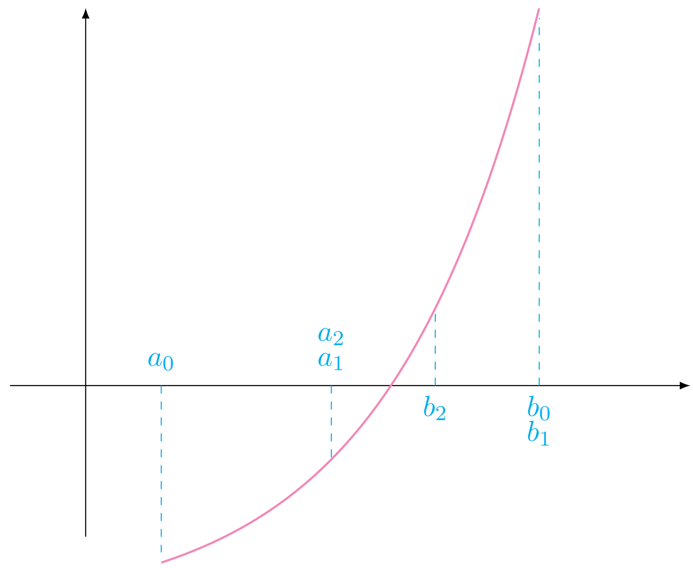
2
Analyse numérique

Sommaire

2.1 Méthode d'approximation des solutions d'une équation $f(x) = 0$

2.1.1 Méthode de la dichotomie

On considère une fonction continue $f : [a, b] \to \mathbb{R}$, vérifiant $f(a) \cdot f(b) \leq 0$, tel que l'équation $f(x) = 0$ admette une unique solution.



On définit deux suites $(a_n)_{n\in\mathbb{N}}$ et $(b_n)_{n\in\mathbb{N}}$ en posant :

$$\begin{cases} a_0 = a & b_0 = b & \\ a_{n+1} = a_n & b_{n+1} = \frac{a_n+b_n}{2} & \text{si } f(a_n) \cdot f\left(\frac{a_n+b_n}{2}\right) \leq 0 \\ a_{n+1} = \frac{a_n+b_n}{2} & b_{n+1} = b_n & \text{si } f(a_n) \cdot f\left(\frac{a_n+b_n}{2}\right) > 0 \end{cases}$$

Alors : ●●

1. Les suites $(a_n)_{n\in\mathbb{N}}$ et $(b_n)_{n\in\mathbb{N}}$ sont adjacentes et convergent vers α.
2. Pour tout $n \in \mathbb{N}$: $\left|\dfrac{a_n + b_n}{2} - \alpha\right| \leq \dfrac{b-a}{2^n}$.

Exercice 2.1. Écrire une fonction `dichotomie(f,a,b,epsilon)` qui retourne une valeur approchée de α à ε près par la méthode de la dichotomie.

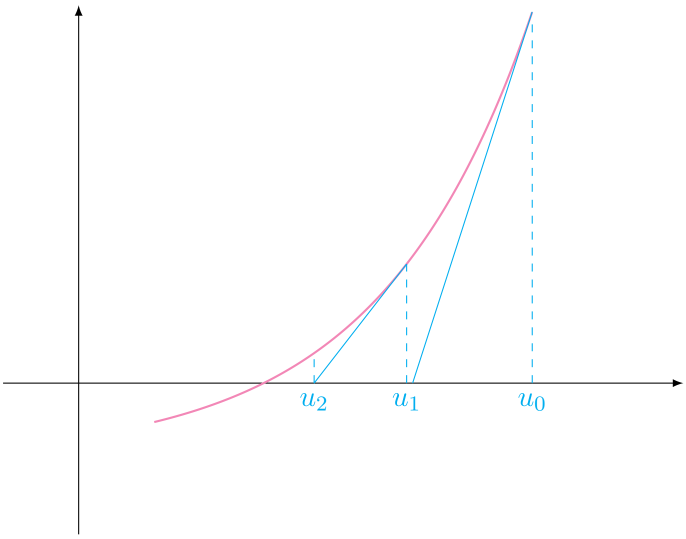
Solution.

2.1.2 La méthode de Newton

On considère une fonction dérivable $f : [a, b] \to \mathbb{R}$, telle que f' ne s'annule pas, vérifiant $f(a) \cdot f(b) \leq 0$, et tel que l'équation $f(x) = 0$ admette une unique solution.

La méthode de Newton :

- On se donne u_0.
- On trace la tangente à la courbe au point d'abscisse u_0, son intersection avec l'axe des abscisses fournit une nouvelle valeur u_1.
- On trace la tangente à la courbe au point d'abscisse u_1, son intersection avec l'axe des abscisses fournit une nouvelle valeur u_2.
- Etc...



