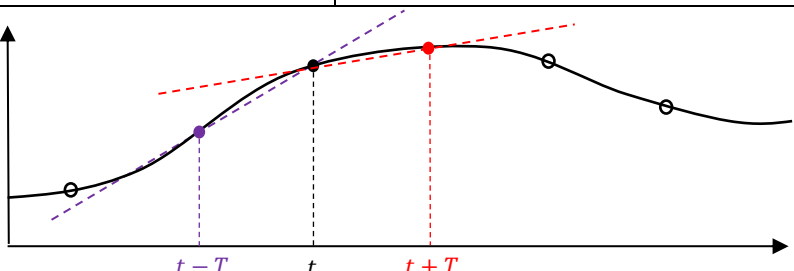
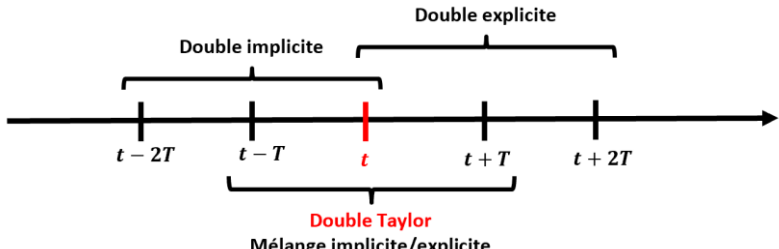


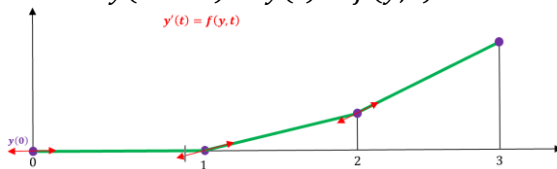
Informatique

Dérivation numérique
Méthode d'Euler

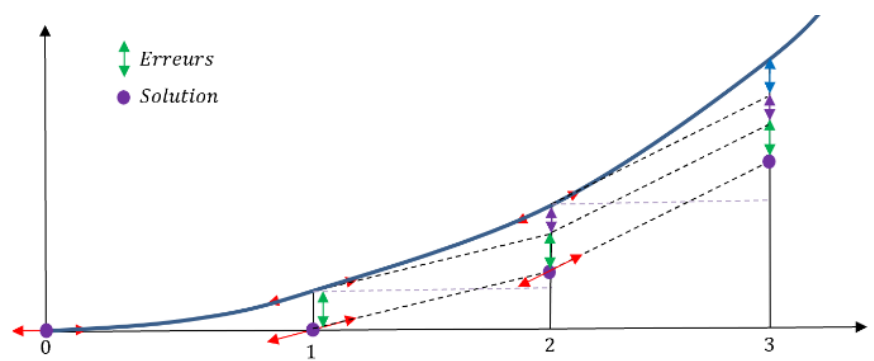
Résumé

	Dérivée première	
Type d'Euler	Explicite	Implicite
Expression de $y'(t)$	$y'(t) = \frac{y(t+T) - y(t)}{T}$	$y'(t+T) = \frac{y(t+T) - y(t)}{T}$ $y'(t) = \frac{y(t) - y(t-T)}{T}$
Nouvelle valeur $y(t+T)$ lors d'une résolution	$y(t+T) = Ty'(t) + y(t)$	$y(t+T) = Ty'(t+T) + y(t)$
Bilan		
Calcul de la nouvelle valeur	$y(t+T) = Tf(y(t), t) + y(t)$	$y(t+T) = Tf(y(t+T), t+T) + y(t)$ Dans f , $y(t+T)$ est inconnu ⊗ Approximation dans f : $y(t+T) \approx y(t)$ $y(t+T) = Tf(y(t), t+T) + y(t)$

