

Résolutions d'équations différentielles par une méthode numérique

I Parabole de sûreté

1 Position du problème

Un projectile est lancé depuis l'origine du repère O avec la vitesse initiale $v_0 = 10 \text{ m.s}^{-1}$, avec un angle α par rapport à l'horizontale, dans un champ de pesanteur $\vec{g} = -g\vec{u}_y = -9,8\vec{u}_y$ et dans le cas où les frottements sont négligeables.

La portée du tir est égale à $x_{\max} = \frac{v_0^2}{g} \sin(2\alpha) = 10,2 \times \sin(2\alpha)$; maximale pour $\alpha = \pi/4$

L'ensemble des points accessibles par le projectile, pour une vitesse initiale v_0 donnée, se situent sous une courbe appelée parabole de sûreté et d'équation $y_s(x) = \frac{v_0^2}{2g} - \frac{g}{2v_0^2} x^2$

Démonstration :

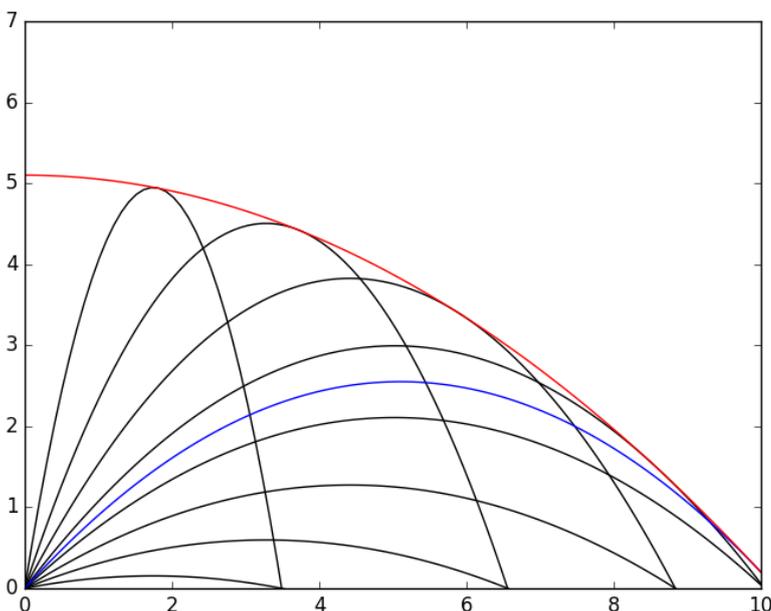
Déterminer l'équation de la trajectoire $y_s(x)$ délimitant la zone des points que peut atteindre le projectile lancé de O avec une vitesse v_0 donnée, de norme constante et de direction α (entre 0 et $\pi/2$).

Aide : écrire l'équation de la trajectoire parabolique $y(x)$ puis remplacer $1/(\cos\alpha)^2$ par $1 + (\tan\alpha)^2$ afin d'obtenir une équation du second degré en $\tan\alpha$. Se demander alors à quelle condition (sur y et x) il existe un angle α tel que le point M de coordonnées (x,y) peut être atteint (à v_0 fixé).

2 Tracés des courbes avec Python

- Par résolution numérique de l'équation différentielle de la chute libre $\ddot{y}(t) = -g$, tracer sur le même graphe plusieurs trajectoires pour $v_0 = 10 \text{ m.s}^{-1}$ et pour différentes valeurs de α (tracer d'une couleur différentes la trajectoire correspondant à $\alpha = \pi/4$).
- Tracer sur le même graphe et d'une autre couleur la parabole de sûreté.

3 Courbes et commentaires



A l'oral :

- vidéo-projeter et expliquer succinctement la démonstration de l'équation de la parabole de sûreté (sans détailler les calculs) en expliquant clairement ce qu'elle représente.
- interpréter les courbes tracées.
- vidéo projeter le programme Python et commenter-le succinctement.

II Oscillations libres dans le cas d'un système du second ordre

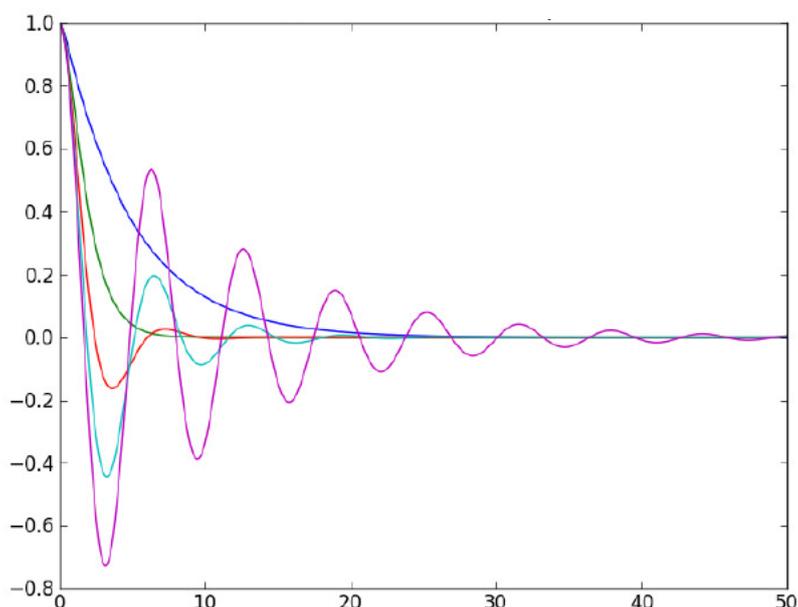
1 Position du problème

L'allongement x d'un ressort horizontal soumis à des frottements fluides (ou la tension aux bornes du condensateur dans le cas d'un circuit RLC série) vérifie l'équation différentielle : $\ddot{x} + \frac{\omega_0}{Q} \dot{x} + \omega_0^2 x = 0$

