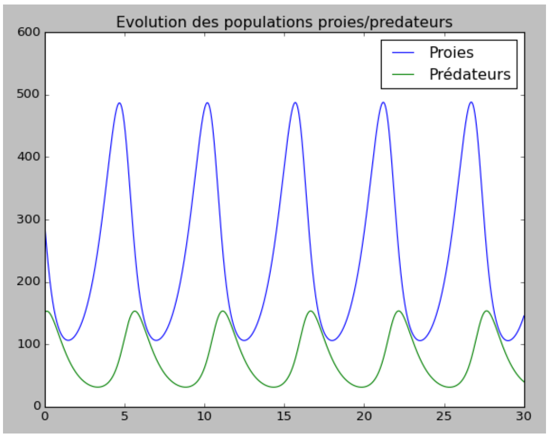
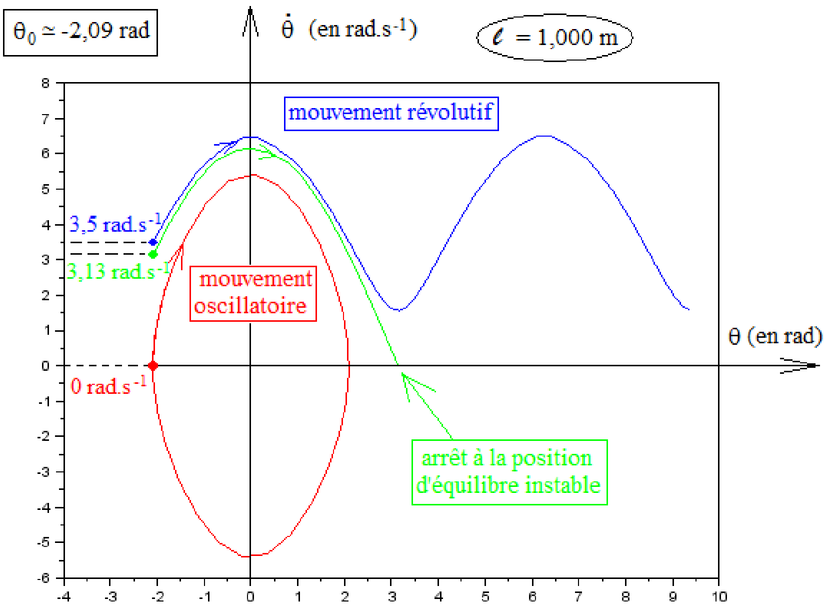
Résolution numérique d'équations différentielles d'ordre 2

- Systèmes différentiels -





Résolution d'équations différentielles d'ordre 2 - Systèmes différentiels -

Comment résoudre : y''+ ay'+by = c ??

1. Le système différentiel

$$\{P\} \begin{cases} x' = f(x,y,t) & \text{pour } \forall t \in [a,b] \\ y' = g(x,y,t) \end{cases}$$

 # on a un système d'équations

et 2 C.I $\begin{cases} x(a)= x_0 \\ y(a)= y_0 \end{cases}$

Cf TP ex1

avec f,g à 3 variables de $R^3 \to R$ *(pas nécessairement linéaire)*

Ici les **inconnues sont les 2 fonctions : t → x(t) et y → y(t)**
On peut avoir des systèmes différentiels à n inconnues…

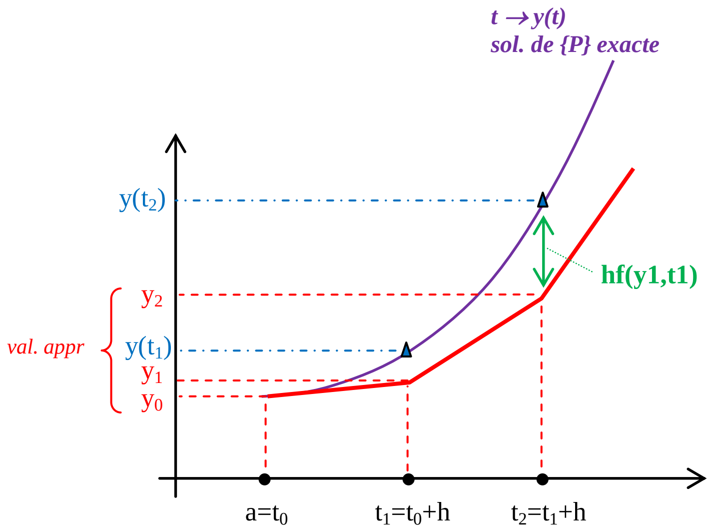
Idée : appliquer la méthode d'EULER à ce système…

2. Rappels sur Euler

$$\begin{cases} \forall t \in [a,b],\ y' = f(y,t) \\ y(a) = y_0 \end{cases} \quad (1)$$