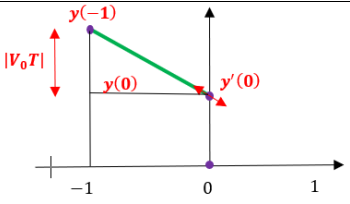
	Dérivée seconde
Méthode à privilégier Double Taylor à l'ordre 2	$y(t+T) = y(t) + T \frac{dy(t)}{dt} + \frac{T^2}{2} \frac{d^2y(t)}{dt^2} + o(T^2)$ $y(t-T) = y(t) - T \frac{dy(t)}{dt} + \frac{T^2}{2} \frac{d^2y(t)}{dt^2} + o(T^2)$ $y(t+T) + y(t-T) = 2y(t) + T^2 \frac{d^2y(t)}{dt^2}$ $\frac{d^2y(t)}{dt^2} = \frac{y(t+T) - 2y(t) + y(t-T)}{T^2}$
Double Euler explicite	$\frac{d^2y(t)}{dt^2} = \frac{y(t+2T) - 2y(t+T) + y(t)}{T^2}$
Double Euler implicite	$\frac{d^2y(t)}{dt^2} = \frac{y(t) - 2y(t-T) + y(t-2T)}{T^2}$
Euler implicite sur la dérivée seconde et explicite sur la dérivée première	$\frac{d^2y(t)}{dt^2} = \frac{y(t+T) - 2y(t) + y(t-T)}{T^2}$
Explicite sur la dérivée seconde et implicite sur la dérivée première	$\frac{d^2y(t)}{dt^2} = \frac{y(t+T) - 2y(t) + y(t-T)}{T^2}$
Bilan	

Une équation différentielle	
$y^{(i)}(t) = f(y^{(i-1)}(t), y^{(i-2)}(t), \dots, y(t), t)$	
1° ordre	2° ordre
$y'(t) = f(y, t)$ $y'(t) = \frac{y(t+dt) - y(t)}{dt}$ $y(t+dt) = y(t) + y'(t)dt$ $y(t+dt) = y(t) + f(y, t)dt$ 	$y''(t) = f(y(t), y'(t), t) = f(Y, t)$ $Y(t) = \begin{pmatrix} y(t) \\ y'(t) \end{pmatrix}$ $Y'(t) = \begin{pmatrix} y'(t) \\ y''(t) \end{pmatrix} = \begin{pmatrix} y'(t) \\ f(Y, t) \end{pmatrix} = F(Y, t)$ $F: (Y, t) \mapsto (y', f(Y, t))$ $Y(t_0) = \begin{pmatrix} y(t_0) \\ y'(t_0) \end{pmatrix}$ $Y(t+dt) = Y(t) + F(Y, t)dt$
Exemples	
<p>Chute libre</p> $m \frac{d^2 z(t)}{dt^2} + \mu \frac{dz(t)}{dt} = mg$ $\ddot{z}(t) = g - \frac{\mu}{m} \dot{z}(t)$ $Z(t) = \begin{pmatrix} z(t) \\ \dot{z}(t) \end{pmatrix} Z'(t) = \begin{pmatrix} \dot{z}(t) \\ \ddot{z}(t) \end{pmatrix} = \begin{pmatrix} \dot{z}(t) \\ f(Z, t) \end{pmatrix}$ $f(Z, t) = g - \frac{\mu}{m} \dot{z}(t)$	<p>Pendule</p> $-mgl \sin(\theta(t)) = ml^2 \ddot{\theta}(t)$ $\ddot{\theta}(t) = -\frac{g}{l} \sin(\theta(t))$ $\theta(t) = \begin{pmatrix} \theta(t) \\ \dot{\theta}(t) \end{pmatrix} \dot{\theta}(t) = \begin{pmatrix} \dot{\theta}(t) \\ \ddot{\theta}(t) \end{pmatrix} = \begin{pmatrix} \dot{\theta}(t) \\ f(\theta, t) \end{pmatrix}$ $f(\theta, t) = -\frac{g}{l} \sin(\theta(t))$