2 Résolution numérique avec Python

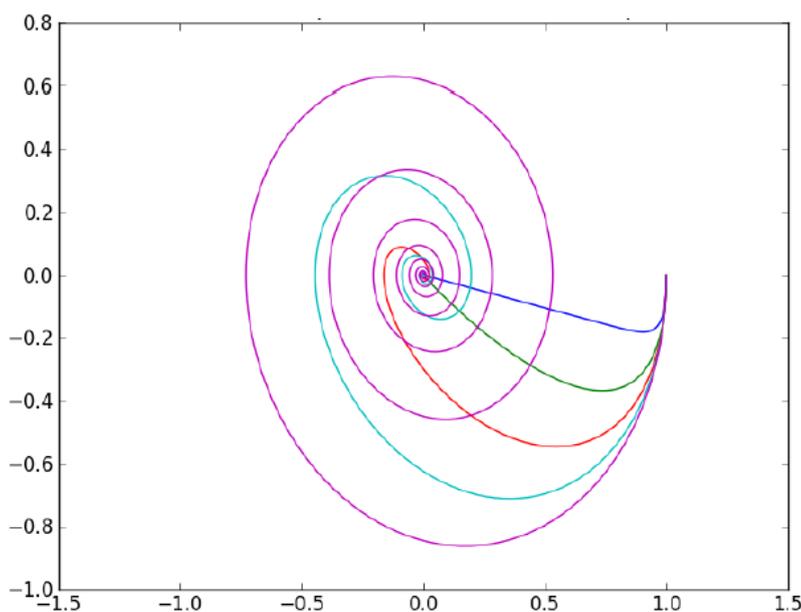
- Tracer $x = f(t)$ par résolution numérique de l'équation différentielle pour différentes valeurs de Q ($Q < \frac{1}{2}$; $Q = \frac{1}{2}$; $Q > \frac{1}{2}$)
- Tracer le portrait de phase $\dot{x} = f(x)$ pour les mêmes valeurs de Q que précédemment.

3 Courbes et commentaires



A l'oral :

- vidéo projeter et expliquer succinctement la démonstration des solutions analytiques de l'équation différentielle.
- commenter les courbes $x = f(t)$ en fonction de la valeur de Q .
- interpréter les courbes correspondantes dans le plan de phase.
- vidéo projeter le programme Python et commenter la méthode de résolution.



III Pendule non linéaire

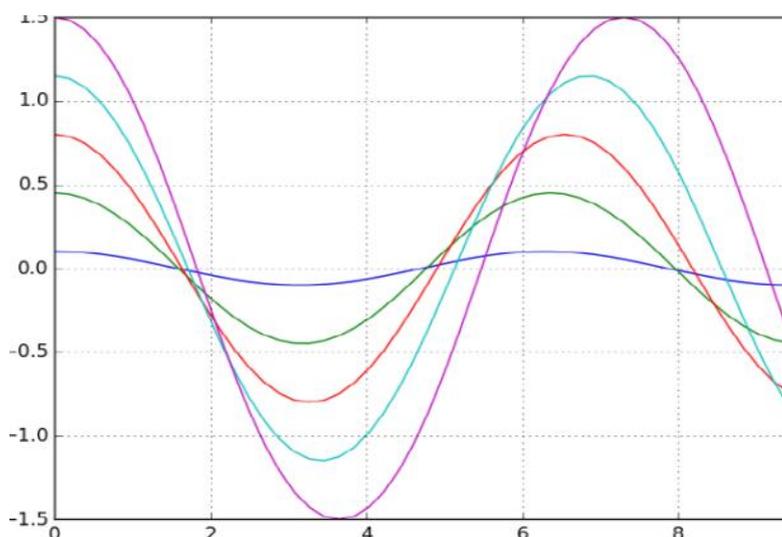
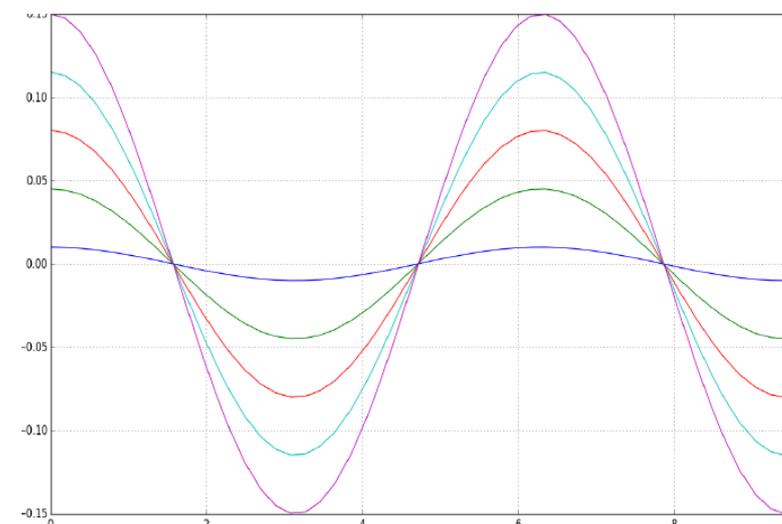
1 Position du problème