On a notre équation différentielle. On sait qu'il y a une solution et on va l'approximer en chaque valeur par le polygone d'Euler *(y_i approximation des $y(t_i)$)*. On remplace la courbe par **sa tangente.**

$\forall n \in N$, $\boxed{y_{n+1} = y_n + hf(y_n,t_n)}$



3. Euler vectoriel

Attention, maintenant il y a 2 équations ⇒ on fait 2 fois Euler…

On a 3 C.I : $\begin{cases} x_0=x_0 \\ y_0=y_0 \\ t_0=a \end{cases}$

Attention, le rôle de y est joué par x dans Euler simple :
et ne pas oublier la variable de discrétisation du temps :

$$\begin{array}{|l} x_{i+1} = x_i + hf(x_i,y_i,t_i) \\ y_{i+1} = y_i + hg(x_i,y_i,t_i) \\ t_{i+1} = t_i + h \quad \textit{avec } h=(b-a)/n \end{array}$$

TP *Dans ex1, Q1 TP, on donne :f,g,x0,y0, [a,b] et n. En sortie, on veut les listes de X,Y,T*

*Programmer la fonction **euler_vectoriel** sous python*

Résolution d'équations différentielles d'ordre 2 - Systèmes différentiels -

4. Modèle de Lokta-Volterra (proie-prédateur)

Ces 2 scientifiques italiens ont fait une étude dans l'atlantique, après la seconde guerre mondiale, il y avait moins de sardines et plus de requins qu'avant. Pourquoi ? Peut-on modéliser cela par un **système différentiel** ?

TP *Dans ex1, Q2 TP :*

$$\left| \begin{array}{l} f(x,y,t) = x(\alpha-\beta y) \\ g(x,y,t) = -y(\gamma-\delta x) \end{array} \right.$$ *avec x(t) = nb proies instant t et y(t) = nb prédateurs instant t*

<u>Req</u> : si $\beta=0$ ds équation 1 (pas prédateurs), $x'=\alpha x$ $\rightarrow$ dont solution est e^{x} (nb sardines croit façon expo)
si $\delta=0$ ds équation 2 (pas proies), $y'=-\gamma y$ $\rightarrow$ dont solution est e^{-y} (nb prédateurs en décroissance expo)

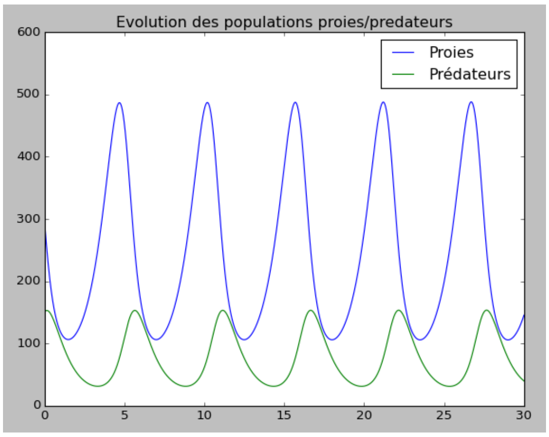
Donc si les proies meurt c'est proportionnel au nombre de prédateurs et prédateurs se reproduisent proportionnellement au nombre de proies !!

On va utiliser la fonction Q1 : euler_vectoriel

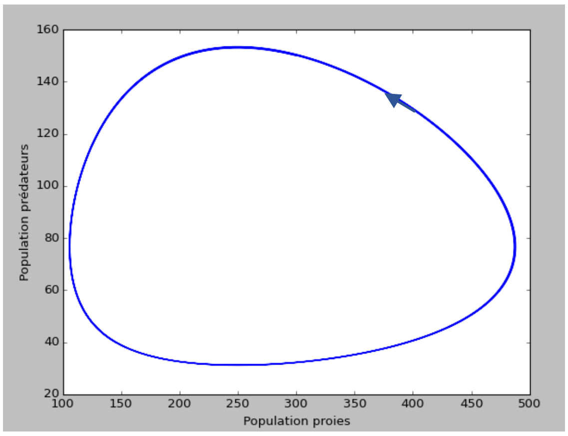
Programmer le modèle de Lokta Volterra et répondre aux questions a) et b).

