sin(theta(1)) -alpha*theta(2)  
endfunction  
alpha=1; [U]=ode([0,1],t,oscill);
```

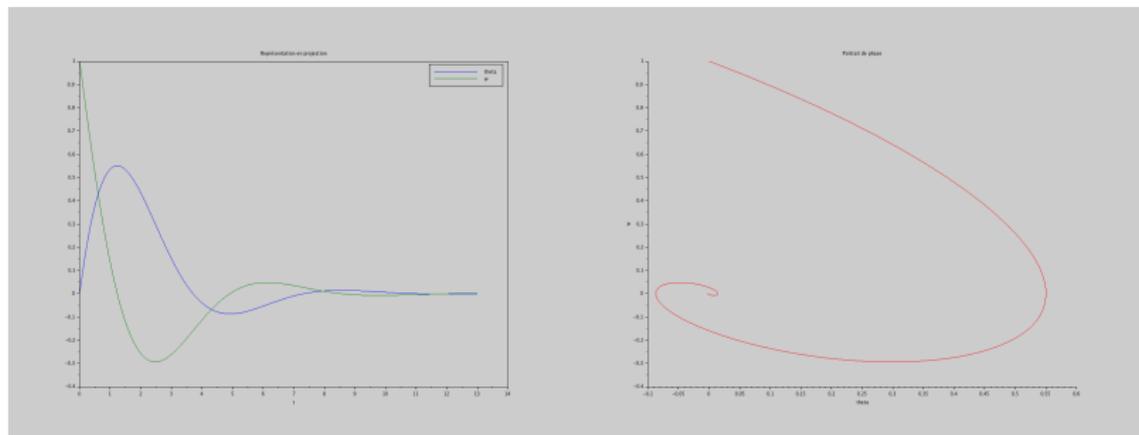


Figure – Oscillateur amorti avec $\alpha = 1$: représentation en projection (gauche) et en portrait de phase (droite)

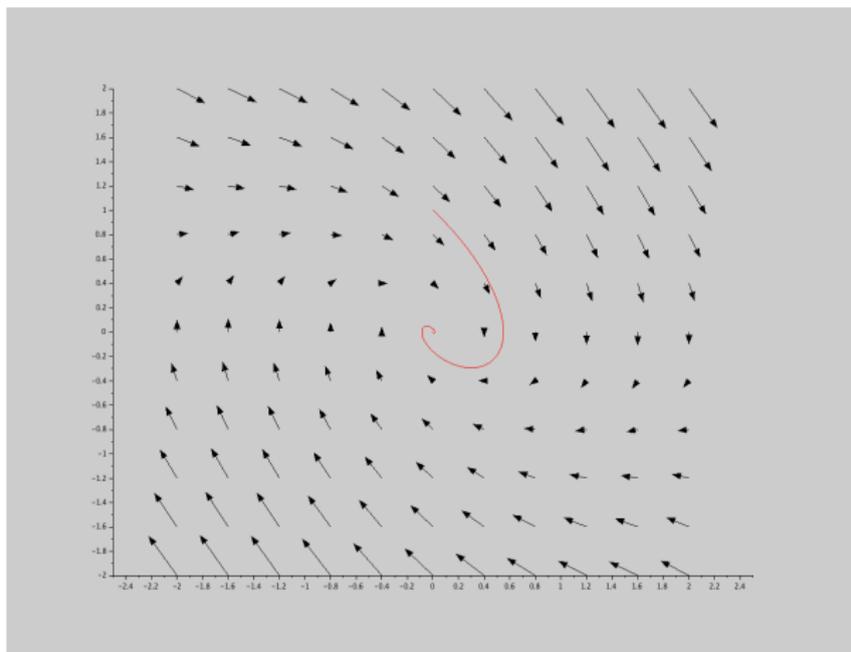


Figure – Oscillateur amorti avec $\alpha = 1$: représentation et en portrait de phase avec champ de vecteur

L'équation de Van der Pol permet de d'écrire des phénomènes vibratoires dans certaines situations ; on peut la voir comme une équation des ondes non linéaire avec terme d'amortissement (actif lorsque $|u| > 1$) et sans dépendance en espace. Elle est aussi utilisée pour modéliser les battements cardiaques.

$$\begin{cases} \left(\frac{d^2}{dt^2} u(t) \right) - \epsilon(1 - u(t)^2) \left(\frac{d}{dt} u(t) \right) + u(t) = 0, & t > 0, \\ u(0) = u_0, \\ u'(0) = u_1. \end{cases}$$

où $\epsilon > 0$.