Soit un pendule de masse m et de longueur L , de pulsation propre $\omega_0 = \sqrt{g/L}$. L'équation différentielle vérifiée par l'angle θ entre la direction du pendule et la verticale est : $\ddot{\theta} + \omega_0^2 \sin(\theta) = 0$

2 Résolution numérique avec Python

- Effectuer le changement de variable $t^* = \omega_0 t$ afin d'adimensionner l'équation différentielle ; on notera $\dot{\theta}^* = d\theta/dt^*$.
- Tracer sur un même graphe plusieurs courbes $\theta = f(t^*)$ pour $\dot{\theta}(0) = 0$ et $\theta(0) \in [0,01 ; 0,1]$
- Tracer sur un autre graphe plusieurs courbes $\theta = f(t^*)$ pour $\dot{\theta}(0) = 0$ et $\theta(0) \in [0,15 ; 1,5]$
- Tracer sur un même graphe les courbes $\dot{\theta}^* = f(\theta)$ pour $\dot{\theta}(0) = 0$ et pour toutes les valeurs de $\theta(0)$ précédentes.

3 Courbes et commentaires



A l'oral :

- vidéo projeter et expliciter succinctement la démonstration de l'équation différentielle adimensionnée du pendule.
- justifier les tracés dans le cas des petites oscillations
- justifier les tracés dans le cas des oscillations d'amplitude plus importante
- interpréter les courbes tracées dans le plan de phase
- vidéo projeter le programme Python et commenter la méthode de résolution.

IV Particule chargée dans un champ électromagnétique

1 Position du problème

- Cas 1 : un électron de charge $-e$ et de masse m est placé au point O (origine du repère) avec une vitesse initiale $\vec{V}_0 = V_0 \vec{u}_x$ dans un champ magnétique $\vec{B} = B \vec{u}_z$.

Les vitesses V_x et V_y vérifient les équations : $\dot{V}_x = -\omega V_y$ et $\dot{V}_y = \omega V_x$ avec $\omega = \frac{eB}{m}$

- Cas 2 : une particule de charge positive q et de masse m est placée au point O (origine du repère) sans vitesse initiale dans un champ électromagnétique : $\vec{E} = E \vec{u}_y$ et $\vec{B} = B \vec{u}_z$

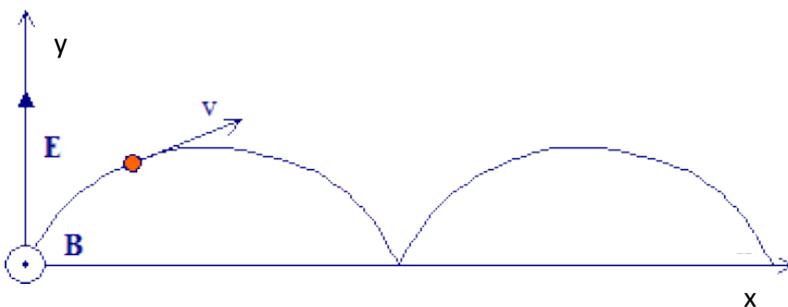
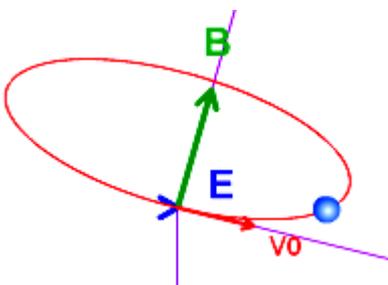
Les vitesses V_y et V_z vérifient les équations : $\dot{V}_x = \omega V_y$ et $\dot{V}_y = -\omega V_x + \omega \frac{E}{B}$ avec $\omega = \frac{qB}{m}$

- Effectuer le changement de variable $t^* = \omega t$ afin d'adimensionner les équations différentielles précédentes ; on notera $V_x^* = dx/dt^*$ et $V_y^* = dy/dt^*$; $x^* = \frac{B\omega}{E}x$ et $y^* = \frac{B\omega}{E}y$

2 Résolution numérique avec Python

- Tracer la trajectoire $y^* = f(x^*)$ dans le cas 1.
- Comment est censée évoluer la norme de la vitesse et pourquoi ? Cette propriété est-elle vérifiée par les solutions des équations différentielles ?
- Tracer la trajectoire $y^* = f(x^*)$ dans le cas 2.

3 Courbes et commentaires



A l'oral :

- vidéo projeter et commenter succinctement les démonstrations des équations différentielles vérifiées par les coordonnées de la particule.
- comparer les courbes obtenues avec les solutions analytiques, celles du cas 1 ayant été vues en cours et celles du cas 2 étant :

$$\begin{cases} x(t) = \frac{E}{B\omega} [\omega t - \sin(\omega t)] \\ y(t) = \frac{E}{B\omega} [1 - \cos(\omega t)] \end{cases}$$

- vidéo projeter le programme Python et commenter la méthode de résolution.

V Les différentes trajectoires d'un système soumis à un champ newtonien en $1/r^2$

1 Position du problème

Un système de masse m soumis à un champ gravitationnel (centripète en $1/r^2$), aussi appelé champ Newtonien, suit une trajectoire de type conique : une hyperbole, une parabole ou une ellipse (dont le cercle est un cas particulier). La trajectoire suivie dépend des conditions initiales.