Cherchons comment construire la suite $(u_n)_{n\in\mathbb{N}}$.

Supposons que l'on ait construit tous les termes jusqu'à $n \in \mathbb{N}$, alors le terme suivant u_{n+1} est solution de l'équation :

$$f'(u_n)(x - u_n) + f(u_n) = 0.$$

Soit :

$$u_{n+1} = u_n - \frac{f(u_n)}{f'(u_n)}.$$

Posons, $g : x \mapsto x - \dfrac{f(x)}{f'(x)}$, alors la suite $(u_n)_{n\in\mathbb{N}}$ est donnée par :

$$\begin{cases} u_0 \in \mathbb{R} \\ u_{n+1} = g(u_n), \ \forall n \in \mathbb{N} \end{cases}$$

En général, on choisit comme condition d'arrêt une inégalité de la forme : $|u_{n+1} - u_n| \leq \varepsilon$.

La suite peut ne pas être correctement définie dans certains cas, et quand elle l'est sa convergence n'est pas acquise, voici un exemple de conditions suffisantes d'existence et de convergence.

Théorème 2.1.

Soit : $[a,b] \to \mathbb{R}$ une fonction de classe $\mathcal{C}^2$ telle que :

— $f(a)f(b) < 0$.

— $\forall x \in [a,b]$, $f'(x) \neq 0$ (la fonction est strictement monotone).

— $\forall x \in [a,b]$, $f''(x) \neq 0$ (la convexité est constante).

— $u_0 \in [a,b]$, est tel que $f(u_0)f''(u_0) > 0$.

Alors la suite $(u_n)_{n\in\mathbb{N}}$ définie par $u_{n+1} = g(u_n)$ converge vers α l'unique zéro de f sur $[a,b]$.

Si on considère l'erreur $e_n = u_n - \alpha$, alors :

$$\frac{e_{n+1}}{e_n} = \frac{u_{n+1}-\alpha}{u_n-\alpha} \to g'(\alpha) = 0.$$

Ce qui montre que la convergence est assez rapide...
On a mieux, on peut montrer que :

$$e_{n+1} \approx \frac{e_n^2}{2}\frac{f''(\alpha)}{f'(\alpha)}.$$

Donc, on double à peu près le nombre de décimales exactes à chaque itération.

Exercice 2.2. Écrire une fonction `newton(f,fprime,a,b,epsilon)` qui retourne une valeur approchée de α à ε près par la méthode de la Newton.

Solution.

2.2 Résolution approchée d'équation différentielle

Dans ce paragraphe nous nous intéresserons à des équations différentielle d'ordre 1 et 2.
On appelle problème de Cauchy la donnée d'une équation différentielle ordinaire résolue et d'une condition initiale :

$$\left\{\begin{array}{rcl} y'(t) & = & \phi(t,y(t)) \\ y(t_0) & = & y_0 \end{array}\right. \quad \text{ou} \quad \left\{\begin{array}{l} y''(t) = \phi(t,y(t),y'(t)) \\ y(t_0) = y_0,\ y'(t_0) = y_1 \end{array}\right.$$

Où ϕ est une fonction de $[a,b]\times\mathbb{R}$ dans $\mathbb{R}$.

Donnons des exemples d'équations (E_1) : $y'(t) = \phi(t,y(t))$ et (E_2) : $y''(t) = \phi(t,y(t),y'(t))$:

— $\phi(t, y(t)) = f(t)$, (E_1) est équivalente à la recherche d'une primitive de f.
— $\phi(t, y(t)) = a(t) \cdot y(t)$, (E_1) est une équation différentielle linéaire homogène d'ordre 1.
— $\phi(t, y(t)) = a(t) \cdot y(t) + f(t)$, (E_1) est une équation différentielle linéaire d'ordre 1.
— $\phi(t, y(t)) = \sin^2(t + y(t))$, (E_1) est une équation différentielle d'ordre 1 non linéaire.
— $\phi(t, y(t), y'(t)) = a(t) \cdot y'(t) + b(t) \cdot y(t)$, (E_2) est une équation différentielle linéaire homogène d'ordre 2.
— $\phi(t, y(t), y'(t)) = a(t) \cdot y'(t) + b(t) \cdot y(t) + f(t)$, (E_2) est une équation différentielle linéaire d'ordre 2.
— $\phi(t, y(t), y'(t)) = t \cdot y'(t) \cdot y(t)$, (E_2) est une équation différentielle non linéaire d'ordre 2.

