Equations différentielles : simulation par la méthode d'Euler

1 Introduction

1.1 Prédire la météo ?

L'objectif d'une prévision météo est de déterminer certaines grandeurs (température, vent, etc.) pendant les trois prochains jours. Pour cette prévision, il faut :

- déterminer certaines quantités $y(t, x)$ (température, pression, vent, humidité)

en fonction de :

- certains paramètres : le temps t et / ou l'espace x ;
- de y_0 la valeur des quantités à l'instant initial ;
- d'entrées ou excitations connues $u(t)$ (l'ensoleillement).

La Physique indique le comportement sous la forme d'équations différentielles (mécanique des fluides, thermodynamique). Une fois que le problème modélisé, la simulation est possible

2 Equation différentielle d'ordre 1

2.1 Formalisation

Soient :

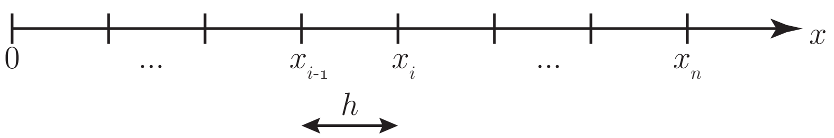
- $y(x)$ une quantité à déterminer ;
- en fonction d'un paramètre x ;
- sous l'effet d'une excitation connue $u(x)$;
- avec la condition initiale $y_0 = y(0)$ connue.

Le problème peut alors s'écrire (conditions d'existance de Cauchy-Lipschitz non étudiées) :

$$\frac{dy}{dx} = f\left(x, y\left(x\right), u\left(x\right)\right)$$

2.2 Echantillonnage

Ne pouvant traiter des grandeurs continues, un programme va traiter des grandeurs **discrétisées**. Dans notre cas, l'axe des abscisses est découpés en intervalles réguliers, dits **de pas** $h = \Delta x$. Seules les valeurs particulières notées x_i sont traitées.



Pour la suite, on note :

$$\begin{array}{rclcrcl} y(x_i) & = & y_i & ; & f(x_i) & = & f_i \\ y(x_{i+1}) & = & y_{i+1} & ; & f(x_{i+1}) & = & f_{i+1} \\ y(x_{i-1}) & = & y_{i-1} & ; & f(x_{i-1}) & = & f_{i-1} \end{array}$$

2.3 Résolution numérique : la simulation

Dans le cadre de ce cours, seule l'approximation d'Euler explicite sera utilisée.

$$\frac{dy}{dx} = f\left(x, y\left(x\right), u\left(x\right)\right)$$

$$\begin{aligned} y &= \int_0^x f\left(\xi, y(\xi), u(\xi)\right) d\xi \\ y_i &= y_{i-1} + h \cdot f_{i-1} \end{aligned}$$

$$\begin{aligned} \frac{dy}{dx} &\approx \frac{\Delta y}{\Delta x} = \frac{y_{i+1} - y_i}{h} = f_i \\ y_{i+1} &= y_i + h \cdot f_i \end{aligned}$$

Equation de récurrence (avec y_0 connu) :

$$y_i = y_{i-1} + h \cdot f_{i-1}$$

On obtient :
- X : liste des valeurs du paramètre x_i ;
- Y : liste des valeurs y_i calculés aux instants d'échantillonnage x_i.

2.4 Mise en œuvre

L'algorithme se résume en 5 étapes.

1. préparer la fonction $f(x, y)$
2. paramétrer h le pas et la durée (nombre de points)
3. initialiser X (à sa valeur initiale) et Y à sa condition initiale y_0
4. calculer y_{i+1} en fonction de y_i et f_i.
5. incrémenter i
6. revenir en 4.

```python
def f(x,y):
    return(#une fonction de x et y)

h=0.1
nb_point=101
X=[0]
Y=[3]
for i in range(1,nb_point):
    Y=Y+[Y[i-1]+f(X[i-1],Y[i-1])*h]
    X=X+[X[i-1]+h]
```

3 Performances

3.1 Précision numérique

Chaque itération est le lieu d'une approximation, qui introduit une erreur numérique. Cette erreur e_l, appelée *local truncation error* **LTE** en anglais, se cumule sur l'ensemble du calcul. L'erreur cumulée e_g est appelée *global truncation error* **GTE**.