On peut prouver que les coordonnées polaires (r, θ) décrivant la position du système vérifient les équations différentielles :

$$\begin{cases} \frac{d^2 r}{dt^2} = \frac{1}{r^3} - \frac{1}{r^2} \\ \frac{d\theta}{dt} = \frac{1}{r^2} \end{cases}$$

(remarque : il s'agit en fait de coordonnées réduites sans dimension $r = \frac{OM}{r_0}$)

2 Résolution numérique avec Python

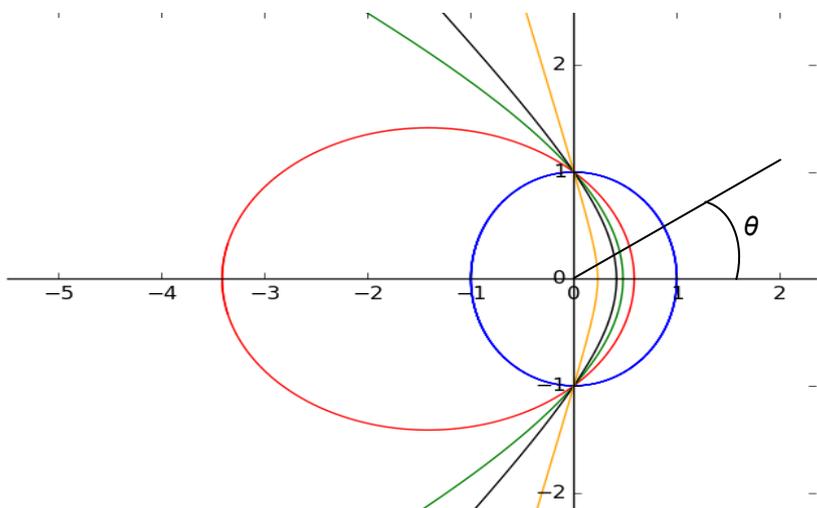
Tracer $r = f(\theta)$ pour un système suivant les équations différentielles précédentes avec les différentes conditions initiales suivantes :

$$\begin{cases} \theta(t=0) = 0 \\ \frac{dr}{dt}(t=0) = 0 \\ r(t=0) = \frac{-1 + \sqrt{1 + \alpha}}{\alpha} \end{cases} \quad \begin{array}{l} \text{avec } \alpha = -1 ; \alpha \in]-1, 0[; \alpha = 0 ; \alpha > 0 \\ \alpha \text{ étant un paramètre constant égal à l'énergie mécanique} \\ \text{adimensionnée du système : } \alpha = E_m / E_0 \in [-1, +\infty[\end{array}$$

Remarques :

- un rappel qui peut-être utile : les coordonnées cartésiennes sont égales à $x = r \cos(\theta)$ et $y = r \sin(\theta)$
- A la date $t = 0$, l'angle θ est fixé à 0, la valeur de α est fixé à 0 (donc tangente initiale selon $\vec{u}_{\theta=0}$ c'est-à-dire verticale) mais la valeur de $r(t = 0)$ dépend de la valeur de α (donc de la valeur de l'énergie mécanique) et n'est pas la même pour chaque courbe tracée.

3 Courbes et commentaires



A l'oral :

- décrire les différentes courbes coniques obtenues.
- interpréter les différents types de courbes obtenues (courbes fermées ou ouvertes).
- vidéo projeter le programme Python et commenter la méthode de résolution.

VI Oscillations dans un puits de potentiel non linéaire

1 Position du problème

Soit une particule de masse m située dans un puits non linéaire d'énergie potentielle $E_p(x) = E_0[1 - e^{-\frac{x}{L}}]$ ou $E_p^*(x) = 1 - e^{-\frac{x}{L}}$ en utilisant la variable adimensionnée $E_p^*(x) = E_p(x)/E_0$

Déterminer la force qui dérive de l'énergie potentielle $E_p(x)$ et en déduire l'équation différentielle vérifiée par l'abscisse x de la particule (dans le cas où les frottements sont négligeables).

En posant les variables adimensionnées $x^* = \frac{x}{L}$ et $t^* = \frac{t}{t_0}$ avec $t_0 = \frac{L}{v_0}$ et $v_0 = \sqrt{\frac{E_0}{m}}$ on peut

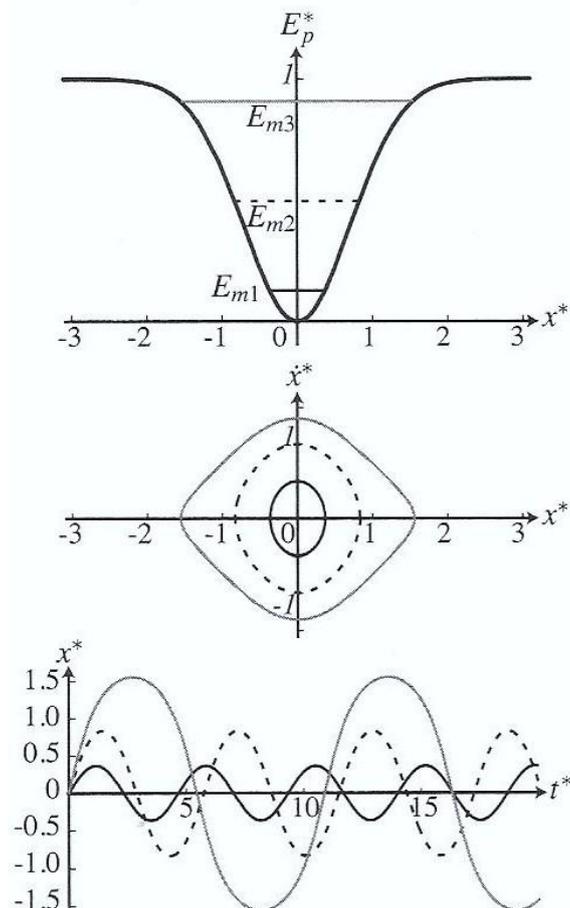
montrer que x^* vérifie l'équation différentielle non linéaire : $\dot{x}^* = -2x^* \exp(-x^{*2})$ (démonstration non demandée)

2 Résolution numérique avec Python

Nous allons résoudre cette équation différentielle avec Python dans le cas $x^*(0) = 0$ et pour différentes valeurs de $\dot{x}^*(0)$.