<u>Solution :</u>

a) Les 2 courbes sont **périodiques et déphasées**



b) Le tracé des points de coordonnées x_t,y_t donne un « **diagramme de phase** »



Résolution d'équations différentielles d'ordre 2
- Systèmes différentiels -

5. Equation différentielle du 2^{nde} ordre (équation diff. Scalaire)

Cas général : y'' + ay' + by = c avec a,b,c des applications de I dans R : I→R

L'objectif est de **se ramener à un système différentiel du 1^{er} ordre.**

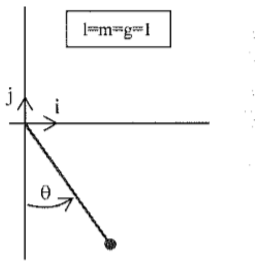
Astuce : on fait un changement de variable

On pose : $\begin{cases} y' = z \\ z' = y'' = -az - by + c \end{cases}$

⇔ $\boxed{\begin{array}{l} y' = f(y,z,t) \\ z' = g(y,z,t) \end{array}}$ *# on a bien un système différentiel*

TP *Dans ex1, Q3.a) TP pendule*

On va transformer l'équation différentielle du pendule en un système du 1^{er} ordre et on va utiliser l'algorithme d'Euler (ex1) pour résoudre ce système.
On aura juste à donner ce que sont les fonctions f et g et utiliser Euler.



Pendule mathématique (pas gravité, pas masse, pas frottement…tout est à 1) :

Equation diff. qui régit θ : $\boxed{\theta'' + \sin(\theta) = 0}$ avec θ angle pendule avec verticale

*Req : on ne sait pas résoudre, sauf aux petites approximations → **résolution numérique***

On fait un changement de variable :

⇔ $\begin{cases} \theta' = f(\theta,\phi,t) \\ \phi' = g(\theta,\phi,t) \end{cases}$

⇔ $\boxed{\begin{array}{l} f(\theta,\phi,t) = \phi \\ g(\theta,\phi,t) -\sin(\theta) \end{array}}$

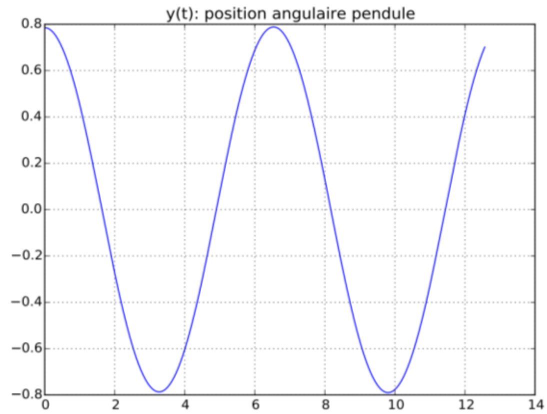
On va rappeler Euler de ex1 Q1 et définir f et g

Programmer le modèle du pendule selon questions du TP.

Solutions:

i) graphique **position θ = f(t)**

sinusoide entretenue entre [-π/4,π/4]
car pas de frottements



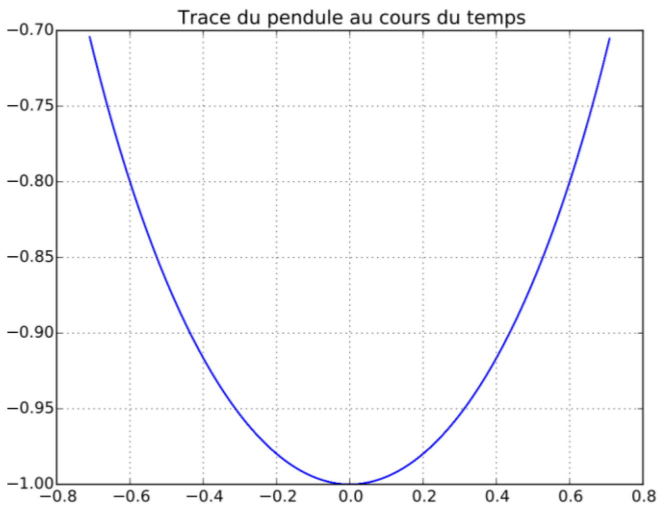
Résolution d'équations différentielles d'ordre 2 - Systèmes différentiels -

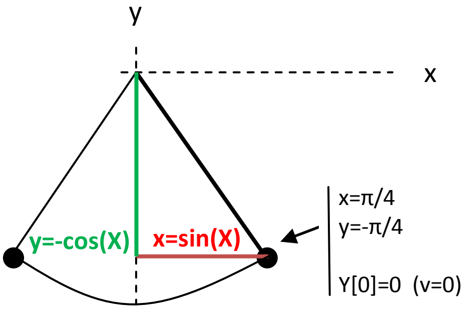
ii) graphique **position θ en x et y obtenues par projections** ☐

théoriquement un arc de cercle sans amortissement
(pas tout à fait car échelles différentes en x et y)

Euler via système va retourner **3 LISTES** :

- T : discrétisation du temps
- X : position θ (angle)
- Y : vitesse θ'





iii) graphique **énergie** : celle-ci **n'est pas conservée** (faible amplitude) dû fait des hypothèses sur l,m,g = 1.

→ méthode numérique pas parfaite (erreurs d'arrondis) donc **schéma numérique dissipatif**