Programme

```
function [f]=VanDerPol(t,u)
f(1)=u(2); f(2)=1*(1-u(1)^2)*u(2)-u(1);
enndfunction n = 30;
delta = 3
x = linspace(-delta,delta,n); // ici y = x
xbasc()
fchamp(VanDerPol,0,x,x)
xselect()
m = 500 ; T = 30 ;
t = linspace(0,T,m); //
u0 = [-2.5 ; 2.5]; // condition initiale
U= ode(u0, 0, t, VanDerPol);
plot2d(u(1,:),u(2,:),2,"000")
```

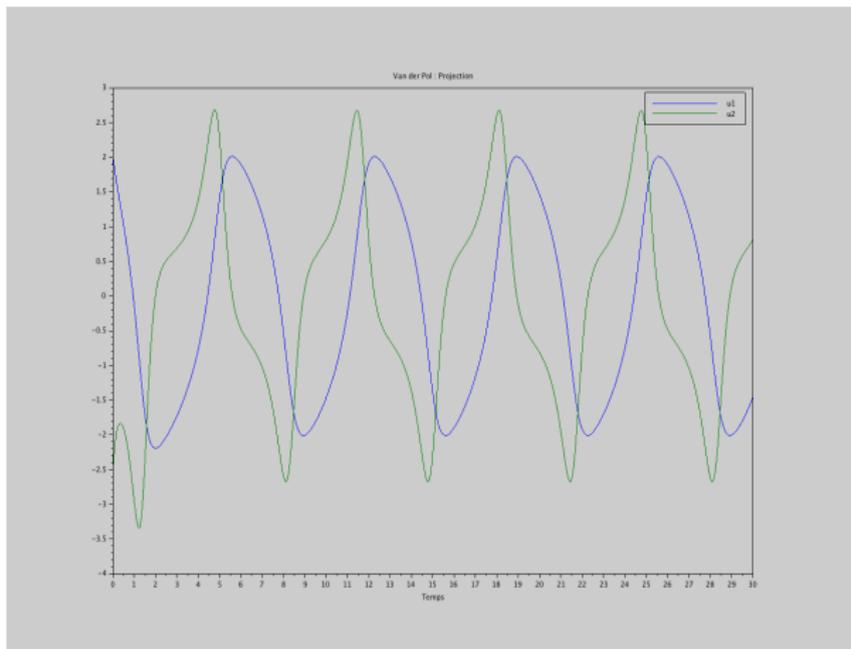


Figure – Equation de Van der Pol - Projection

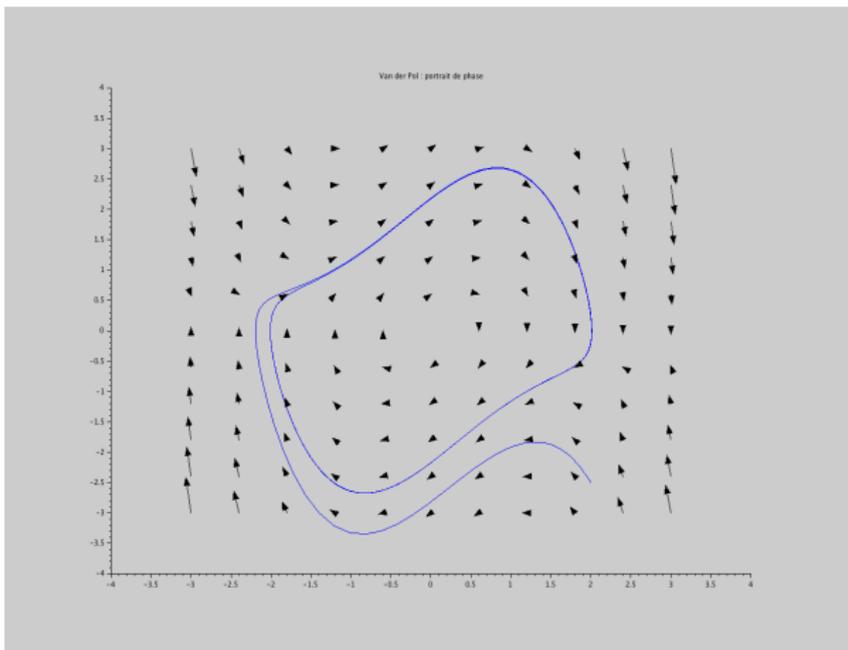


Figure – Equation de Van der Pol - Portrait de phase

Modèle à une espèce

On s'intéresse à l'évolution d'une population au cours du temps. En l'absence de causes de mortalité, on peut considérer que le taux d'accroissement d'une population est constant. Le modèle obtenu est

$$\dot{x}(t) = ax(t), \quad (7)$$

$$x(0) = x_0. \quad (8)$$

Un tel modèle n'est pas réaliste ! La population connaîtrait une croissance exponentielle.

Modèle à une espèce

On s'intéresse à l'évolution d'une population au cours du temps. En l'absence de causes de mortalité, on peut considérer que le taux d'accroissement d'une population est constant. Le modèle obtenu est

$$\dot{x}(t) = ax(t), \quad (7)$$

$$x(0) = x_0. \quad (8)$$

Un tel modèle n'est pas réaliste ! La population connaîtrait une croissance exponentielle. On le corrige en tenant compte par exemple du taux de mortalité dû à la compétition, le manque de nourriture pour tous. Le taux de croissance a est corrigé proportionnellement à la population existante :

$$\dot{x}(t) = (a - b x(t))x(t) = ax(t) - b x(t)^2, \quad (9)$$

$$x(0) = x_0. \quad (10)$$

C'est une équation de Bernoulli. On pose $z = \frac{1}{x}$. On trouve $\dot{z} + az = b$, d'où

$$x(t) = \frac{ax_0}{e^{-at} + bx_0(1 - e^{-at})}$$

Ainsi $\lim_{t \rightarrow +\infty} x(t) = \frac{a}{b}$: la population se stabilise.

