

Informatique

Méthode d'Euler

Cours

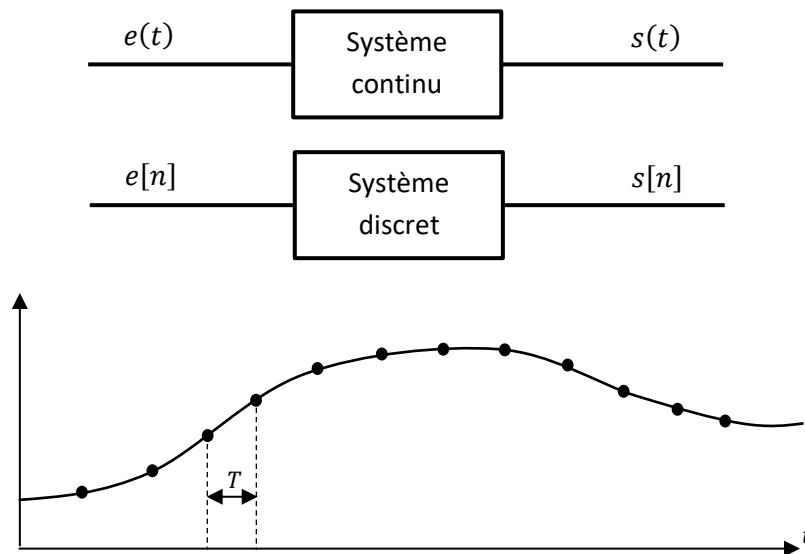
Méthode d'Euler	4
1.I. Discrétisation des problèmes	4
1.II. Dérivation numérique	5
1.II.1 Dérivée première	5
1.II.1.a Euler explicite : dérivée à droite	5
1.II.1.a.i Approximation de la dérivée	5
1.II.1.a.ii Résolution numérique Euler explicite	6
1.II.1.b Euler implicite : dérivée à gauche	6
1.II.1.b.i Approximation de la dérivée	6
1.II.1.b.ii Résolution numérique Euler implicite	7
1.II.1.c Bilan	7
1.II.2 Dérivée seconde	8
1.II.2.a Taylor à l'ordre 2	8
1.II.2.b Double Euler explicite	9
1.II.2.c Double Euler implicite	9
1.II.2.d Mélange implicite – explicite	10
1.II.2.e Conclusion	10
1.II.3 Dérivées d'ordre supérieurs	10
1.II.4 Précision des solutions	10
1.III. Equations différentielles simples du premier et second ordre	11
1.III.1 1° ordre	11
1.III.2 2° ordre	12
1.III.2.a Méthode	12
1.III.2.b Remarques	13
1.III.2.b.i Concernant les array	13
1.III.2.b.ii Conditions initiales atypiques	13
1.III.2.c Exemples de mise en équation	14
1.III.2.c.i Exemple 1 : Masse en chute libre	14
1.III.2.c.ii Exemple 2 : Pendule	14
1.IV. Systèmes d'équations différentielles du premier et du second ordre	15
1.IV.1 1° ordre	15
1.IV.2 2° ordre	16
1.V. Modélisation par équations aux différences	17
1.V.1 Contexte	17
1.V.2 Méthode de résolution	17
1.V.3 Exemple 1 : Chute libre	18
1.V.3.a Equation récurrente	18
1.V.3.b Prise en compte des conditions initiales	19
1.V.3.c Remarque : Regroupement de termes dépendant de z dans le second membre	19
1.V.4 Exemple 2 : Pendule	20
1.V.4.a Equation récurrente	20
1.V.4.b Prise en compte des conditions initiales	20
1.VI. Résolution d'équations et précision	21
1.VI.1 Précision sur la solution	21
1.VI.2 Précision sur le temps	22
1.VII. Fonction de résolution intégrée à Python	23

Méthode d'Euler

1.1. Discrétisation des problèmes

Jusqu'ici, nous avons étudié des systèmes continus. Les variables traitées étaient des fonctions continues du temps.

Dans bon nombre de systèmes, les variables sont échantillonnées et on ne connaît qu'une liste de valeurs à différents temps. On parle de variables discrètes.



Cette figure représente un signal continu en fonction du temps sur lequel ont été prises des valeurs à différents temps espacés d'un temps T supposé ici constant et appelé « période d'échantillonnage ».

1.II. Dérivation numérique

Soit la variable discrète y .

Voyons comment déterminer une approximation de la dérivée de y . On parle de différences finies, de méthode d'Euler, de développement de Taylor. Voyons par ailleurs comment réaliser une résolution numérique d'une équation différentielle à l'aide de ces approximations de dérivées.

1.II.1 Dérivée première

1.II.1.a Euler explicite : dérivée à droite

1.II.1.a.i Approximation de la dérivée

Proposons le développement de Taylor de la variable y à l'ordre 1

$$y(t + T) = y(t) + T \frac{dy(t)}{dt} + o(T)$$

On a donc :

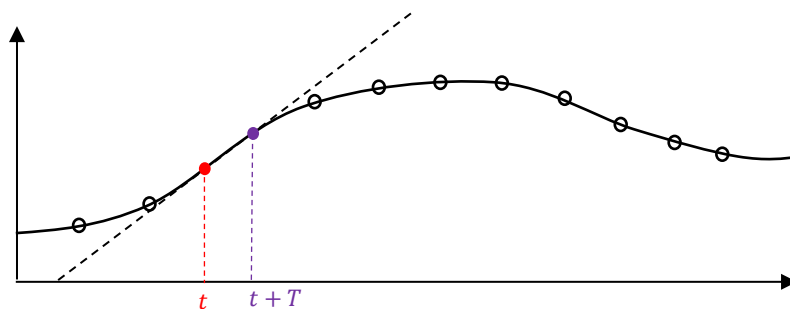
$$y(t + T) \approx y(t) + T \frac{dy(t)}{dt}$$

$$T \frac{dy(t)}{dt} = y(t + T) - y(t)$$

$$\frac{dy(t)}{dt} = \frac{y(t + T) - y(t)}{T}$$

Cette approximation de la dérivée correspond à une méthode Euler Explicite.