Dans le cas de l'approximation par Euler explicite, on démontre pour h suffisamment petit :

$$\begin{aligned} \textbf{LTE} : e_l &= \mathcal{O}(h^2) \\ \textbf{GTE} : e_g &= \mathcal{O}(h) \end{aligned}$$

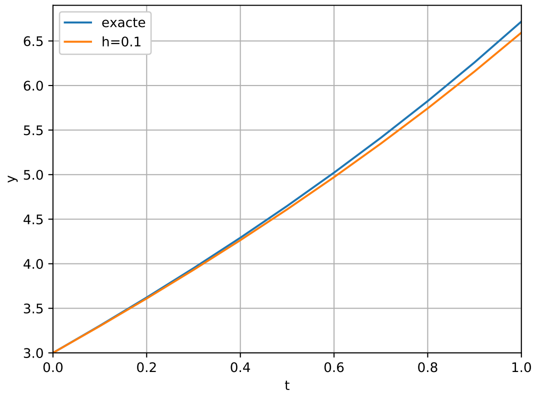
Exemple :

Equation à résoudre :

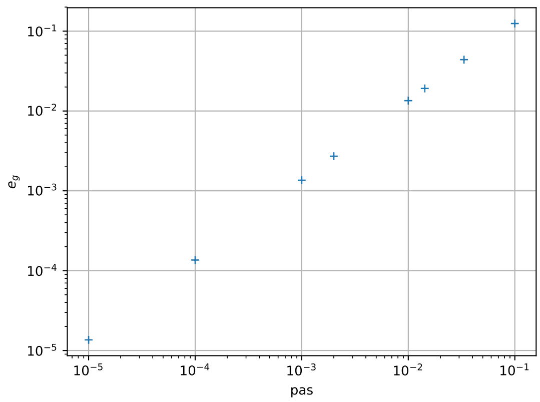
$$\begin{cases} \dot{y}(t) &= y(t) - 2t \\ y(0) &= 3 \end{cases}$$

Solution exacte

$$y(t) = 2 + 2t + e^t$$



h	nb points	y(1)
		6.7182818285
0.1	11	6.5937424601
0.03	31	6.6743187759
0.01	101	6.7048138294
2e-3	501	6.7155685207
1e-3	1001	6.7169239322
1e-4	10001	6.7181459268
1e-5	100001	6.7182682372



La précision d'une méthode numérique d'intégration est qualifiée par son ordre n : $e_g = \mathcal{O}(h^n)$.

méthode	ordre	e_g
Euler avant ou arrière	1	$\mathcal{O}(h)$
Trapèze	2	$\mathcal{O}(h^2)$
Range-Kutta	4	$\mathcal{O}(h^4)$

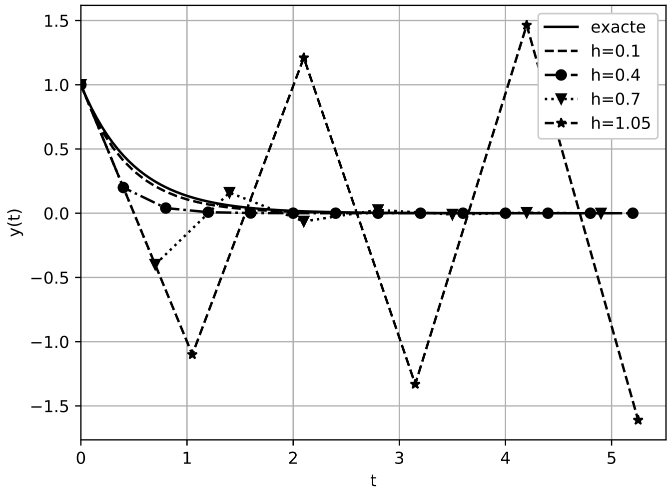
3.2 Stabilité numérique

Equations à résoudre :

$$\begin{cases} \dot{y} &= -2y \\ y(0) &= 1 \end{cases}$$

Solution exacte

$$y(t) = e^{-2t}$$



3.3 Compléxité

4 Equation différentielle d'ordre supérieur

4.1 Exemple : circuit RLC

$$LC\frac{d^2u_C}{dt^2} + RC\frac{du_C}{dt} + u_C = u_e$$

$$\begin{aligned} u_e &= u_C + u_L + u_R \\ &= u_C + L\frac{di}{dt} + R \cdot i \\ i &= C\frac{du_C}{dt} \end{aligned}$$

$$\begin{cases} \dot{i} = \frac{1}{L}\left(u_e - u_C + R \cdot i\right) \\ \dot{u_C} = \frac{1}{C} i \end{cases}$$