Exercice 2.3. Déterminer la fonction ϕ dans chacun des exemples d'équations différentielles suivants :

1. $(1 + t^2)y'(t) + 2ty(t) = 0$;
2. $t^2 y'(t) + \sin(t)y(t) = \cos(t)$;
3. $\dfrac{1}{1 + t^2} y'(t) - 2ty(t)^2 = \tan(t)$;
4. $2y''(t) - 4y'(t) + 6y(t) = 10e^t$;
5. $y''(t) + 2y'(t)y(t) = t^2$;

Solution.

Considérons y la solution exacte de notre problème de Cauchy d'ordre 1 ou 2, et $t > t_0$. Pour approcher y on commence par discrétiser l'intervalle $[t_0, t]$ en un nombre fini de points $t_0 < t_1 < \ldots < t_n = t$, puis on cherche successivement une valeur approchée y_k de $y(t_k)$. On pose toujours $y_0 = y(t_0)$.

Le nombre $h_k = t_k - t_{k-1}$ s'appelle le pas de la discrétisation au temps t_k, on noter $h = {}_{\max} 1 \leq k \leq n h_k$. Si tous les h_k sont égaux on parle d'une méthode à pas constants, dans ce cas on a $t_k = t_0 + k\dfrac{t_n - t_0}{n}$, sinon on parle d'une méthode à pas variable.

Lorsque y_{k+1} est calculé à partir de y_k seulement, on parle de méthode à un pas, si le calcul fait intervenir y_k et y_{k-1} on parle de méthode à deux pas.

Méthode d'Euler explicite à un pas

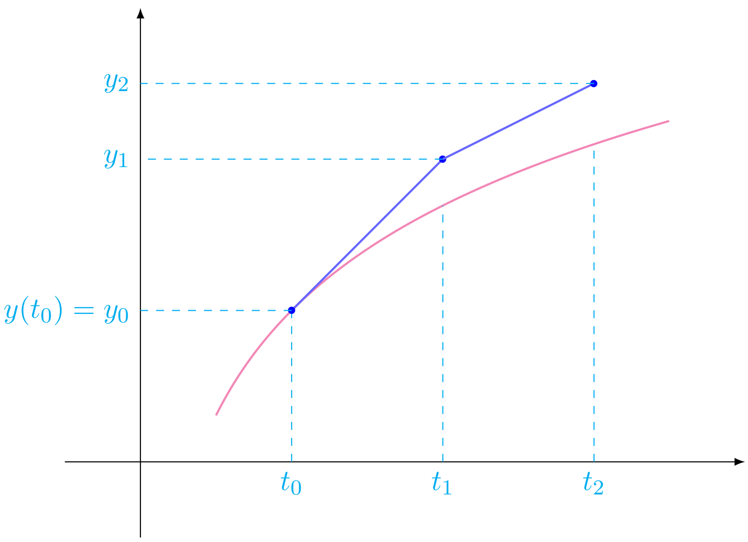
On considère le problème de Cauchy $y' = \phi(t, y(t))$, où ϕ est définie sur $[a, b] \times \mathbb{R}$ et la condition initiale $(t_0 = a, y(t_0))$.

Soit $n \in \mathbb{N}^*$, la méthode d'Euler explicite, ou schéma d'Euler, consiste :

1. à se donner une subdvision $a = t_0 < t_1 < \ldots < t_n = b$ de $[a, b]$,
2. à construire de proche en proche une suite de $n+1$ points $(y_k)_{0\leq k\leq n}$, tels que y_k soit une approximation de $y(t_k)$, obtenue en approchant $y(t_{k+1})$ par la tangente à y au point d'abscisse t_k :

$$\begin{aligned} \frac{y(t_{k+1}) - y(t_k)}{t_{k+1} - t_k} &\approx y'(t_k) = \phi(t_k, y(t_k)) \\ y_{k+1} &= y_k + (t_{k+1} - t_k)\phi(t_k, y_k) \end{aligned}$$

$$\boxed{y_{k+1} = y_k + h_k \times \phi(t_k, y_k) \text{ pour } 0 \leq k \leq n-1.}$$



Sous de bonnes conditions la méthode converge en $O(|h|)$.

Exercice 2.4. Écrire une fonction `euler_explicite(phi,a,b,y0,n)` qui prend comme arguments la fonction $\phi : [a, b] \times \mathbb{R} \to \mathbb{R}$, les bornes a et b de l'intervalle, la condition initiale y_0 et n le nombre d'intervalles de la subdivision, et qui retourne le couple (t, y) du tableau de valeurs des t_k et le tableau des valeurs y_k.

Solution.