Cela revient à approcher la dérivée à un instant t en utilisant la valeur à l'instant t et la valeur à l'instant $t + dt$



$$y'(t) = \frac{y(t + T) - y(t)}{T}$$

La dérivée à l'instant t est approchée à l'aide de la valeur en t et **après**.

Remarque : Méthode nécessaire pour calculer la dérivée d'une liste de valeurs en son **premier point** !

1.II.1.a.ii Résolution numérique Euler explicite

A l'aide de l'approximation précédente, on a : $y(t + T) = Ty'(t) + y(t)$. Il est possible d'approcher la prochaine valeur $y(t + T)$ en fonction de la valeur actuelle $y(t)$ connue et de la dérivée $y'(t)$ à l'instant t connue. On dit que ce calcul est explicite car dépendant du temps actuel t , ou tout est connu.

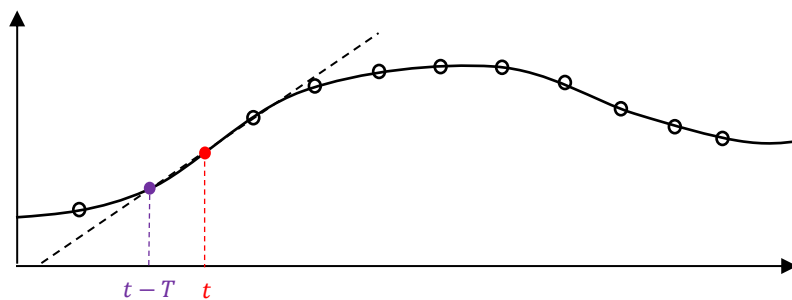
Que la dérivée de y ne dépende que du temps ($y' = f(t)$) ou du temps et de y ($y'(t) = f(y(t), t)$), on peut réaliser le calcul de $y(t + T) = Tf(y(t), t) + y(t)$ car tout est connu au temps t 😊.

1.II.1.b Euler implicite : dérivée à gauche

1.II.1.b.i Approximation de la dérivée

On peut aussi associer la dérivée en $t + dt$ à cette expression :

$$y'(t + T) = \frac{y(t + T) - y(t)}{T}$$



Cela revient à écrire :

$$y'(t) = \frac{y(t) - y(t - T)}{T}$$

Cette approximation de la dérivée correspond à une méthode Euler implicite.

La dérivée à l'instant t est approchée à l'aide de la valeur en t et **avant**.

Remarque : Méthode nécessaire pour calculer la dérivée d'une liste de valeurs en son **dernier point** !

1.II.1.b.ii Résolution numérique Euler implicite

A l'aide de l'approximation précédente, on a : $y(t + T) = Ty'(t + T) + y(t)$. La prochaine valeur $y(t + T)$ s'exprime en fonction de la valeur actuelle $y(t)$ connue et de la dérivée $y'(t + T)$ à l'instant $t + T$, qui peut être connue... ou non ! On dit que ce calcul est implicite car dépendant du temps suivant $t + T$, où tout n'est pas forcément connu...

Lorsque la dérivée de y ne dépend que du temps ($y' = f(t)$), il est possible de réaliser le calcul de $y(t + T)$ car f est connue.

Cependant, lorsque la dérivée dépend aussi de la valeur de y ($y'(t) = f(y(t), t)$), il n'est pas possible de calculer $y(t + T) = Tf(y(t + T), t + T) + y(t)$ car $y(t + T)$ est inconnu. Dans ce cas, on peut faire l'approximation $y(t + T) \approx y(t)$ afin d'estimer $y(t + T) = Tf(y(t), t + T) + y(t)$.

Les résolutions sont très généralement réalisées avec la méthode d'Euler explicite. Euler implicite sera plutôt utilisé pour calculer la dérivée d'une fonction représentée de manière discrète dont les valeurs sont connues à tout instant et lorsque l'on se trouve à son dernier point. En effet, $y(t + T)$ n'existant pas, on calculera $y'(t) = \frac{y(t) - y(t-T)}{T}$.

1.II.1.c Bilan

Euler explicite	Euler implicite
$y'(t) = \frac{y(t + T) - y(t)}{T}$	$y'(t + T) = \frac{y(t + T) - y(t)}{T}$ $y'(t) = \frac{y(t) - y(t - T)}{T}$
$y(t + T) = Ty'(t) + y(t)$ <p>Nouvelle valeur dépendant de la dérivée au temps présent Explicite</p>	$y(t + T) = Ty'(t + T) + y(t)$ <p>Nouvelle valeur dépendant de dérivée au temps suivant, pas forcément calculable Implicite</p>

1.II.2 Dérivée seconde

On trouve plusieurs méthodes pour estimer la dérivée seconde d'une variable discrète.