C'est une équation de Bernoulli. On pose $z = \frac{1}{x}$. On trouve $\dot{z} + az = b$, d'où

$$x(t) = \frac{ax_0}{e^{-at} + bx_0(1 - e^{-at})}$$

Ainsi $\lim_{t \rightarrow +\infty} x(t) = \frac{a}{b}$: la population se stabilise.

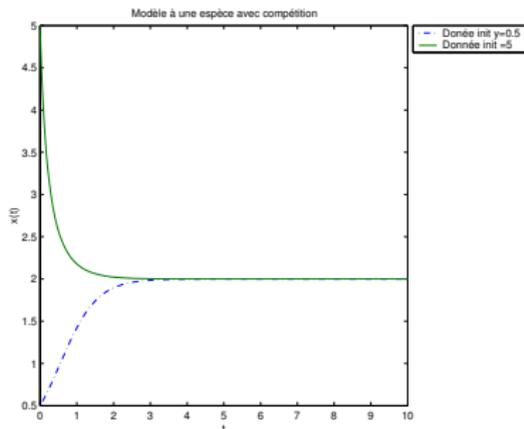


Figure – Modèle à une espèce avec compétition, $a = 2, b=1$.

Modèle à deux espèces - Equations de Lokta-Volterra

On considère à présent un modèle à deux populations (équation de Lokta-Volterra), de type "proies-prédateurs" :

- $x(t)$ est le nombre d'individus proies au temps t
- $y(t)$ est le nombre d'individus prédateurs au temps t

Sans interaction entre proies et prédateurs

Le modèle est :

$$\begin{cases} \frac{d}{dt}x(t) = a x, \\ \frac{d}{dt}y(t) = -c y \\ x(0) = x_0, y(0) = y_0. \end{cases}$$

En l'absence de prédateurs, la population des proies croit exponentiellement tandis que celle des prédateurs, en l'absence des proies, elle, décroît exponentiellement.

Modèle à deux espèces - Equations de Lokta-Volterra

On considère à présent un modèle à deux populations (équation de Lokta-Volterra), de type "proies-prédateurs" :

- $x(t)$ est le nombre d'individus proies au temps t
- $y(t)$ est le nombre d'individus prédateurs au temps t

Le couplage

Il repose sur la correction suivante : au taux de croissance (positif) de la population de proies x diminue proportionnellement au nombre de prédateurs ; le taux de croissance (négatif) de la population de proies augmente proportionnellement au nombre de proies On obtient :

$$\begin{cases} \frac{d}{dt}x(t) = x(a - by), \\ \frac{d}{dt}y(t) = y(-c + dx), \\ x(0) = x_0, y(0) = y_0. \end{cases}$$

Programme

```
function [f]=LoktaVoltera(t,u)
f(1)=1./2*u(1)*u(2)-u(1)./2
f(2)=-u(1)*u(2)+u(2)
endfunction
m=150;
u0=[2;2];
t=linspace(0,25,m);
u=ode(u0,0,t,LoktaVoltera);
subplot(1,3,1)
n=100;
deltax=5;
deltay=5;
x=linspace(-deltax,deltax,n);
y=linspace(-deltay,deltay,n);