4.2 Formalisation

Soit l'équation différentielle d'ordre 2 suivante :

$$\ddot{y}(t) + \dot{y}(t) + y(t) = 0$$

On introduit le changement de variable suivant :

$$\begin{cases} y_0(t) = y(t) \\ y_1(t) = \dfrac{dy}{dt} \end{cases}$$

Alors :

$$\begin{aligned} \dot{y_0} &= y_1 \\ \dot{y_1} + y_1 + y_0 &= 0 \end{aligned}$$

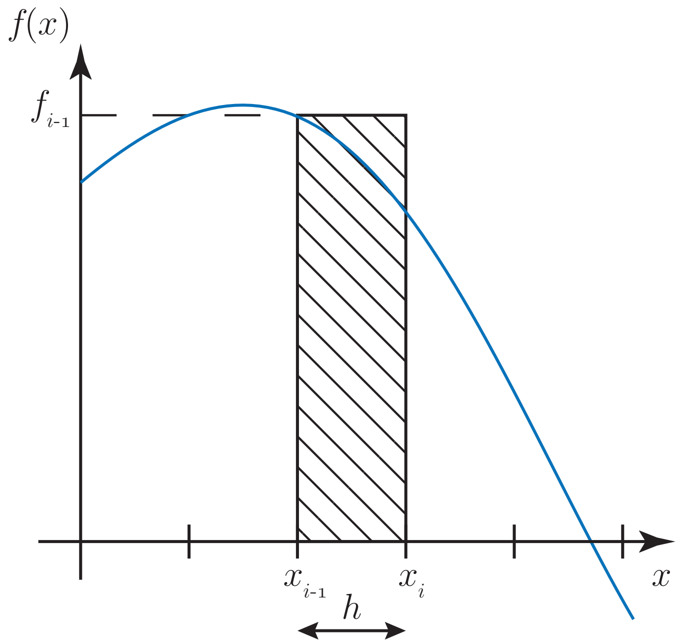
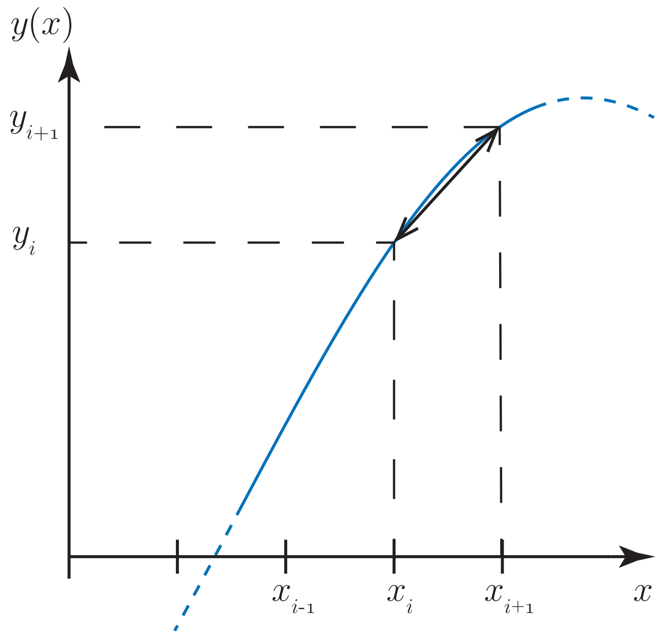
$$\begin{cases} \dot{y_0} = & y_1 \\ \dot{y_1} = & -y_1 - y_0 \end{cases}$$

$$\begin{cases} \dfrac{y_{0_{i+1}} - y_{0_i}}{\Delta t} = & y_{1_i} \\ \dfrac{y_{1_{i+1}} - y_{1_i}}{\Delta t} = & -y_{1_i} - y_{0_i} \end{cases}$$

$$\begin{cases} y_{0_{i+1}} = & y_{0_i} + \Delta t \cdot y_{1_i} \\ y_{1_{i+1}} = & y_{1_i} + \Delta t \left(-y_{1_i} - y_{0_i}\right) \end{cases}$$

5 Annexe : Approximation d'Euler explicite ou avant

$$\frac{dy}{dx} = f(x)$$

Figure	Approximation
	du calcul d'intégrale $y_i = y_{i-1} + h \cdot f_{i-1}$
	du calcul de la dérivée $\frac{dy}{dx} \approx \frac{\Delta y}{\Delta x} = \frac{y_{i+1} - y_i}{h}$