1.II.2.a Taylor à l'ordre 2

Proposons deux développements de Taylor de la variable y à l'ordre 2

$$y(t + T) = y(t) + T \frac{dy(t)}{dt} + \frac{T^2}{2} \frac{d^2y(t)}{dt^2} + o(T^2)$$

$$y(t - T) = y(t) - T \frac{dy(t)}{dt} + \frac{T^2}{2} \frac{d^2y(t)}{dt^2} + o(T^2)$$

Faisons la somme de ces deux expressions :

$$y(t + T) + y(t - T) = 2y(t) + T^2 \frac{d^2y(t)}{dt^2}$$

Soit :

$$\frac{d^2y(t)}{dt^2} = \frac{y(t + T) - 2y(t) + y(t - T)}{T^2}$$

Cette approximation calcule la dérivée seconde de manière centrée autour du temps t

1.II.2.b Double Euler explicite

Ecrivons l'expression de la dérivée seconde de y avec la méthode d'Euler Explicite vue précédemment :

$$\frac{d^2y(t)}{dt^2} = \frac{\frac{dy(t+T)}{dt} - \frac{dy(t)}{dt}}{T}$$

On a par ailleurs :

$$\frac{dy(t)}{dt} = \frac{y(t+T) - y(t)}{T}$$

Soit

$$\frac{d^2y(t)}{dt^2} = \frac{\frac{y(t+2T) - y(t+T)}{T} - \frac{y(t+T) - y(t)}{T}}{T}$$

$$\frac{d^2y(t)}{dt^2} = \frac{y(t+2T) - y(t+T) - y(t+T) + y(t)}{T^2}$$

$$\frac{d^2y(t)}{dt^2} = \frac{y(t+2T) - 2y(t+T) + y(t)}{T^2}$$

On remarque ici que l'approximation de la dérivée est faite avec les valeurs en t et après.

1.II.2.c Double Euler implicite

Ecrivons l'expression de la dérivée seconde de y avec la méthode d'Euler Explicite vue précédemment :

$$\frac{d^2y(t)}{dt^2} = \frac{\frac{dy(t)}{dt} - \frac{dy(t-T)}{dt}}{T}$$

On a par ailleurs :

$$\frac{dy(t)}{dt} = \frac{y(t) - y(t-T)}{T}$$

Soit

$$\frac{d^2y(t)}{dt^2} = \frac{\frac{y(t) - y(t-T)}{T} - \frac{y(t-T) - y(t-2T)}{T}}{T}$$

$$\frac{d^2y(t)}{dt^2} = \frac{y(t) - 2y(t-T) + y(t-2T)}{T^2}$$

On remarque ici que l'approximation de la dérivée est faite avec les valeurs en t et avant.

1.II.2.d Mélange implicite – explicite

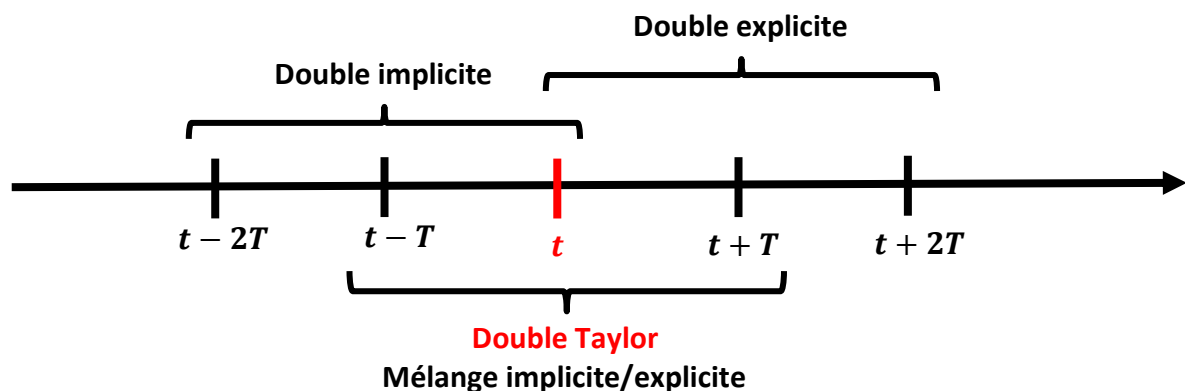
On peut aussi mélanger les méthodes :

Implicite sur la dérivée seconde et explicite sur la dérivée première	Explicite sur la dérivée seconde et implicite sur la dérivée première
$\frac{d^2 y(t)}{dt^2} = \frac{\frac{dy(t+T)}{dt} - \frac{dy(t)}{dt}}{T}$ $\frac{dy(t)}{dt} = \frac{y(t) - y(t-T)}{T}$ $\frac{d^2 y(t)}{dt^2} = \frac{\frac{y(t+T) - y(t)}{T} - \frac{y(t) - y(t-T)}{T}}{T}$ $\frac{d^2 y(t)}{dt^2} = \frac{y(t+T) - 2y(t) + y(t-T)}{T^2}$	$\frac{d^2 y(t)}{dt^2} = \frac{\frac{dy(t)}{dt} - \frac{dy(t-T)}{dt}}{T}$ $\frac{dy(t)}{dt} = \frac{y(t+T) - y(t)}{T}$ $\frac{d^2 y(t)}{dt^2} = \frac{\frac{y(t+T) - y(t)}{T} - \frac{y(t) - y(t-T)}{T}}{T}$ $\frac{d^2 y(t)}{dt^2} = \frac{y(t+T) - 2y(t) + y(t-T)}{T^2}$

Dans les deux cas, on retrouve l'approximation centrée issue de l'application de deux développements de Taylor à l'ordre 2.

1.II.2.e Conclusion

La dérivée en t s'exprime en fonction de valeurs de la fonction en :



On veut généralement trouver $f(t+T)$ connaissant $f(t)$ et $f(t-T)$. On préférera donc écrire proprement un double développement de Taylor

1.II.3 Dérivées d'ordre supérieurs

On pourra procéder de la même manière pour obtenir des dérivées d'ordres supérieurs

1.II.4 Précision des solutions

Plus T sera petit, plus les résultats seront proches de la réalité, la dérivée réelle étant la limite de nos formules quand T tend vers 0...

1.III. Equations différentielles simples du premier et second ordre

Nous allons dans un premier temps voir une méthode permettant de résoudre des équations du type :

$$y^{(i)}(t) = f(y^{(i-1)}(t), y^{(i-2)}(t), \dots, y(t), t)$$

On suppose connaître les conditions initiales.