figure(1)
subplot(1,3,1),plot2d(u(1,:)',u(2,:)',5,"121")
fchamp(LoktaVoltera,0,x,y,strf="000")
xlabel('Portrait de phase', 'y','x')
subplot(1,3,2),plot2d(t,u(1,:)',5)
xlabel('Prédateurs en projection', 't','x')
subplot(1,3,3),plot2d(t,u(2,:)',2)
xlabel('Proies en projection', 't','y')
```

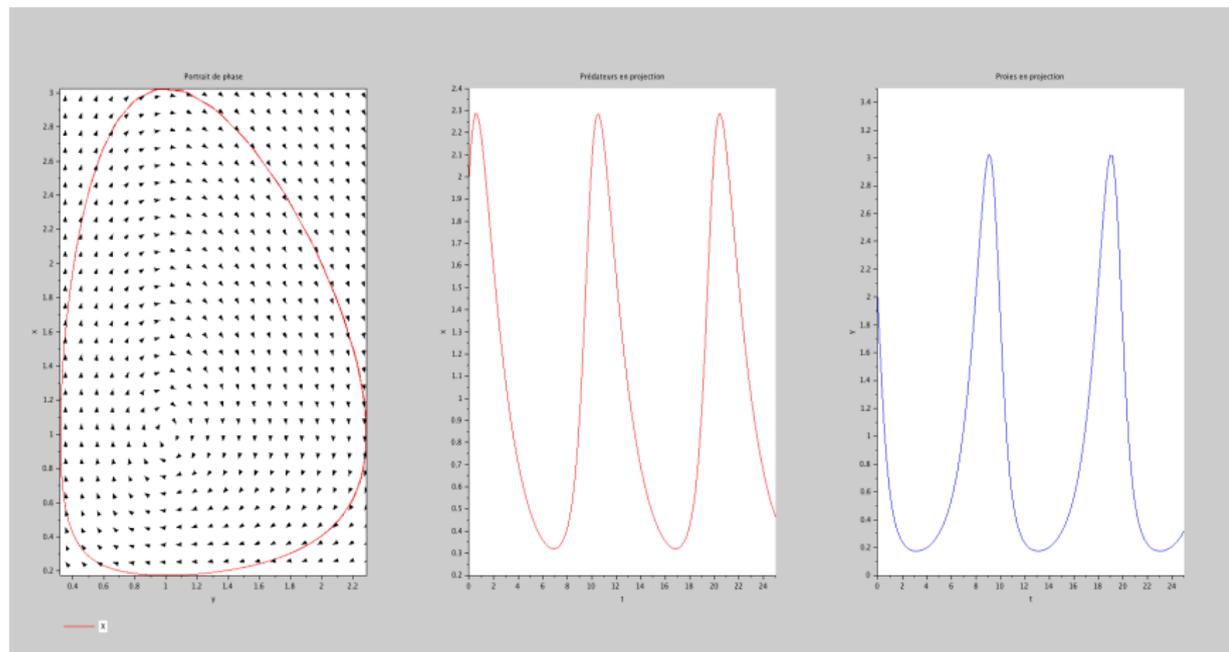


Figure – Modèle Proies-prédateurs

Il s'agit d'un modèle simplifié en météorologie, du à Edward Lorenz en 1963. On considère le système différentiel. Il met en évidence la très grande sensibilité aux conditions initiales et le caractère imprévisible des orbites. Le voici :

$$\frac{dx}{dt} = \sigma(y - x), \quad (11)$$

$$\frac{dy}{dt} = x(\rho - z) - y, \quad (12)$$

$$\frac{dz}{dt} = xy - \beta z. \quad (13)$$

Ici σ, ρ et β sont des constantes strictement positives ; on prend en général $\sigma = 10$ et $\beta = 8/3$, des cas particulièrement intéressants sont obtenus pour $\rho = 28$ et $\rho = 99.96$.

Programme

```
function dxdt=lorenzsys(t,x)
sigma=10
rho=28;
beta=8/3;
dxdt=[sigma*(x(2)-x(1));rho*x(1)-x(2)-x(1)*x(3);x(1)*x(2)-beta*x(3)];
endfunction
//
t=linspace(0,40,4000)';
X0=[-3 ; -6 ; 12]; X=ode(X0,0,t,lorenzsys);
param3d(X(1,:),X(2,:),X(3,:),5);
xlabel('x')
ylabel('y')
zlabel('z')
```

Système de Lorenz

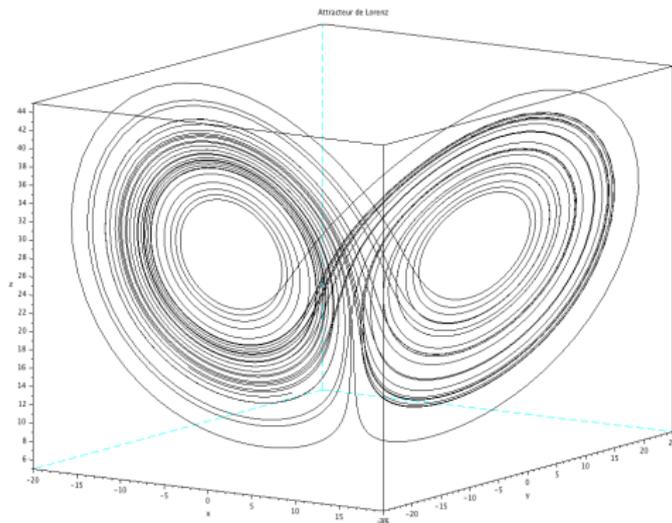


Figure – Système de Lorenz : $\sigma = 10, \beta = 8/3, \rho = 28$

Attracteur de Rössler

Il est dû à Otto Rössler en 1976. Il a été d'abord conçu dans un but purement théorique. Il a été montré plus tard qu'il avait un intérêt pour la modélisation de l'équilibre dans les réactions chimiques. Le voici :

$$\frac{dx}{dt} = -y - z, \quad (14)$$

$$\frac{dy}{dt} = x + ay, \quad (15)$$

$$\frac{dz}{dt} = b + z(x - c). \quad (16)$$

Ici a , b et c sont des paramètres réels ; les cas particulièrement intéressants sont $a = 0.1$, $b = 0.1$, $c = 14$ puis $a = 0.2$, $b = 0.2$ et $c = 5.7$.