La méthode d'Euler implicite à un pas

On considère le problème de Cauchy $y' = \phi(t, y(t))$, où ϕ est définie sur $[a, b] \times \mathbb{R}$ et la condition initiale $(t_0 = a, y(t_0))$.

Soit $n \in \mathbb{N}^*$, la méthode d'Euler implicite, ou schéma d'Euler d'implicite, consiste :

1. à se donner une subdvision $a = t_0 < t_1 < \ldots < t_n = b$ de $[a, b]$,
2. à construire de proche en proche une suite de $n+1$ points $(y_k)_{0\leq k\leq n}$, tels que y_k soit une approximation de $y(t_k)$, obtenue en approchant $y'(t_{k+1})$ par le taux d'accroissement entre t_k et t_{k+1} :

$$\frac{y(t_{k+1}) - y(t_k)}{t_{k+1} - t_k} \quad \approx \quad y'(t_{k+1}) = \phi(t_{k+1}, y(t_{k+1}))$$

Il faut donc résoudre numériquement l'équation : $y_{k+1} = y_k + (t_{k+1} - t_k)\phi(t_{k+1}, y_{k+1})$ d'inconnue y_{k+1}. On a alors :

$$\boxed{y_{k+1} = y_k + h_k \times \phi(t_{k+1}, y_{k+1}) \text{ pour } 0 \leq k \leq n-1.}$$

Sous de bonnes hypothèses encore l'erreur est en $O(|h|)$.

Méthode de prédiction-correction

Ici on construit comme dans la méthode d'Euler une suite de $n+1$ points $(y_k)_{0\leq k\leq n}$, à partir d'une subdivision (t_k) de l'intervalle $[a, b]$ en n segments de même longueur, de telle manière que y_k soit une valeur approchée de $y(t_k)$.

Partant de
$$y_{k+1} - y_k = \int_{t_k}^{t_{k+1}} y'(u)\mathrm{d}u = \int_{t_k}^{t_{k+1}} \phi(u, y(u))\mathrm{d}u,$$
l'idée est d'approcher l'intégrale de droite par l'aire d'un trapèze.

Comme son nom l'indique cette méthode se passe en deux temps :

Prédiction : On utilise la méthode d'Euler pour trouver une valeur temporaire de y_{k+1} qu'on note $\tilde{y}_{k+1}$:

$$\tilde{y}_{k+1} = y_k + h \times \phi(t_k, y_k).$$

Correction : Ensuite on corrige le tir en prenant pour l'intégrale qui donne y_{k+1} à partir de y_k l'« aire » du trapèze construit sur y_k et $\tilde{y}_{k+1}$:

On obtient le schéma suivant :

$$\boxed{y_{k+1} = y_k + \int_{t_k}^{t_{k+1}} \phi(u, y(u))\mathrm{d}u \approx y_k + \frac{h}{2}\left(\phi(t_k, y_k) + \phi(t_{k+1}, \tilde{y}_{k+1}\right).}$$

— Le temps de calcul sera environ le double que celui de la méthode d'Euler, mais la convergence est plus rapide sous de bonnes conditions.

— On montre que l'erreur est $O(h^2)$, et on dit que c'est une méthode d'ordre 2.

Exercice 2.5. Écrire une fonction `prediction_correction(phi,a,b,y0,n)` qui prend comme arguments la fonction $\phi : [a, b] \times \mathbb{R} \to \mathbb{R}$, les bornes a et b de l'intervalle, la condition initiale y_0 et n le nombre d'intervalles de la subdivision, et qui retourne le couple (t, y) du tableau de valeurs des t_k et le tableau des valeurs y_k obtenues par cette méthode.

Solution.

Les méthodes RK à pas constant

Les méthodes de Runge-Kutta d'intégrations différentielles, utilisent la même idée que celle de la méthode d'Euler, mais utilise des pentes pour l'approximations prises en d'autres points de y_k. ••

1. La méthode RK1 est celle d'Euler :

$$y_{k+1} = y_k + h \cdot \phi(t_k, y_k).$$

2. Pour la méthode RK2 on utilise la pente au milieu de $[t_k, t_{k+1}]$:

   (a) On approche le milieu par : $y_m = y_k + \dfrac{h}{2}\phi(t_k, y_k)$.

(b) La pente par $y'_m = \phi\left(t_k + \frac{h}{2}, y_m\right)$.

(c) On définit : $\boxed{y_{k+1} = y_k + h \cdot y'_m}$.

Sous de bonnes hypothèses on peut montrer que l'erreur est en $O(h^2)$.