On pourrait par exemple résoudre $y'(t) + y(t) = \cos(t)$, mais aussi des équations plus complexes comme $y''(t) = y(t) * y'(t) * \cos(t)$

1.III.1 1° ordre

Soit l'équation :

$$y'(t) = f(y, t)$$

Discretisons $y'(t)$ avec la méthode d'Euler explicite :

$$y'(t) = \frac{y(t + dt) - y(t)}{dt}$$

On a donc :

$$y(t + dt) = y(t) + y'(t)dt$$

Ou encore :

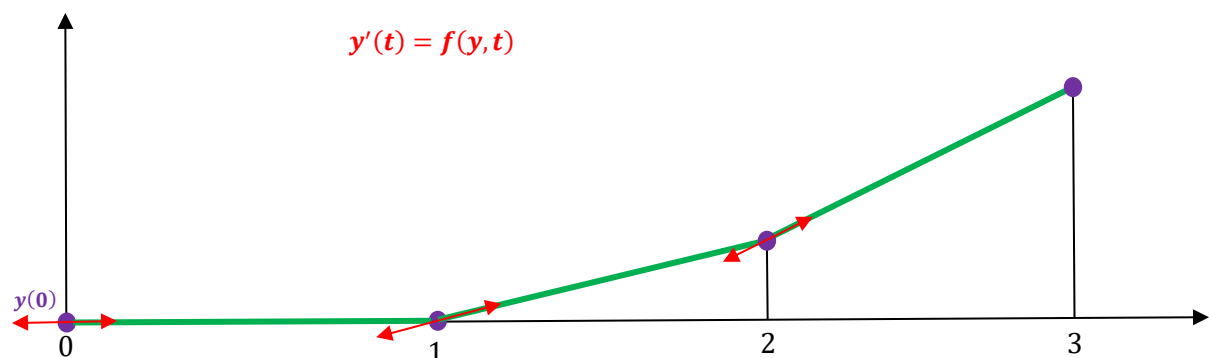
$$y(t + dt) = y(t) + f(y, t)dt$$

Il reste alors numériquement à définir la fonction $f(y, t)$, puis à partir d'une valeur $y_0 = y(t_0)$ connue, de procéder par itérations en calculant $y(t + dt)$ jusqu'à ce que t soit supérieur ou égal à t_1 .

On pourra adapter dt pour arriver exactement sur t_1 .

On pourra par ailleurs ne renvoyer que la solution en t_1 , ou l'ensemble des solutions sur l'intervalle $[t_0, t_1]$.

Illustration :



1.III.2 2° ordre

1.III.2.a Méthode

Soit l'équation :

$$y''(t) = f(y(t), y'(t), t) = f(Y, t) \quad ; \quad Y(t) = \begin{pmatrix} y(t) \\ y'(t) \end{pmatrix}$$

Nous pourrions résoudre cette équation comme vu précédemment en introduisant un développement de Taylor à l'ordre 2 pour $y''(t)$. Voici comment faire autrement.

Soit le vecteur $Y(t) = \begin{pmatrix} y(t) \\ y'(t) \end{pmatrix}$.

On aura alors :

$$Y'(t) = \begin{pmatrix} y'(t) \\ y''(t) \end{pmatrix} = \begin{pmatrix} y'(t) \\ f(Y, t) \end{pmatrix} = F(Y, t)$$

On discrétise $Y'(t)$ à l'ordre 1 :

$$Y'(t) = \frac{Y(t + dt) - Y(t)}{dt}$$

On a alors :

$$Y(t + dt) = Y(t) + F(Y, t)dt$$

avec la fonction vectorielle $F: (Y, t) \mapsto (y', f(Y, t))$

$$\text{En effet : } \begin{pmatrix} y(t + dt) \\ y'(t + dt) \end{pmatrix} = \begin{pmatrix} y(t) \\ y'(t) \end{pmatrix} + \begin{pmatrix} y'(t)dt \\ y''(t)dt \end{pmatrix} \quad ; \quad \begin{pmatrix} y'(t)dt \\ y''(t)dt \end{pmatrix} = \begin{pmatrix} y'(t) \\ y''(t) \end{pmatrix} * dt = F(Y, t) * dt$$

On est alors ramené à la résolution d'une équation différentielle du 1° ordre (méthode détaillée au paragraphe précédent).

$$Y_0 = Y(t_0) = \begin{pmatrix} y(t_0) \\ y'(t_0) \end{pmatrix}$$

La solution $Y(t)$ obtenue contiendra les valeurs approchées de $y(t)$ mais aussi de $y'(t)$.

1.III.2.b Remarques

1.III.2.b.i Concernant les array

Il est possible de tout programmer avec des listes, en particulier si vous utilisez odeint. Avec une fonction Euler explicite, cela dépendra de votre programmation, mais les array peuvent présenter l'avantage de permettre de calculer directement $y = y + F(y, t) * dt$

Avec des listes, on fera la mise à jour ainsi :

- $y = y + y_p * dt$ si une équation d'ordre 1
- $y = [y[i] + y_p[i] * dt \text{ for } i \text{ in range(len(y))}]$ sinon

Quelques remarques :

- Par chance, la somme d'une liste et d'un array donne un array. Si Y_0 est une liste, et que F renvoie bien un array, tout se passera bien, l'inverse est faux car le produit $F(Y, t)dt$ ne passera pas (si $dt = 1$, cas particulier qui fonctionne, et si dt est un entier, il y aura un problème !). Idéalement, bien faire deux fois des array
- ATTENTION : Nous l'avons déjà dit, mais un rappel est nécessaire je pense. Ajouter un array V dans une liste L puis modifier V modifie V dans L aussi ! Dès que vous manipulerez des array en Euler, si vous écrivez « $Y += Y_p * dt$ » au lieu de « $Y = Y + Y_p * dt$ », vous risquez d'avoir à la fin n fois le même Y dans L . Illustration de ce problème :

<pre>import numpy as np V = np.array([1,1]) L = [] L.append(V) print(L) V += 1 L.append(V) print(L) V = V + 1 L.append(V) print(L)</pre>	<pre>[array([1, 1])] [array([2, 2]), array([2, 2])] [array([2, 2]), array([2, 2]), array([3, 3])]</pre>
--	---

- Si vous créez un array d'entiers et qu'en plus, vous utilisez += pour ajouter un flottant, vous obtiendrez une erreur de type, la somme d'un int32 et d'un float64 ne fonctionnant pas...