Système d'équations différentielles	
1° ordre	2° ordre
$\begin{cases} \frac{df_1(t)}{dt} = g_1(f_1(t), f_2(t), \dots, f_n(t), t) \\ \frac{df_2(t)}{dt} = g_2(f_1(t), f_2(t), \dots, f_n(t), t) \\ \vdots \\ \frac{df_n(t)}{dt} = g_n(f_1(t), f_2(t), \dots, f_n(t), t) \end{cases}$ $Y(t) = \begin{pmatrix} f_1(t) \\ f_2(t) \\ \vdots \\ f_n(t) \end{pmatrix} \Rightarrow \begin{cases} \frac{df_1(t)}{dt} = g_1(Y, t) \\ \frac{df_2(t)}{dt} = g_2(Y, t) \\ \vdots \\ \frac{df_n(t)}{dt} = g_n(Y, t) \end{cases}$ $F_p: (Y, t) \mapsto \begin{pmatrix} f_1'(t) \\ f_2'(t) \\ \vdots \\ f_n'(t) \end{pmatrix} = \begin{pmatrix} g_1(Y, t) \\ g_2(Y, t) \\ \vdots \\ g_n(Y, t) \end{pmatrix}$ $Y'(t) = F_p(Y, t)$ $Y(t+dt) = Y(t) + F_p(Y, t)dt$	$\begin{cases} \frac{d^2 f_1(t)}{dt^2} = g_1(f_1, f_1', f_2, f_2', \dots, f_n, f_n', t) \\ \frac{d^2 f_2(t)}{dt^2} = g_2(f_1, f_1', f_2, f_2', \dots, f_n, f_n', t) \\ \vdots \\ \frac{d^2 f_n(t)}{dt^2} = g_n(f_1, f_1', f_2, f_2', \dots, f_n, f_n', t) \end{cases}$ $Y(t) = \begin{pmatrix} f_1(t) \\ f_1'(t) \\ f_2(t) \\ f_2'(t) \\ \vdots \\ f_n(t) \\ f_n'(t) \end{pmatrix} \Rightarrow \begin{cases} \frac{d^2 f_1(t)}{dt^2} = g_1(Y, t) \\ \frac{d^2 f_2(t)}{dt^2} = g_2(Y, t) \\ \vdots \\ \frac{d^2 f_n(t)}{dt^2} = g_n(Y, t) \end{cases}$ $F_p: (Y, t) \mapsto \begin{pmatrix} f_1'(t) \\ f_1''(t) \\ f_2'(t) \\ f_2''(t) \\ \vdots \\ f_n'(t) \\ f_n''(t) \end{pmatrix} = \begin{pmatrix} f_1'(t) \\ g_1(Y, t) \\ f_2'(t) \\ g_2(Y, t) \\ \vdots \\ f_n'(t) \\ g_n(Y, t) \end{pmatrix}$ $Y'(t) = F_p(Y, t)$ $Y(t+dt) = Y(t) + F_p(Y, t)dt$

<p>Programmation de la méthode sous python</p> <p>Eviter les cumuls d'erreurs d'arrondis « $t += dt$ »</p> <p>Ne pas mettre à jour y mais bien écrire « $y =$ » sinon unique y stocké dans Y</p>	<pre>def Euler_Explicite(f,y0,t0,t1,dt): t,y = t0,y0 T,Y = [t],[y] i = 0 while t < t1: yp = f(y,t) n = len(y) y = y + yp*dt # 1^{er} ordre Y = [Y[i] + yp[i]*dt for i in range(n)] # Sinon i += 1 t = t0 + i*dt T.append(t) Y.append(y) return T,Y</pre> <div> <pre>def f(y,t): # 1^{er} ordre yp = ... return yp</pre> <pre>def f(Y,t): # 2^{er} ordre y,yp = Y ypp = ... return ypp</pre> <pre>def F(Y,t): yp = Y[1] ypp = f(Y,t) Sol = [yp,ypp] return Sol</pre> </div>
<p>Remarques</p>	<p>Les erreurs se propagent</p>  <p>Plus les pas (dt, $dx...$) sont faibles, plus la solution est correcte mais plus les temps de calcul sont longs</p> <p>Préférer le calcul absolu (du temps par exemple) : $t = t0 + i * dt$ au calcul relatif qui cumule les erreurs d'arrondi en machine : $t = t + dt$</p>