Programme

```
function dxdt=rossler(t,x)
a=0.1; b=0.1; c=14; dxdt=[-x(2)-x(3);x(1)+a*x(2);b+x(3)*(x(1)-c)];
endfunction
//
t=linspace(0,400,20000)';
X0=[-3 ; -6 ; 12]; X=ode(X0,0,t,rossler);
param3d(X(1,:),X(2,:),X(3,:),5);
xlabel('x')
ylabel('y')
zlabel('z')
```

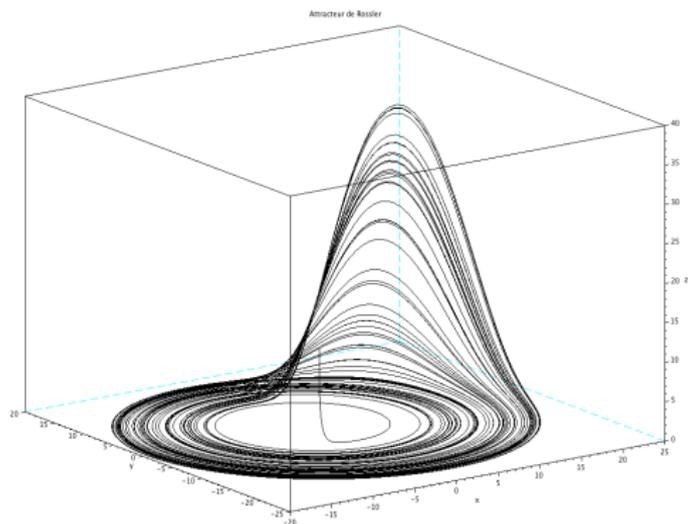


Figure – Système de Rössler : $a = 0.2$, $b = 0.1$, $c = 14$

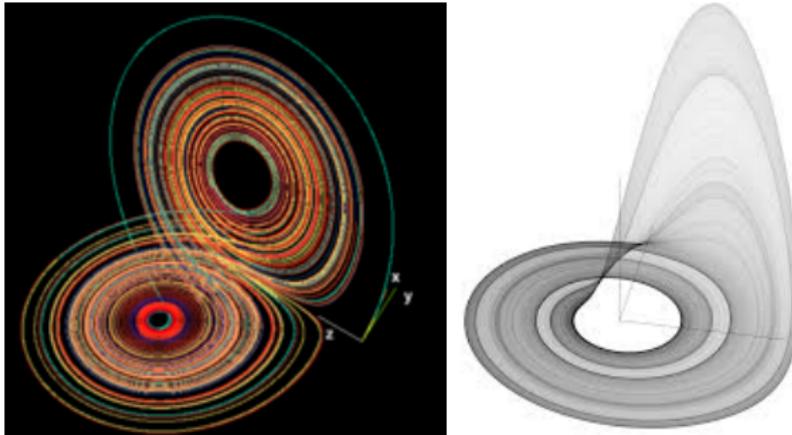


Figure – Systèmes de Lorenz et de Rössler

Système de Van der Pol

$$\frac{dx}{dt} = ky + mx(b - y^2), \quad (17)$$

$$\frac{dy}{dt} = -x + sz, \quad (18)$$

$$\frac{dz}{dt} = px - qy. \quad (19)$$

Il est particulièrement intéressant pour les valeurs
 $(k, m, b, s, p, q) = (0.02, 0.4, 0.2, 0.2, 10, 0.1)$ en partant de
 $(x_0, y_0, z_0) = (0.1, 0.1, 0.1)$. On fait alors varier le paramètre s .