- Tracer la courbe $E_p^*(x^*) = 1 - e^{-x^{*2}}$
- Faire apparaître sur ce graphe différentes valeurs de l'énergie mécanique adimensionnée $E_m^* = \frac{E_m}{E_0} = \frac{1}{2}(\dot{x}^*(0))^2$ pour différentes valeurs de $\dot{x}^*(0)$.
- Tracer $\dot{x}^* = f(x^*)$ pour les différentes valeurs de $\dot{x}^*(0)$ précédentes.
- Tracer $x^* = f(t^*)$ pour les différentes valeurs de $\dot{x}^*(0)$ précédentes.

3 Courbes et commentaires



A l'oral :

- vidéo projeter et commenter succinctement la démonstration de l'équation différentielle vérifiée par x (pas celle vérifiée par x^*).
- commenter les tracés des courbes obtenues.
- vidéo projeter le programme Python et commenter la méthode de résolution.

VII Projectile avec frottement

1 Position du problème

Soit un projectile lancé avec une vitesse initiale \vec{v}_0 dans un champ de pesanteur $\vec{g} = -g\vec{u}_y = -9,8\vec{u}_y$ (l'axe (Oy) est dirigé vers le haut) et soumis à une force de frottement $\vec{f} = -k\vec{v}$

- Déterminer l'équation différentielle vérifiée par le vecteur vitesse \vec{v}
- Prouver que la vitesse atteint la valeur limite $v_{lim} = \sqrt{mg/k}$

En posant $v^* = \frac{v}{v_{lim}}$ et $t^* = \frac{g}{v_{lim}}t$, on peut prouver que la vitesse adimensionnée \vec{v}^* vérifie l'équation différentielle (1) : $\frac{d\vec{v}^*}{dt^*} = -v^*\vec{v}^* - \vec{u}_y$ (justification non demandée).

Dans le cas où $v_x^* = 0$ à chaque instant et en choisissant un axe (Oy) orienté vers le bas la norme $v^* = -v_y^*$

Montrer alors que v^* vérifie l'équation différentielle $\frac{dv^*}{dt^*} = 1 - v^{*2}$

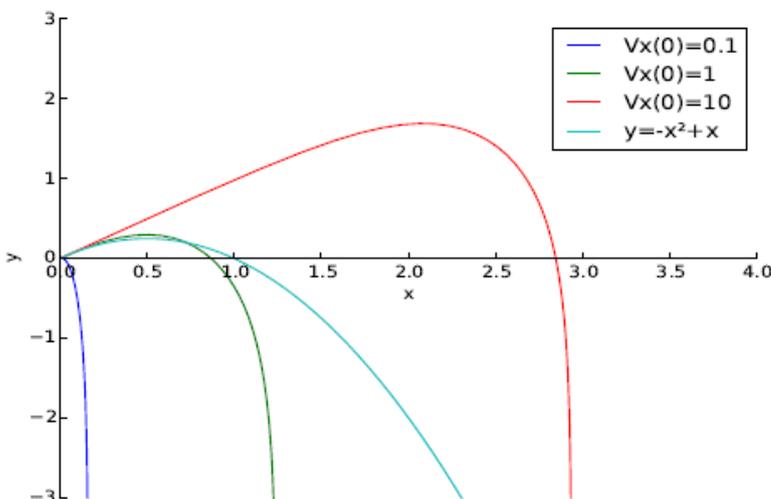
2 Résolution numérique avec Python

- De l'équation différentielle vectorielle (1), on en déduit le système différentiel suivant :

$$\begin{cases} \dot{v}_x^* = -v_x^* \sqrt{(v_x^*)^2 + (v_y^*)^2} & \text{avec } v_x^* = \frac{dx^*}{dt^*} \\ \dot{v}_y^* = -v_y^* \sqrt{(v_x^*)^2 + (v_y^*)^2} - 1 & \text{avec } v_y^* = \frac{dy^*}{dt^*} \end{cases}$$

- Tracer les trajectoires $y^* = f(x^*)$ pour un angle initial $\alpha = 45^\circ$ ($v_x^*(0) = v_y^*(0)$) et pour différentes valeurs de $v_x^*(0)$: $v_x^*(0) \ll 1$; $v_x^*(0) = 1$; $v_x^*(0) \gg 1$
- Sur un second graphe, tracer à nouveau la courbe correspondant à $v_x^*(0) = 1$ ainsi que la courbe d'équation $y^* = -x^{*2} + x^*$ correspondant à la trajectoire du projectile dans le cas $v_x^*(0) = 1$, $\alpha = 45^\circ$ et sans frottement.
- Sur un troisième graphe, tracer $v^* = f(t^*)$ dans le cas $v^*(0) = 0$,

3 Courbes et commentaires



A l'oral :

- vidéo projeter et commenter succinctement les démonstrations demandées.
- commenter les tracés des courbes obtenues.
- vidéo projeter le programme Python et commenter la méthode de résolution.