<pre>a = 10 b = 10 V = np.array([a,b]) V += 0.1</pre>	<pre>Cannot cast ufunc 'add' output from dtype('float64') to dtype('int32') with casting rule 'same_kind'</pre>
---	---

Solutions : $V = V + 0.1$ ou définir a et b en float en ajoutant « .0 », ex $a = 10.0$

1.III.2.b.ii Conditions initiales atypiques

Lorsque les deux conditions initiales sont en position $y(t_0)$ et $y(t_f)$, une résolution itérative partant de $y(t_0)$ et d'une valeur arbitraire $y'(t_0)$ qui sera incrémentée peut permettre de trouver la solution qui respecte $y(t_f)$... (cf équation de chaîne).

1.III.2.c Exemples de mise en équation

1.III.2.c.i Exemple 1 : Masse en chute libre

Soit une masse soumise à la gravité dont on veut déterminer les positions successives en chute libre sous l'action de la gravité et d'une force de frottement fluide. L'application du PFD donne :

$$mg - \mu v = ma$$

$$ma + \mu v = mg$$

$$m \frac{d^2 z(t)}{dt^2} + \mu \frac{dz(t)}{dt} = mg$$

$$m\ddot{z}(t) + \mu\dot{z}(t) = mg$$

$$\ddot{z}(t) = g - \frac{\mu}{m}\dot{z}(t)$$

On note donc $Z(t) = \begin{pmatrix} z(t) \\ \dot{z}(t) \end{pmatrix}$ et on a $Z'(t) = \begin{pmatrix} \dot{z}(t) \\ \ddot{z}(t) \end{pmatrix} = \begin{pmatrix} \dot{z}(t) \\ f(Z, t) \end{pmatrix}$ avec $f(Z, t) = g - \frac{\mu}{m}\dot{z}(t)$

On définit alors $F: (Z, t) \mapsto (\dot{z}, f(Z, t))$.

Et on trouvera le nouveau Z en utilisant : $Z(t + dt) = Z(t) + F(Z, t)dt$

Après résolution, $z(t)$ sera la 1ère coordonnée de $Z(t)$

1.III.2.c.ii Exemple 2 : Pendule

Soit un pendule assimilé à une masse ponctuelle m en B , en rotation avec le bâti en A , en oscillations libres à partir d'une position initiale θ_0 . On suppose qu'il n'est soumis à aucun frottement. L'application du TMD donne :

$$-mgl \sin(\theta(t)) = ml^2 \ddot{\theta}(t)$$

$$\ddot{\theta}(t) + \frac{g}{l} \sin(\theta(t)) = 0$$

$$\ddot{\theta}(t) = -\frac{g}{l} \sin(\theta(t))$$

On note donc $\theta(t) = \begin{pmatrix} \theta(t) \\ \dot{\theta}(t) \end{pmatrix}$ et on a $\dot{\theta}(t) = \begin{pmatrix} \dot{\theta}(t) \\ \ddot{\theta}(t) \end{pmatrix} = \begin{pmatrix} \dot{\theta}(t) \\ f(\theta, t) \end{pmatrix}$ avec $f(\theta, t) = -\frac{g}{l} \sin(\theta(t))$

On définit alors $F: (\theta, t) \mapsto (\dot{\theta}, f(\theta, t))$

Et on trouvera le nouveau θ en utilisant : $\theta(t + dt) = \theta(t) + F(\theta, t)dt$

Après résolution, $\theta(t)$ sera la 1ère coordonnée de $\theta(t)$

1.IV. Systèmes d'équations différentielles du premier et du second ordre

Lorsque l'on a compris les principes précédents, il devient assez simple de résoudre des systèmes d'équations différentielles en « vectorisant » le problème.

Ci-contre, un exemple (Mines Ponts MP 2016) de modèle d'évolution d'une population face à une épidémie avec les nombres de personnes S (saine), I (infectée), R (remise) et D (décédée).

$$\begin{cases} \frac{dS(t)}{dt} = -rS(t)I(t) \\ \frac{dI(t)}{dt} = rS(t)I(t) - (a+b)I(t) \\ \frac{dR(t)}{dt} = aI(t) \\ \frac{dD(t)}{dt} = bI(t) \end{cases}$$

1.IV.1 1° ordre

Soit le système :

$$\begin{cases} \frac{df_1(t)}{dt} = g_1(f_1(t), f_2(t), \dots, f_n(t), t) \\ \frac{df_2(t)}{dt} = g_2(f_1(t), f_2(t), \dots, f_n(t), t) \\ \vdots \\ \frac{df_n(t)}{dt} = g_n(f_1(t), f_2(t), \dots, f_n(t), t) \end{cases}$$

Il suffit de définir le vecteur : $Y(t) = \begin{pmatrix} f_1(t) \\ f_2(t) \\ \vdots \\ f_n(t) \end{pmatrix}$

On a alors :

$$\begin{cases} \frac{df_1(t)}{dt} = g_1(Y, t) \\ \frac{df_2(t)}{dt} = g_2(Y, t) \\ \vdots \\ \frac{df_n(t)}{dt} = g_n(Y, t) \end{cases}$$

On définit alors la fonction :

$$F_p: (Y, t) \mapsto \begin{pmatrix} f'_1(t) \\ f'_2(t) \\ \vdots \\ f'_n(t) \end{pmatrix} = \begin{pmatrix} g_1(Y, t) \\ g_2(Y, t) \\ \vdots \\ g_n(Y, t) \end{pmatrix}$$