Références : <https://images.math.cnrs.fr/Sculptures-du-chaos.html>

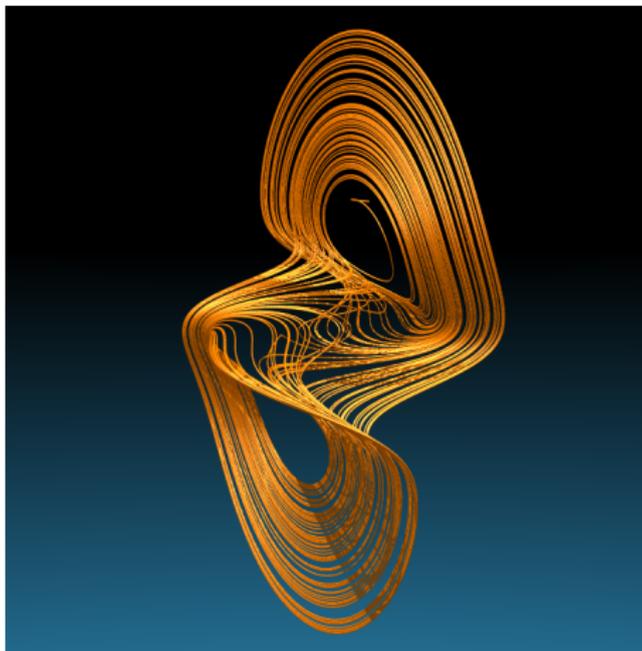


Figure – Système de Van der Pol : $s = 0.2$

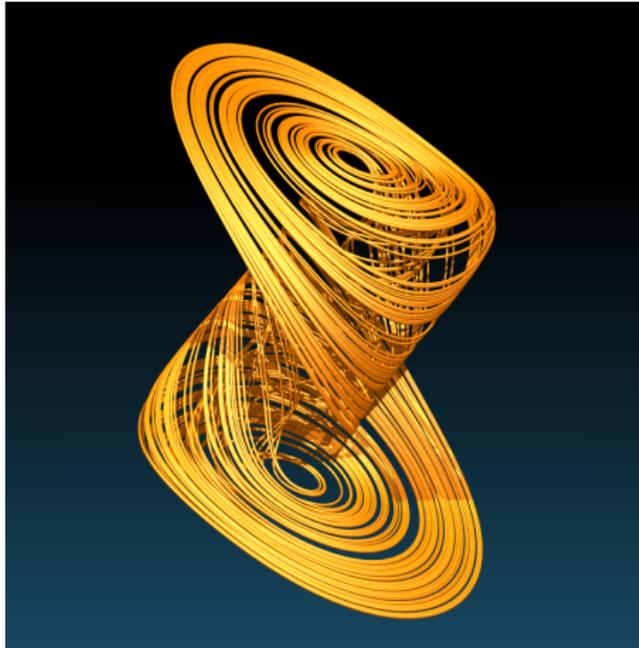


Figure – Système de Van der Pol : $s = 35$

Principe d'approximation

On se propose d'approcher la solution du problème de Cauchy

$$\begin{aligned}\frac{du}{dt} &= f(t, u), \quad t > 0 \\ u(0) &= u_0\end{aligned}$$

en des temps discrets $t_0 = 0 < t_1 < \dots < t_N = T$. On notera $u_k \approx u(t_k)$, $k = 1, \dots, N$. Nous avons deux approches

- Approximation de l'intégrale

$$u(t_{k+1}) = u(t_k) + \int_{t_k}^{t_{k+1}} f(u(t), t) dt,$$

On utilisera des formules de quadrature

- Approximation de la dérivée $\frac{du}{dt}$. On utilisera la formule de Taylor

Schémas d'Euler

Euler explicite

On peut l'obtenir comme suit :

- On utilise la méthode des rectangles **à gauche** :

$$\int_{t_k}^{t_{k+1}} g(t) dt \approx (t_{k+1} - t_k)g(t_k).$$

$$u(t_{k+1}) \approx u(t_k) + (t_{k+1} - t_k)f(t_k, u(t_k))$$

- On suppose $u \in \mathcal{C}^2$. La formule de Taylor donne $u(t_{k+1}) = u(t_k) + (t_{k+1} - t_k) \frac{du}{dt} \Big|_{t=t_k} + O((t_{k+1} - t_k)^2)$. Comme u est solution $\frac{du}{dt} \Big|_{t=t_k} = f(t_k, u(t_k))$. Ainsi

$$\frac{u(t_{k+1}) - u(t_k)}{t_{k+1} - t_k} \approx f(t_k, u(t_k)) \iff u(t_{k+1}) \approx u(t_k) + (t_{k+1} - t_k)f(t_k, u(t_k))$$

Schémas d'Euler

Euler explicite

On peut l'obtenir comme suit :

- On utilise la méthode des rectangles **à gauche** :

$$\int_{t_k}^{t_{k+1}} g(t) dt \approx (t_{k+1} - t_k)g(t_k).$$

$$u(t_{k+1}) \approx u(t_k) + (t_{k+1} - t_k)f(t_k, u(t_k))$$

- On suppose $u \in \mathcal{C}^2$. La formule de Taylor donne $u(t_{k+1}) = u(t_k) + (t_{k+1} - t_k) \frac{du}{dt} \Big|_{t=t_k} + O((t_{k+1} - t_k)^2)$. Comme u est solution $\frac{du}{dt} \Big|_{t=t_k} = f(t_k, u(t_k))$. Ainsi

$$\frac{u(t_{k+1}) - u(t_k)}{t_{k+1} - t_k} \approx f(t_k, u(t_k)) \iff u(t_{k+1}) \approx u(t_k) + (t_{k+1} - t_k)f(t_k, u(t_k))$$

Schéma obtenu - on approche $u(t_k)$ par u_k défini par récurrence :

$u_{k+1} = u_k + (t_{k+1} - t_k)f(t_k, u_k)$: c'est la **méthode d'Euler explicite**

Schémas d'Euler

Euler implicite

On peut l'obtenir comme suit :

- On utilise la méthode des rectangles **à droite** :

$$\int_{t_k}^{t_{k+1}} g(t) dt \approx (t_{k+1} - t_k)g(t_{k+1}).$$

$$u(t_{k+1}) \approx u(t_k) + (t_{k+1} - t_k)f(t_{k+1}, u(t_{k+1}))$$