3. La méthode RK4 est la plus utilisée. Elle consiste à utiliser une moyenne pondérée des estimations des pentes en début, milieu et fin de l'intervalle $[t_k, t_{k+1}]$:

   (a) $\alpha_1 = \phi(t_k, y_k)$ pente en t_k.

   (b) $\alpha_2 = \phi(t_k + h/2, y_k + \alpha_1 \cdot h/2)$ pente au milieu en prenant α_1 pour pente en partant de y_k.

   (c) $\alpha_3 = \phi(t_k + h/2, y_k + \alpha_2 \cdot h/2)$ pente au milieu en prenant α_2 pour pente en partant de y_k.

   (d) $\alpha_4 = \phi(t_{k+1}, y_k + \alpha_3 \cdot h)$ pente en t_{k+1} en prenant α_3 pour pente en partant de y_k.

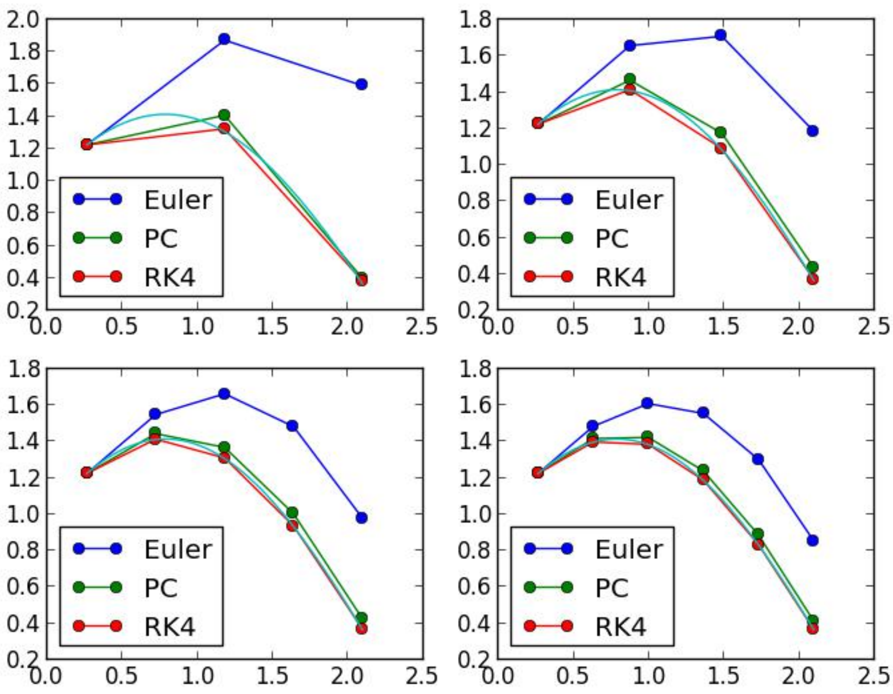
   On pose alors :

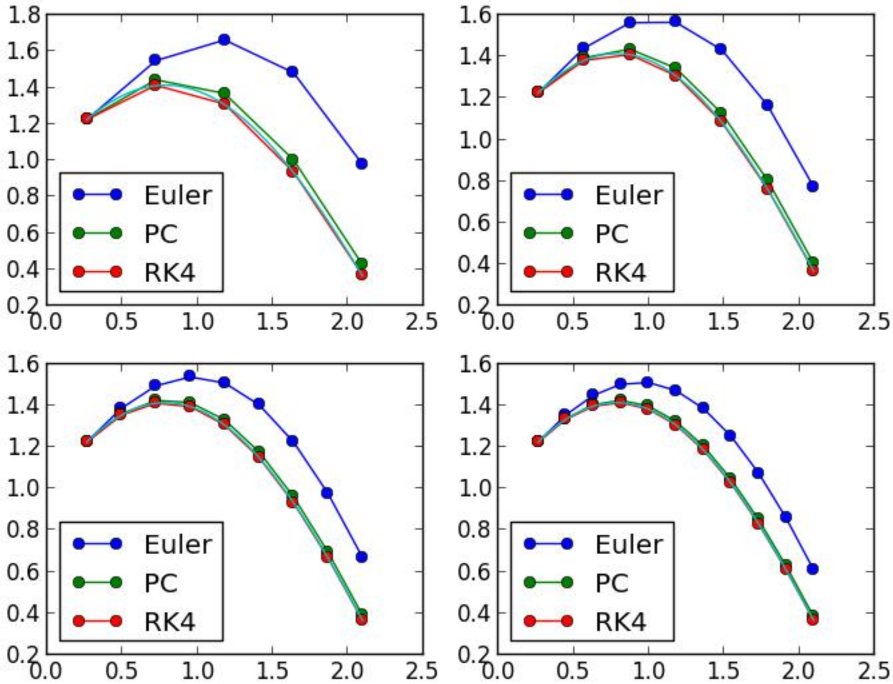
$$\boxed{y_{k+1} = y_k + \frac{h}{6}(\alpha_1 + 2\alpha_2 + 2\alpha_3 + \alpha_4).}$$

   On peut montrer que l'erreur commise est en $O(h^4)$.