On a alors : $Y'(t) = F_p(Y, t)$

Puis de résoudre comme pour une équation différentielle du 1° ordre :

$$Y(t + dt) = Y(t) + F_p(Y, t)dt$$

1.IV.2 2° ordre

Soit le système :

$$\begin{cases} \frac{d^2 f_1(t)}{dt^2} = g_1(f_1, f_1', f_2, f_2', \dots, f_n, f_n', t) \\ \frac{d^2 f_2(t)}{dt^2} = g_2(f_1, f_1', f_2, f_2', \dots, f_n, f_n', t) \\ \vdots \\ \frac{d^2 f_n(t)}{dt^2} = g_n(f_1, f_1', f_2, f_2', \dots, f_n, f_n', t) \end{cases}$$

Il suffit de définir le vecteur inconnu :

$$Y(t) = \begin{pmatrix} f_1(t) \\ f_1'(t) \\ f_2(t) \\ f_2'(t) \\ \vdots \\ f_n(t) \\ f_n'(t) \end{pmatrix}$$

On a donc :

$$\begin{cases} \frac{d^2 f_1(t)}{dt^2} = g_1(Y, t) \\ \frac{d^2 f_2(t)}{dt^2} = g_2(Y, t) \\ \vdots \\ \frac{d^2 f_n(t)}{dt^2} = g_n(Y, t) \end{cases}$$

On définit alors la fonction :

$$F_p(Y, t) \mapsto \begin{pmatrix} f_1'(t) \\ f_1''(t) \\ f_2'(t) \\ f_2''(t) \\ \vdots \\ f_n'(t) \\ f_n''(t) \end{pmatrix} = \begin{pmatrix} f_1'(t) \\ g_1(Y, t) \\ f_2'(t) \\ g_2(Y, t) \\ \vdots \\ f_n'(t) \\ g_n(Y, t) \end{pmatrix}$$

On a alors :

$$Y'(t) = F_p(Y, t)$$

Puis de résoudre comme pour une équation différentielle du 1° ordre :

$$Y(t + dt) = Y(t) + F_p(Y, t)dt$$

1.V. Modélisation par équations aux différences

A la différence des méthodes vues au paragraphe précédent pour la résolution d'équations simples de la forme $y'(t) = f(y, t)$ ou $y''(t) = f((y, y'), t)$, on va pouvoir ici résoudre tous types d'équations différentielles ☺. Au concours, c'est plus généralement ce type de fonctions simples qui sont proposées.

La méthode que nous allons aborder ici est toutefois utilisable aussi pour les fonctions ci-dessus !

1.V.1 Contexte

La méthode des équations aux différences est une méthode plus puissante que ce que nous venons de voir précédemment. Elle permet certes de traiter les équations vues précédemment du type $y^{(i)}(t) = f(y^{(i-1)}(t), y^{(i-2)}(t), \dots, y(t), t)$, mais elle va nous permettre de résoudre des équations de plusieurs variables comme l'équation de la chaleur 1D ci-dessous, l'équation d'onde etc., non solvables avec la méthode classique :

$$\frac{\partial T(t, x)}{\partial t} = D_{th} \frac{\partial^2 (T(t, x))}{\partial x^2}$$

1.V.2 Méthode de résolution

Supposons que e est une entrée connue et s une sortie recherchée.

Pour déterminer, à tout temps $t = t_0 + kT$, on exprime chacune des dérivées avec les méthodes vues précédemment (Euler, Taylor, différences finies) afin d'obtenir une relation entre divers états de l'entrée et de la sortie et on exprime ensuite à l'aide de cette équation la valeur recherchée en fonction des valeurs connues de l'entrée et des valeurs précédentes de la sortie.

Autrement dit, une équation aux différences est de la forme :

$$s(t + dt) = f(t, s(t), s(t - dt), s(t - 2dt) \dots, e(t), e(t - dt), e(t - 2dt) \dots)$$

On veillera à traiter correctement les conditions initiales.

1.V.3 Exemple 1 : Chute libre

1.V.3.a Equation récurrente

Reprenons l'exemple de la masse en chute libre étudiée avec la méthode précédente, dont l'équation était :

$$mg - \mu v = ma$$

$$ma + \mu v = mg$$

$$m \frac{d^2 z(t)}{dt^2} + \mu \frac{dz(t)}{dt} = mg$$

On choisit d'exprimer les dérivées première et seconde de z à l'aide d'un schéma Euler Explicite pour la dérivée première et Euler centré pour la dérivée seconde :

$$\frac{dz(t)}{dt} = \frac{z(t+T) - z(t)}{T}$$

$$\frac{d^2 z(t)}{dt^2} = \frac{z(t+T) - 2z(t) + z(t-T)}{T^2}$$

On obtient donc l'équation aux différences suivante :

$$m \frac{z(t+T) - 2z(t) + z(t-T)}{T^2} + \mu \frac{z(t+T) - z(t)}{T} = mg$$

On obtient alors la position à tout instant connaissant les positions antérieures :

$$\frac{mz(t+T)}{T^2} - \frac{2mz(t)}{T^2} + \frac{mz(t-T)}{T^2} + \frac{\mu z(t+T)}{T} - \frac{\mu z(t)}{T} = mg$$

$$\left(\frac{m}{T^2} + \frac{\mu}{T}\right) z(t+T) - \frac{2m + \mu T}{T^2} z(t) + \frac{m}{T^2} z(t-T) = mg$$

$$\frac{m + \mu T}{T^2} z(t+T) = \frac{2m + \mu T}{T^2} z(t) - \frac{m}{T^2} z(t-T) + mg$$

$$z(t+T) = \frac{2m + \mu T}{m + \mu T} z(t) - \frac{m}{m + \mu T} z(t-T) + \frac{T^2}{m + \mu T} mg$$

1.V.3.b Prise en compte des conditions initiales

Attention, à l'instant initial $t = t_0$, on a $z(t_0)$.