- On suppose $u \in \mathcal{C}^2$. La formule de Taylor donne $u(t_k) = u(t_{k+1}) - (t_{k+1} - t_k) \frac{du}{dt} \Big|_{t=t_{k+1}} + O((t_{k+1} - t_k)^2)$. Comme u est solution $\frac{du}{dt} \Big|_{t=t_{k+1}} = f(t_k, u(t_{k+1}))$. Ainsi

$$\frac{u(t_{k+1}) - u(t_k)}{t_{k+1} - t_k} \approx f(t_{k+1}, u(t_{k+1})) \iff u(t_{k+1}) \approx u(t_k) + (t_{k+1} - t_k)f(t_{k+1}, u(t_{k+1}))$$

Schémas d'Euler

Euler implicite

On peut l'obtenir comme suit :

- On utilise la méthode des rectangles **à droite** :

$$\int_{t_k}^{t_{k+1}} g(t) dt \approx (t_{k+1} - t_k)g(t_{k+1}).$$

$$u(t_{k+1}) \approx u(t_k) + (t_{k+1} - t_k)f(t_{k+1}, u(t_{k+1}))$$

- On suppose $u \in \mathcal{C}^2$. La formule de Taylor donne $u(t_k) = u(t_{k+1}) - (t_{k+1} - t_k) \frac{du}{dt} \Big|_{t=t_{k+1}} + O((t_{k+1} - t_k))^2$. Comme u est solution $\frac{du}{dt} \Big|_{t=t_{k+1}} = f(t_k, u(t_{k+1}))$. Ainsi

$$\frac{u(t_{k+1}) - u(t_k)}{t_{k+1} - t_k} \approx f(t_{k+1}, u(t_{k+1})) \iff u(t_{k+1}) \approx u(t_k) + (t_{k+1} - t_k)f(t_{k+1}, u(t_{k+1}))$$

Schéma obtenu - on approche $u(t_k)$ par u_k défini par récurrence :

$u_{k+1} = u_k + (t_{k+1} - t_k)f(t_{k+1}, u_{k+1})$: c'est la **méthode d'Euler implicite**

Illustration $x' = ax$

L'équation différentielle la plus simple pour tester ces méthodes numériques est

$$x' = ax$$

avec $a < 0$. La solution $x(t) = e^{ta}$ tend vers 0 lorsque t tend vers l'infini.

Pour simplifier on considère le cas du pas de temps constant :

$t_{k+1} - t_k = \Delta t$, $\forall k \geq 0$ soit $t_k = k\Delta t$. Les schémas d'Euler s'écrivent

Schémas d'Euler

$$u_{k+1} = u_k + \Delta t f(k\Delta t, u_k) : \text{Euler explicite}$$

$$u_{k+1} = u_k + \Delta t f((k+1)\Delta t, u_{k+1}) : \text{Euler implicite}$$

Illustration $x' = ax$

L'équation différentielle la plus simple pour tester ces méthodes numériques est

$$x' = ax$$

avec $a < 0$. La solution $x(t) = e^{ta}$ tend vers 0 lorsque t tend vers l'infini.

Pour simplifier on considère le cas du pas de temps constant :

$t_{k+1} - t_k = \Delta t$, $\forall k \geq 0$ soit $t_k = k\Delta t$. Les schémas d'Euler s'écrivent

Schémas d'Euler

$u_{k+1} = u_k + \Delta t f(k\Delta t, u_k)$: Euler explicite

$u_{k+1} = u_k + \Delta t f((k+1)\Delta t, u_{k+1})$: Euler implicite

Questions : sur un intervalle de temps $[0, T]$

- Les approximations u_k tendent-elles vers 0 ? (Stabilité)
- Les résultats fournis par ode et par les schémas d'Euler sont-ils proches de la solution exacte ? (Consistance)
A-t-on $u_k \simeq u(k \Delta t)$
- Que rôle joue Δt ?