Résolutions avec Odeint	
1° ordre - 1 équation	
$-mg + ky^2(t) = my'(t)$ $y'(t) = \frac{k}{m}y^2(t) - g$	<pre>def F(Y,t): k,g,m = 0.001,9.81,0.1 dy = (k/m)*(y**2)-g return dy</pre> <p>Y0 = 0</p>
1° ordre - 2 équations	
$\begin{cases} \frac{dH(t)}{dt} = aH(t) - bA(t)H(t) & (1) \\ \frac{dA(t)}{dt} = -cA(t) + dA(t)H(t) & (2) \end{cases}$ $Y(t) = \begin{pmatrix} H(t) \\ A(t) \end{pmatrix}$	<pre>def F(Y,t): a,b,c,d = 3,1,1,2 H,A = Y dH = a*H - b*A*H dA = -c*A + d*A*H dY = [dH,dA] return dY</pre> <p>H0,A0 = 10.0,1.0 Y0 = [H0,A0]</p>
2° ordre - 1 équation	
$\ddot{\theta}(t) = \frac{-k\dot{\theta}(t) - Rmg \sin \theta(t)}{J}$ $Y(t) = \begin{pmatrix} \theta(t) \\ \dot{\theta}(t) \end{pmatrix}$	<pre>def f(Y,t): m,R,k,g = 0.1,0.1,0.001,9.81 J = m*R**2 Y,yp = Y ypp = (1/J)*(-k*yp- R*m*g*sin(y)) return ypp def F(V,t): Y,yp = V ypp = f(V,t) dY = [yp,ypp] return dY</pre> <p>from math import pi,sin y0,yp0 = 45*pi/180,0 Y0 = [y0,yp0]</p>
2° ordre - 2 équations	
$\begin{cases} \frac{d^2x(t)}{dt^2} = -\frac{\beta}{m} \frac{dx(t)}{dt} \sqrt{\frac{dx(t)^2}{dt} + \frac{dy(t)^2}{dt}} \\ \frac{d^2y(t)}{dt^2} = -g - \frac{\beta}{m} \frac{dy(t)}{dt} \sqrt{\frac{dx(t)^2}{dt} + \frac{dy(t)^2}{dt}} \end{cases}$ $Y(t) = \begin{pmatrix} x(t) \\ x'(t) \\ y(t) \\ y'(t) \end{pmatrix}$	<pre>def f(Y,t): m,B,g = 1,1,9.81 x,xp,y,yp = Y Vit = (xp**2+yp**2)**(0.5) xpp = (1/m)*(-B*Vit*xp) ypp = (1/m)*(-g-B*Vit*yp) return xpp,ypp def F(Y,t): x,xp,y,yp = Y xpp,ypp = f(Y,t) dY = [xp,xpp,y,ypp] return dY</pre> <p>x0,y0 = 0,0 xp0,yp0 = 10,10 Y0 = [x0,xp0,y0,yp0]</p>
Code de résolution	
<pre>t0,t1 = 0,10 N = 1000 dt = (t1-t0)/(N-1) Lt = [t0+i*dt for i in range(N)] from scipy.integrate import odeint Sol = odeint(F,Y0,Lt) print(Sol)</pre>	

Equations aux différences	
Principe	<p>Possibilité de résoudre des équations du type :</p> $\frac{\partial T(t, x)}{\partial t} = D_{th} \frac{\partial^2 (T(t, x))}{\partial x^2}$ <p>Donc, des équations qui lient les dérivées de variables selon plusieurs paramètres, comme le temps et l'espace dans l'équation de la chaleur L'objectif est de discrétiser toutes les dérivées puis d'exprimer :</p> $s(t + dt) = f(t, s(t), s(t - dt), s(t - 2dt) \dots, e(t), e(t - dt), e(t - 2dt) \dots)$ <p>Solution dans laquelle on trouve s à l'instant d'après $t + dt$, connaissant e et s à l'instant présent t et aux instants précédents</p>
Exemple Chute libre Un seul paramètre t $ma(t) = mg - \mu v(t)$	$m \frac{d^2 z(t)}{dt^2} + \mu \frac{dz(t)}{dt} = mg$ $\frac{dz(t)}{dt} = \frac{z(t + T) - z(t)}{T}$ $\frac{d^2 z(t)}{dt^2} = \frac{z(t + T) - 2z(t) + z(t - T))}{T^2}$ $m \frac{z(t + T) - 2z(t) + z(t - T))}{T^2} + \mu \frac{z(t + T) - z(t)}{T} = mg$ $z(t + T) = \frac{2m + \mu T}{m + \mu T} z(t) - \frac{m}{m + \mu T} z(t - T) + \frac{T^2}{m + \mu T} mg$ $\frac{d^2 z(t)}{dt^2} = \frac{A}{m} ; \quad A = mg - \mu v(t)$ <p>Calculer $v(t)$ à chaque pas de temps avec Euler implicite :</p> $v(t) = \frac{dz(t)}{dt} = \frac{z(t) - z(t - T)}{T}$ <p>Cela permet de calculer la valeur de A pour chaque pas de temps, puis :</p> $\frac{z(t + T) - 2z(t) + z(t - T))}{T^2} = \frac{A}{m}$ $z(t + T) = \frac{AT^2}{m} + 2z(t) - z(t - T)$ <p>Très intéressant si par exemple, $A = mg - \mu v(t)^2 \dots$</p>
Prise en compte des conditions initiales	 <p>Traiter le/les premier(s) pas de temps de manière particulière en imposant des valeurs aux variables en temps négatif (ex $z(0 - T)$) permettant d'obtenir la condition initiale voulue.</p> <p>Exemple :</p> $\frac{dz(t_0)}{dt} = V_0 = \frac{z(t_0) - z(t_0 - T)}{T}$ $z(t_0 - T) = z(t_0) - V_0 T$