VIII Oscillations dans un double puits de potentiel

1 Position du problème

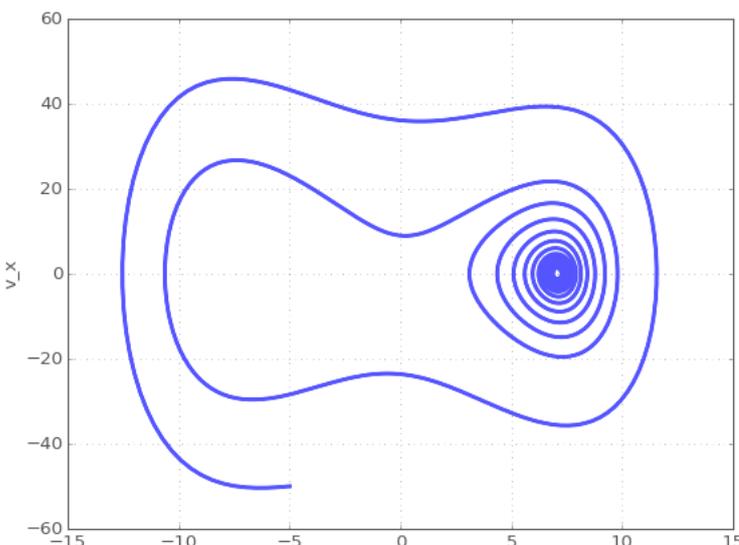
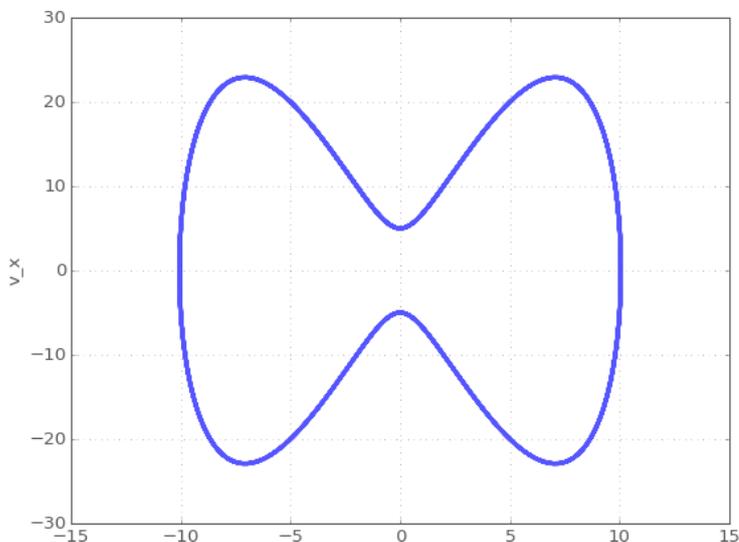
- Soit une particule de masse m située dans un puits d'énergie potentielle parabolique $E_p(x) = x^2$ centré sur $x = 0$. Déterminer la force qui dérive de l'énergie potentielle $E_p(x)$ et en déduire l'équation différentielle vérifiée par l'abscisse x de la particule (dans le cas où les frottements sont négligeables).
- Etablir de même l'équation différentielle vérifiée par l'abscisse x de la particule dans le cas où la force de frottement est selon (Ox) et du type $F_x = -\mu \frac{dx}{dt}$
- Pour un double puits paraboliques symétriques par rapport à $x = 0$, on peut proposer l'expression de l'énergie potentielle suivante : $E_p(x) = -4x^2 + 4x^4 + 1$

En déduire de même les équations différentielles vérifiées par x dans les cas avec et sans frottement.

2 Résolution numérique avec Python

- Tracer les courbes représentant $E_p(x)$ dans les cas du puits parabolique et du double puits.
- Résoudre les 4 équations différentielles.
- Tracer dans chacun des cas les évolutions de $x(t)$.
- Tracer dans chacun des cas les trajectoires dans le plan de phase.

3 Courbes et commentaires



A l'oral :

- vidéo projeter et commenter succinctement les démonstrations demandées.
- commenter les tracés des courbes obtenues.
- étudier l'influence de la vitesse initiale dans le cas du double puits.
- vidéo projeter le programme Python et commenter la méthode de résolution.

IX Influence de différents paramètres sur l'évolution d'un système chimiques

1 Position du problème

On étudie un équilibre chimique entre les espèces chimiques A, B, C et D.

On modélise cet équilibre par l'équation suivante : $A + B = C + D$ (avec des nombres stœchiométriques égaux à 1 pour simplifier).

On note K la constante de la réaction.

Du point de vu cinétique, on note k_1 la constante cinétique pour la réaction se faisant dans le sens direct et on considère pour simplifier que les ordres partiels sont égaux à 1 pour les espèces A et B.

On note de la même façon k_2 la constante cinétique pour la réaction se faisant dans le sens inverse et on considère pour simplifier que les ordres partiels sont égaux à 1 pour les espèces C et D.