```
//---données numériques du schémas---  
a=-1;//valeur de a  
x0=1;//donnée initiale  
Tmax=4;//temps maximum  
N=20;//nombre de subdivisions  
dt=Tmax/N;//pas de temps  
t=0:dt:Tmax;//liste des temps discrets  
function [z]=f(t,x)  
a=-10;//valeur de a  
z=a*x;  
endfunction  
solex=exp(a*t)*x0;//solution exacte  
// mise en place de la méthode numérique  
X=[x0];Y=[x0]; Y=[x0]; //Euler implicite  
x=x0;//initialisation  
y=x0;//  
for k=1:N  
x=x+dt*(a*x); X=[X,x]; //Euler explicite  
y=y/(1-a*dt);Y=[Y,y]; //Euler implicite  
end  
z=ode(x0,0,t,f);// Solution via edo de Scilab  
clf();//---Post traitement (graphique)  
plot(t,X,'-ob',t,Y,'-dk',t,z,'*-g',t,solex,'-r')
```

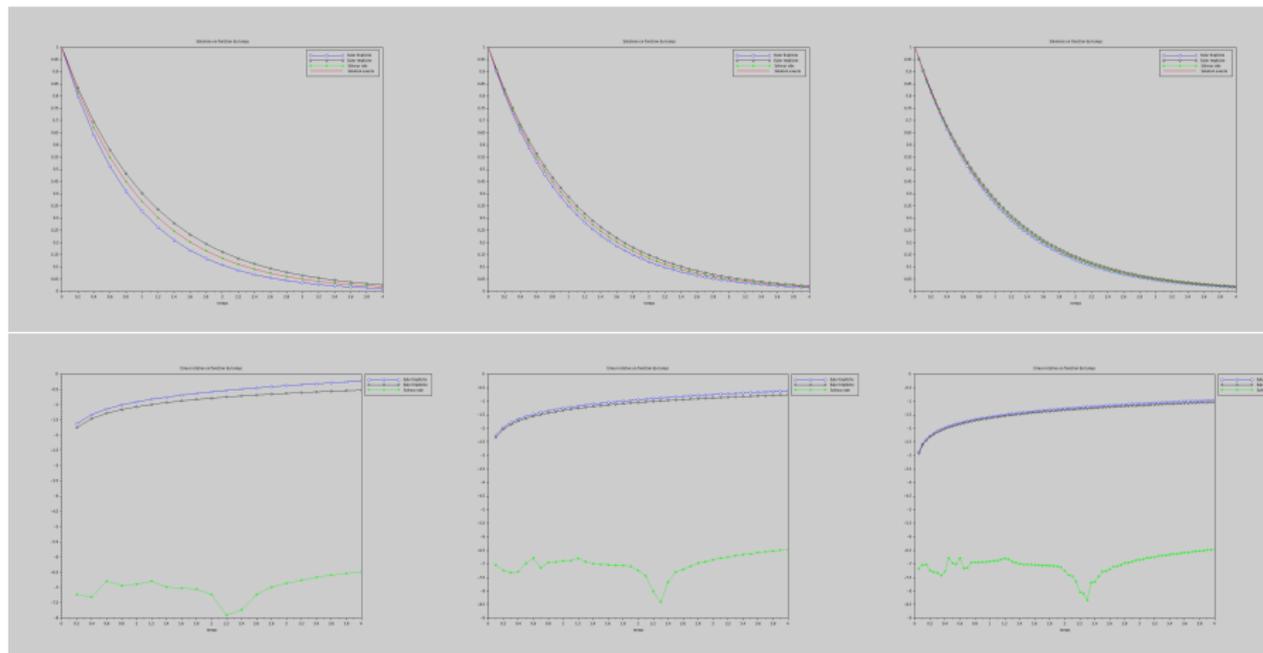


Figure – (haut) Solutions , (bas) Erreur relative, $a = -1$, $\Delta t = 0.2, 0.1, 0.05$

Explication

Précision : erreur de consistance

On remplace u_k par la solution exacte $u(t_k)$ dans le schéma

- Euler explicite :

$$\begin{aligned}\frac{u(t_{k+1}) - u(t_k)}{t_{k+1} - t_k} - f(t_k, u(t_k)) &= \frac{u(t_{k+1}) - u(t_k)}{t_{k+1} - t_k} - \frac{du}{dt} \Big|_{t=t_k} \\ &= \frac{1}{2}(t_{k+1} - t_k) \frac{d^2u}{dt^2} \Big|_{t=\theta_k} \\ &= O(\Delta t)\end{aligned}$$

- Euler implicite :

$$\begin{aligned}\frac{u(t_{k+1}) - u(t_k)}{t_{k+1} - t_k} - f(t_{k+1}, u(t_{k+1})) &= \frac{u(t_{k+1}) - u(t_k)}{t_{k+1} - t_k} - \frac{du}{dt} \Big|_{t=t_{k+1}} \\ &= -\frac{1}{2}(t_{k+1} - t_k) \frac{d^2u}{dt^2} \Big|_{t=\eta_k} \\ &= O(\Delta t)\end{aligned}$$

Stabilité ($a < 0$)

- Euler explicite :

$$u_{k+1} = (1 + \Delta t a)u_k : u_k = (1 + \Delta t a)^k u_0,$$

$$\lim_{k \rightarrow +\infty} u_k = 0 \iff |1 - |a|\Delta t| < 1 \iff 0 < \Delta t < \frac{2}{|a|}$$

- Euler implicite :

$$u_{k+1} = \frac{1}{1 - \Delta t a} u_k : u_k = \frac{1}{(1 + \Delta t a)^k} u_0,$$

$$\lim_{k \rightarrow +\infty} u_k = 0 \quad \forall \Delta t > 0$$