Comparaison graphique

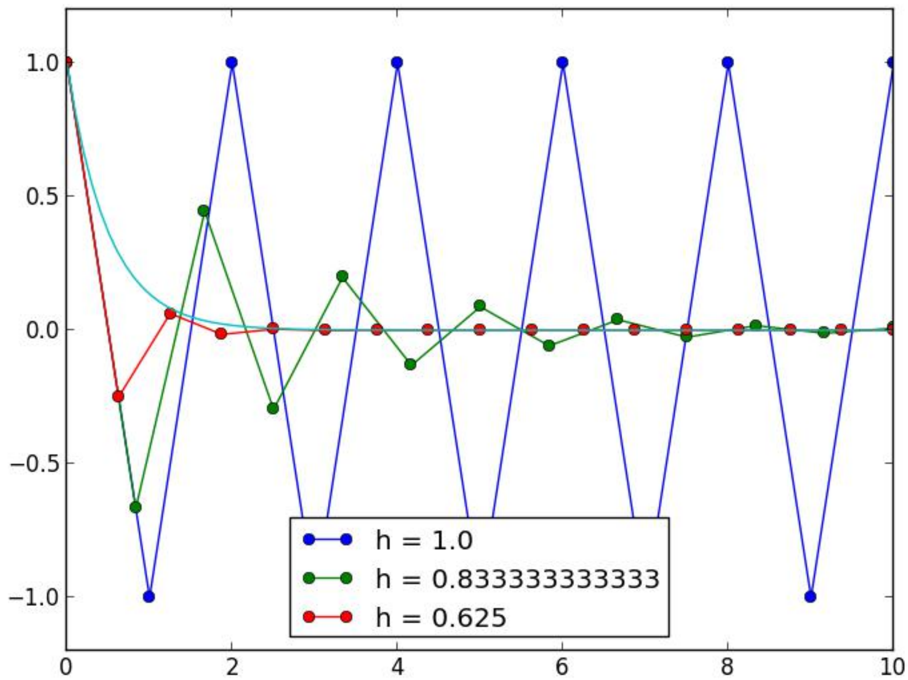
On considère le problème de Cauchy : $\sin(t)y' - \cos(t)y + 1 = 0$ sur $[\pi/12, 3 * \pi/2]$ et $y(\pi/2) = 1$.





Choix du pas

L'équation : $y' = -2y$ et $y(0) = 1$, avec la méthode d'Euler :



Système différentiels

Un exemple : Les Équations de Lotka-Volterra qui décrivent la dynamique de systèmes biologiques dans lesquels un prédateur et sa proie interagissent.

$$\begin{cases} x'(t) &=& x(t)\left(\alpha - \beta y(t)\right) \\ y'(t) &=& -y(t)\left(\gamma - \delta x(t)\right) \end{cases}$$

- t est le temps.
- $x(t)$ est l'effectif des proies en fonction du temps.
- $y(t)$ est l'effectif des prédateurs en fonction du temps.
- x' et y' représentent la variation des populations au cours du temps.

$$\begin{cases} x'(t) &=& x(t)\left(\alpha - \beta y(t)\right) \\ y'(t) &=& -y(t)\left(\gamma - \delta x(t)\right) \end{cases}$$

Les paramètres caractérisent les interactions entre les deux espèces :

- α taux de reproduction des proies (constant et indépendant du nombre de prédateurs).
- β taux de mortalité des proies dû aux prédateurs rencontrés.
- γ taux de mortalité des prédateurs (constant et indépendant du nombre de proies).
- δ taux de reproduction des prédateurs en fonction des proies rencontrées et mangées.