Pour calculer la valeur suivante, il faut $z(t_0)$ et $z(t_0 - T)$ qui n'existe pas !

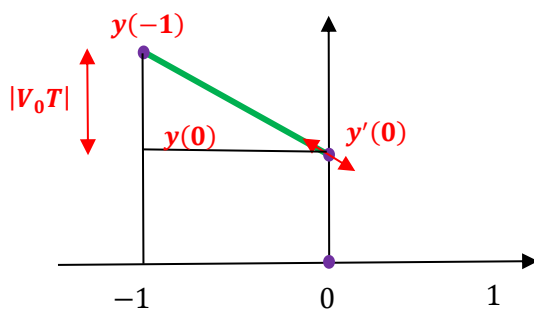
Deux solutions (qui en réalité sont les mêmes, la première étant un cas particulier):

- La vitesse initiale est nulle, alors on sait qu'avant t_0 , $z(t) = z(t_0)$ et on peut donc affecter la valeur en $t_0 - T$ égale à la valeur en t_0
- La vitesse initiale est non nulle, on utilise la formule d'Euler implicite :

$$\frac{dz(t_0)}{dt} = V_0 = \frac{z(t_0) - z(t_0 - T)}{T}$$

Ce qui permet d'imposer (en traitant le cas particulier du premier pas de temps sous Python):

$$z(t_0 - T) = z(t_0) - V_0 T$$



1.V.3.c Remarque : Regroupement de termes dépendant de z dans le second membre

Il serait possible de résoudre l'équation en intégrant la force de frottement fluide en second membre :

$$ma = mg - \mu v$$

On pose

$$A = mg - \mu v \Leftrightarrow a = \frac{A}{m}$$

A chaque instant, on calcule donc la valeur de $mg - \mu v$ que l'on appellera A . Pour calculer v à chaque étape, on utilise Euler implicite (utilisation de valeurs connues) à l'ordre 1 :

$$v = \frac{dz(t)}{dt} = \frac{z(t) - z(t - T)}{T}$$

On peut donc trouver une nouvelle équation aux différences :

$$\begin{aligned} \frac{z(t+T) - 2z(t) + z(t-T)}{T^2} &= \frac{A}{m} \Leftrightarrow z(t+T) - 2z(t) + z(t-T) = \frac{AT^2}{m} \\ \Leftrightarrow z(t+T) &= \frac{AT^2}{m} + 2z(t) - z(t-T) \end{aligned}$$

Cela peut être très utile en présence de termes non linéaires, par exemple une force en carré de la vitesse (frottements fluides réalistes...).

1.V.4 Exemple 2 : Pendule

1.V.4.a Equation récurrente

Reprenons l'exemple du pendule étudié avec la méthode précédente, dont l'équation était :

$$\frac{d^2\theta(t)}{dt^2} + \frac{g}{l} \sin \theta = 0$$

On choisit d'exprimer la dérivée seconde de θ à l'aide d'un schéma Euler centré :

$$\frac{d^2\theta(t)}{dt^2} = \frac{\theta(t+T) - 2\theta(t) + \theta(t-T)}{T^2}$$

On obtient donc l'équation aux différences suivante :

$$\frac{\theta(t+T) - 2\theta(t) + \theta(t-T)}{T^2} + \frac{g}{l} \sin \theta(t) = 0$$

On obtient alors l'angle à tout instant connaissant les angles antérieurs :

$$\theta(t+T) = 2\theta(t) - \theta(t-T) - \frac{gT^2}{l} \sin \theta(t)$$

1.V.4.b Prise en compte des conditions initiales

Attention, à l'instant initial $t = t_0$, on a $\theta(t_0)$.

Pour calculer la valeur suivante, il faut $\theta(t_0)$ et $\theta(t_0 - T)$. Deux solutions :

- La vitesse initiale est nulle, alors on sait qu'avant t_0 , $\theta(t) = \theta(t_0)$ et on peut donc affecter la valeur en $t_0 - T$ égale à la valeur en t_0
- La vitesse initiale est non nulle, on a alors :

$$\frac{d\theta(t_0)}{dt} = V_0 = \frac{\theta(t+T) - \theta(t)}{T}$$

On suppose qu'on a aussi :

$$\frac{d\theta(t_0)}{dt} = V_0 = \frac{\theta(t) - \theta(t-T)}{T}$$

On impose alors :

$$\theta(t-T) = \theta(t_0) - V_0 T$$

1.VI. Résolution d'équations et précision

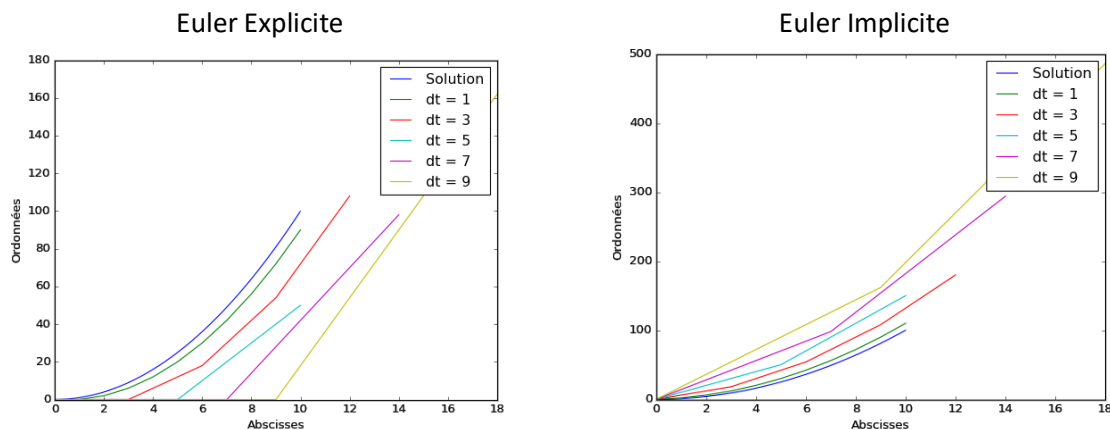
1.VI.1 Précision sur la solution

Comme nous l'avons dit, nos approximations de dérivées ne seraient exactes que si le pas de temps était infiniment nul. Ce n'est évidemment pas le cas, et plus le pas de temps est petit, plus les temps de calcul deviennent élevés...