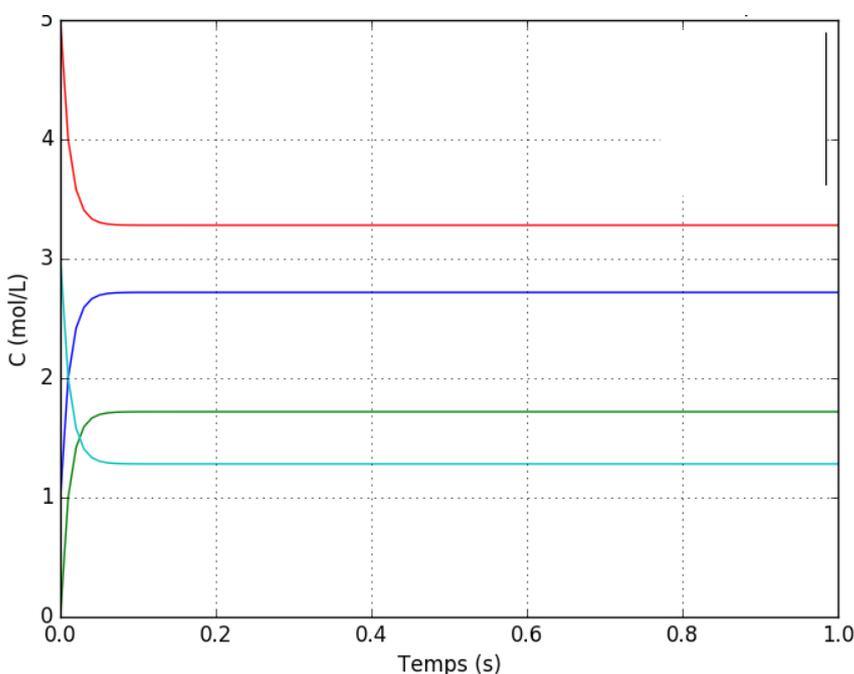
- Justifier que $\frac{d[A]}{dt} = -k_1[A][B] + k_2[C][D]$
- En déduire par analogie les expressions de $\frac{d[B]}{dt}$, $\frac{d[C]}{dt}$ et $\frac{d[D]}{dt}$.
- Démontrer l'expression de la constante de réaction K en fonction des constantes cinétiques k_1 et k_2 .

2 Résolution numérique avec Python

L'objectif est d'étudier l'influence de la valeur de K , de la valeur de k_1 ainsi que des concentrations initiales sur le sens d'évolution et sur l'état final.

- Résoudre le système d'équations différentielles dans le cas $[A](0) = [B](0)$ et $[C](0) = [D](0) = 0$
- Préciser le sens d'évolution de la réaction ainsi que la valeur de l'avancement volumique final.
- Tracer l'évolution des concentrations $[A](t)$, $[B](t)$, $[C](t)$ et $[D](t)$ pour différentes valeurs de K .
- Tracer l'évolution des concentrations $[A](t)$, $[B](t)$, $[C](t)$ et $[D](t)$ pour différentes valeurs de k_1 , la valeur de K étant maintenue constante.
- Tracer l'évolution des concentrations $[A](t)$, $[B](t)$, $[C](t)$ et $[D](t)$ pour différentes conditions initiales.

3 Courbes et commentaires



A l'oral :

- vidéo projeter et commenter succinctement les démonstrations demandées.
- vidéo projeter le programme Python et commenter la méthode de résolution.
- étudier l'influence de la valeur de K sur le sens d'évolution et sur l'état final.
- étudier de même l'influence de la valeur de k_1 .
- étudier de même l'influence des conditions initiales.

X Filtre passe-bande d'ordre 2

1 Position du problème

On s'intéresse à un filtre passe-bande d'ordre 2 réalisé à l'aide d'un circuit RLC. On cherche à simuler la réponse à une tension sinusoïdale. On rappelle que la tension d'entrée et la tension de

sortie sont reliées par l'équation différentielle : $LC \frac{d^2s}{dt^2} + RC \frac{ds}{dt} + s = RC \frac{de}{dt}$

La continuité du courant dans la bobine et de la tension aux bornes du condensateur imposent respectivement : $s(0^+) = 0$ et $\frac{ds}{dt}(0^+) = \frac{R}{L}[e(0) - u_c(0)]$

On supposera le condensateur initialement chargé : $u_c(0^+) \neq 0$. Dans un premier temps, on pourra prendre $R = 1 \text{ k}\Omega$, $L = 10 \text{ mH}$ et $C = 100 \text{ nF}$. La tension d'entrée peut être écrite sous la forme :