●●

1. Comment se ramener à ce que l'on connait ?
2. L'idée c'est que deux scalaires... c'est un vecteur !
3. Posons : $Y(t) = (x(t), y(t))$ et

$$\Phi(t, Y(t)) = \left(x(t)\left(\alpha - \beta y(t)\right), -y(t)\left(\gamma - \delta x(t)\right)\right).$$

Alors notre système s'écrit :

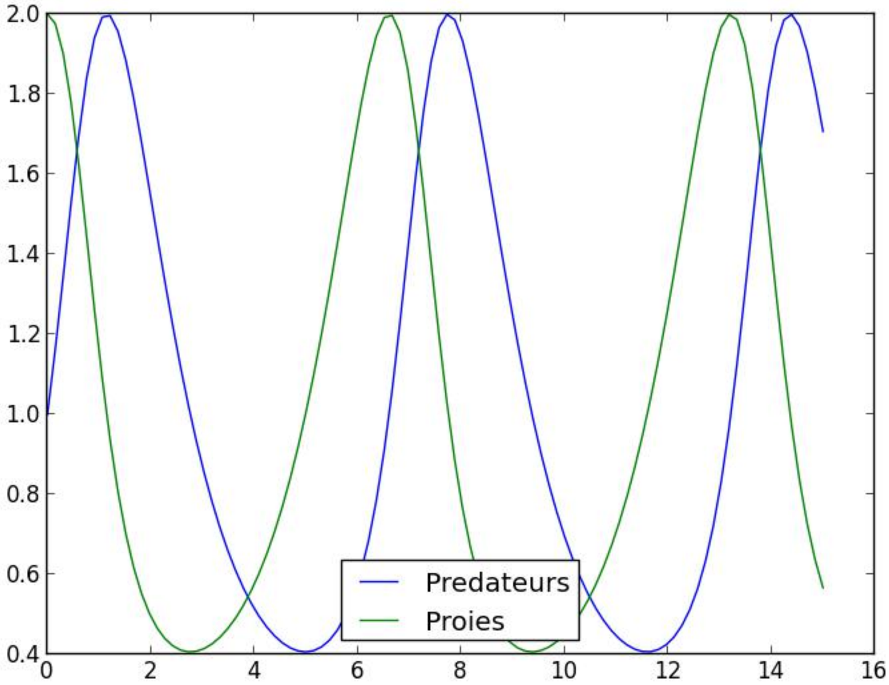
$$Y'(t) = \Phi(t, Y(t)).$$

La théorie montre que l'on peut utiliser la même méthode pour approcher les solutions :

$$\boxed{Y_{k+1} = Y_k + h \times \Phi(t_k, Y_k).}$$

La méthode `odeint()` permet de résoudre les systèmes différentiels d'ordre 1 avec condition initiale :

```
>>> import numpy as np
>>> import scipy.integrate as spi
>>>phi = lambda Y,t :np.array([Y[0]*(1-1*Y[1]),-Y[1]*(1-1*Y[0])])
>>> t = np.linspace(0,15,100)
>>> Y0 = np.array([2,1])
>>> Y = spi.odeint(phi,Y0,t)
>>> x = Y[:,0] # Recuperation des valeurs de x
>>> y = Y[:,1] # Recuperation des valeurs de y
>>> plot(t,y,label='Predateurs')
[<matplotlib.lines.Line2D object at 0x105bbb110>]
>>> plot(t,x,label='Proies')
[<matplotlib.lines.Line2D object at 0x105bbdc90>]
>>> legend(loc='lower center')
>>> show()
```


2.0
1.8
1.6
1.4
1.2
1.0
0.8
0.6
0.4
0
2
4
6
8
10
12
14
16
Predateurs
Proies

Exercice 2.6. Adapter notre fonction `euler_explicite` au cas d'un système différentiel.

Solution.

Savoir approcher les solutions d'un système différentiel permet d'approcher les solutions d'une équation différentielle d'ordre 2. En effet, notre problème de Cauchy d'ordre 2 :

$$y'' = \phi(t, y, y') \text{ avec } y(0) = y_0 \text{ et } y'(0) = y_1.$$

On pose $Y = \begin{pmatrix} y \\ y' \end{pmatrix}$, alors $Y' = \begin{pmatrix} y' \\ y'' \end{pmatrix}$, et il suffit de trouver ϕ telle que :

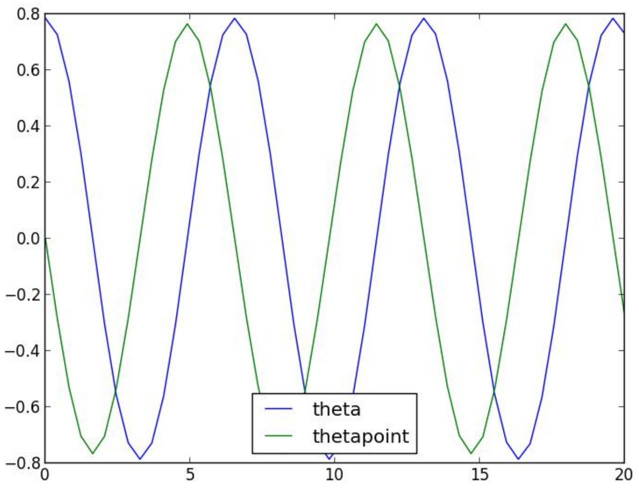
$$Y' = \phi(t, Y).$$

Par exemple, pour le pendule simple :

$$\ddot{\theta} + \omega^2 \sin(\theta) = 0 \text{ avec } \theta(0) \text{ et } \dot{\theta}(0) \text{ donnés.}$$