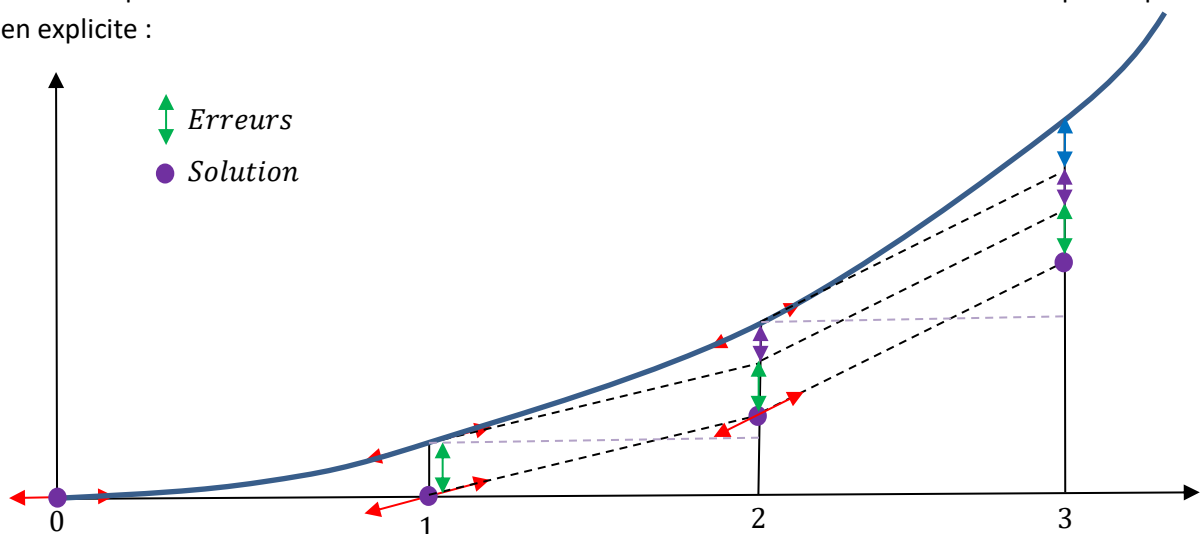
Il faut donc être conscient que la méthode de résolution par discrétisations d'Euler induit des erreurs ! Et celles-ci se cumulent.

Soit la fonction $f(t) = t^2$.

Voici le résultat de la résolution de l'équation $f'(t) = 2t$ par la méthode d'Euler explicite ($y(t + dt) = y(t) + y'(t)dt$) et implicite ($y(t + dt) = y(t) + y'(t + dt)dt$ possible car $y'(t + dt)$ calculable au temps $t + dt$, la formule étant connue) décrite au paragraphe précédent entre 0 et 10 pour des pas de temps différents :



On remarque bien que plus le pas de temps est petit, plus la courbe tend vers la solution réelle. On peut noter que pour le pas de temps de 9, la première valeur calculée est très fautive. La suivante sera calculée à partir de cette fautive valeur... L'erreur se cumule ! Voici une illustration de ce qu'il se passe en explicite :



La méthode d'Euler est d'autant plus précise qu'il y a de points dans les zones à forte variation. Ainsi, si la solution est une droite, la méthode d'Euler ne fait aucune erreur...

1.VI.2 Précision sur le temps

Ceci est une simple remarque mais cela a son importance dans les simulations sur un grand nombre de pas de temps.

Lorsque le pas de temps n'est pas un nombre décimal fini, il existe une différence entre le calcul du temps avec les deux méthodes suivantes :

- Calcul absolu : $t = t_0 + i * dt$
- Calcul relatif : $t = t + dt$

Illustration :

```
dt = 1/3
T = 1e8
t0 = 0

Liste_T1 = []
Liste_T2 = []

i = 0
t1 = t0
Pas = int(T/dt)
while i < Pas:
    i += 1
    t1 = t1 + dt
    t2 = t0 + i * dt
    Liste_T1.append(t1)
    Liste_T2.append(t2)

Tf1 = Liste_T1[-1]
Tf2 = Liste_T2[-1]
DTf = Tf2 - Tf1
print(Tf1)
print(Tf2)
print(DTf)
```

```
>>> (executing file "<tmp 1>")
99999999.70988387
100000000.0
0.2901161313056946
```

Lorsque les temps de simulation deviennent grands, le cumul des erreurs d'arrondi en calcul relatif peut conduire à des imprécisions non négligeables.

Lequel choisir ? Cela dépend du contexte... Il faut surtout être conscient du problème !

1.VII. Fonction de résolution intégrée à Python

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 Y0 = 0</pre>
1° ordre - 2 équations	
$\begin{cases} \frac{dH(t)}{dt} = aH(t) - bA(t)H(t) \\ \frac{dA(t)}{dt} = -cA(t) + dA(t)H(t) \end{cases}$ $Y(t) = \begin{pmatrix} H(t) \\ A(t) \end{pmatrix}$	<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 H0,A0 = 10.0,1.0 Y0 = [H0,A0]</pre>
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 from math import pi,sin y0,yp0 = 45*pi/180,0 Y0 = [y0,yp0]</pre>
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,yp,ypp] return dY x0,y0 = 0,0 xp0,yp0 = 10,10 Y0 = [x0,xp0,y0,yp0]</pre>
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>	