On pose :

$$Y' = \begin{pmatrix} \dot{\theta} \\ \ddot{\theta} \end{pmatrix} = \phi(t, \begin{pmatrix} \theta \\ \dot{\theta} \end{pmatrix}) = \begin{pmatrix} \dot{\theta} \\ -\sin(\theta) \end{pmatrix}.$$



2.3 Exercices

Exercice 2.7. On considère le problème de Cauchy suivant :

$$\begin{cases} y'(t) &= y(t) \\ y(0) &= 1 \end{cases}$$

que l'on étudie sur l'intervalle $[0, T]$, où $T > 0$.

1. Résoudre le problème...
2. Soit $n \in \mathbb{N}^*$, on note $(t_k)_{0 \leq k \leq n}$, la subdivision de $[0, T]$ a pas constant h. Expliciter h.
3. Déterminer la suite $(y_k)_{0 \leq k \leq n}$ définie par la méthode d'Euler explicite pour ce problème.
4. Montrer que la quantité $\max\limits_{0 \leq k \leq n} |y_k - \exp(t_k)|$ tend vers 0 quand h tend vers 0.
5. Déterminer la suite $(z_k)_{0 \leq k \leq n}$ définie par la méthode d'Euler implicite pour ce problème.

Exercice 2.8. On considère l'équation différentielle suivante :

$$\begin{cases} y'(t) &= -\lambda(y(t) - \sin(t)) \\ y(0) &= 1 \end{cases},$$

que l'on étudie sur l'intervalle $[0, T]$, où $T > 0$.

1. Résoudre l'équation différentielle.
2. Soit $n \in \mathbb{N}^*$, on note $(t_k)_{0 \leq k \leq n}$, la subdivision de $[0, T]$ a pas constant h. Expliciter h.
3. Déterminer la suite $(y_k)_{0 \leq k \leq n}$ définie par la méthode d'Euler explicite pour ce problème.
4. Vérifier que la solution approchée est proche de la solution exacte si et seulement si $|1 - \lambda h| < 1$.
5. Déterminer la suite $(z_k)_{0 \leq k \leq n}$ définie par la méthode d'Euler implicite pour ce problème.
6. Comparer les résultats obtenus.

Exercice 2.9. Écrire une fonction `euler_implicite(phi,y0,tps,dphi1)` d'arguments une fonction `phi(t,y)`, un flottant `y0`, une liste (ou tableau) de temps `tps = [t_0, . . . ,t_n]` et une fonction `dphi1(t,y)` qui est la dérivée partielle de `phi` par rapport à sa première variable, et qui renvoie le tableau des valeurs approchées aux temps `t_i` de la solution de l'équation différentielle $y(t) = \phi(t, y(t))$ vérifiant la condition initiale $y(t_0) = y0$, les valeurs approchées étant calculées avec la méthode d'Euler implicite. L'équation étant résolue par la méthode de Newton.

Corrigé 2.1 Le code :

Corrigé 2.2 Le code :

Corrigé 2.3

Corrigé 2.4 Le code :

```
def euler_explicite(phi,a,b,y0,n):
    import numpy as np
    t = np.linspace(a,b,n+1) #subdivision
    y = np.empty(n+1)  #tableau vide
    y[0] = y0
    h = (b-a)/n
    for k in range(n):
        y[k+1] = y[k] + h*phi(t[k],y[k])
    return (t,y)
```

Corrigé 2.5 Le code :

```
def prediction_correction(phi,a,b,y0,n):
    import numpy as np
    t = np.linspace(a,b,n+1)
    y = np.empty(n+1)
    y[0] = y0
    h = (b-a)/n
    # On ne calcule qu'une fois tp
    for k in range(n):
        tp = phi(t[k],y[k])
        y_tp = y[k] + h*tp
        y[k+1] = y[k] + (h/2)*(tp+phi(t[k+1],y_tp))
    return (t,y)
```

Corrigé 2.6 Le code :

```
def euler_explicite_sys(phi,a,b,Y0,n):
        p = len(Y0)
        t = np.linspace(a,b,n+1) #subdivision
        Y=np.zeros((n+1,p)) #tableau vide
        Y[0]=Y0 # fonctionne aussi si Y0 est une liste
        h = (b-a)/n
```