$$e(t) = e_0 \cos(2\pi ft)$$

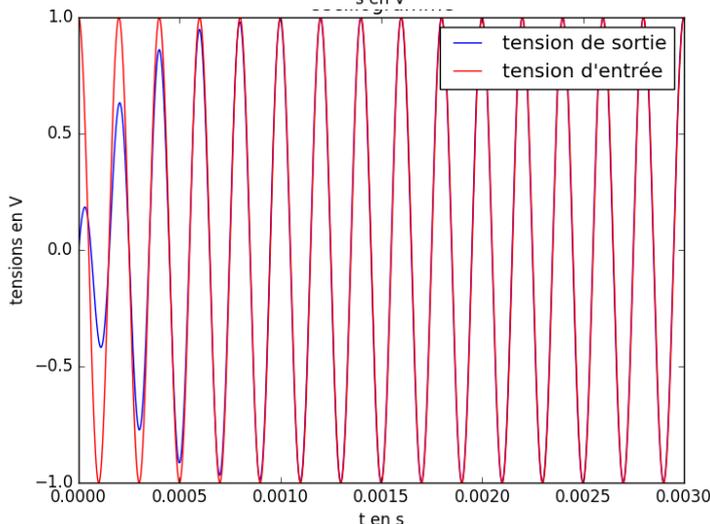
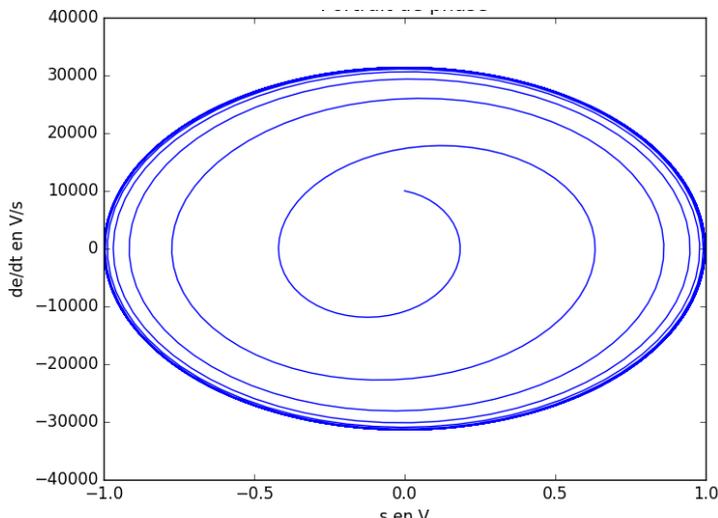
avec par exemple $f = 5 \text{ kHz}$ et $e_0 = 1 \text{ V}$. On n'aura pas intérêt à calculer numériquement la dérivée de

$e(t)$, mais à utiliser l'expression explicite : $\frac{de}{dt}(t) = -2\pi f e_0 \sin(2\pi ft)$

2 Résolution numérique avec Python

- Résoudre l'équation différentielle avec Python.
- Etudier l'effet du régime transitoire : que ce passe-t-il en fonction de $u_c(0)$? Observe-t-on ce régime transitoire en TP d'électronique ? Voir en particulier ce qui se passe si R est faible.
- En régime établi, étudier le comportement du filtre en fonction de la fréquence du signal d'entrée.

3 Courbes et commentaires



A l'oral :

- Présenter le principe de la démonstration de l'équation différentielle proposée.
- vidéo projeter le programme Python et commenter la méthode de résolution.
- Commenter les courbes obtenues en répondant entre autre aux questions du paragraphe 2.

XI Attracteur de Lorenz

1 Position du problème

En 1963, le météorologue Edward Lorenz cherchait à modéliser certains phénomènes météorologiques. Ne pouvant résoudre analytiquement les équations (non-linéaires) de la mécanique des fluides, et ne disposant que d'un ordinateur de puissance limitée, il entreprit de simplifier fortement ses équations pour décrire un phénomène de convection (instabilité de Rayleigh-Bénard). Il aboutit au système d'équations différentielles (adimensionnées) suivant :

$$\begin{cases} \frac{dx}{dt}(t) = \sigma[y(t) - x(t)] \\ \frac{dy}{dt}(t) = \rho x(t) - y(t) - x(t)z(t) \\ \frac{dz}{dt}(t) = x(t)y(t) - \beta z(t) \end{cases}$$

Pour certaines valeurs de σ , ρ et β , le système comporte un attracteur étrange : pour des conditions initiales (presque) quelconques, il évolue vers un ensemble appelé « attracteur de Lorenz ».

2 Résolution numérique avec Python

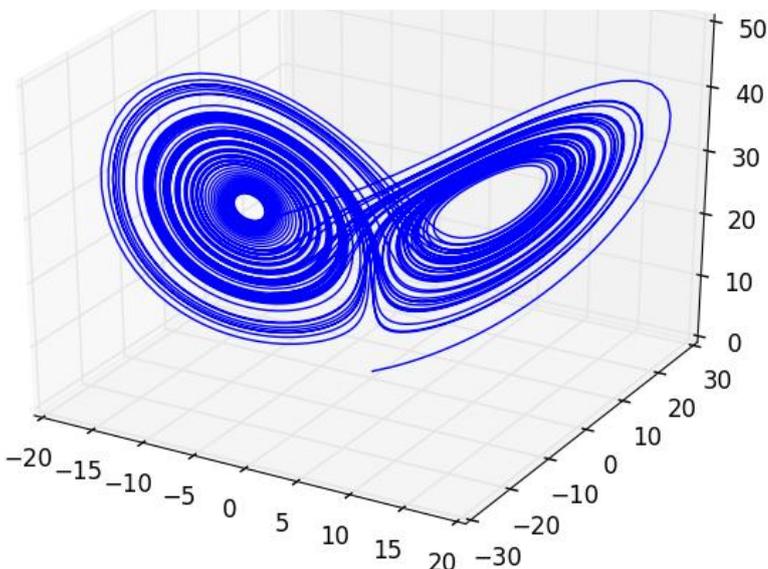
- Pour les simulations, on pourra prendre dans un premier temps $\sigma = 10$, $\rho = 28$ et $\beta = 8/3$, faire varier t entre 0 et 80 et choisir comme condition initiale $(0,1 ; 0,1 ; 0,1)$.
- Résoudre numériquement le système d'équations différentielles. Tracer (en 3 dimensions) la « trajectoire » $(x(t), y(t), z(t))$. Vérifier que le système évolue vers l'attracteur pour des conditions initiales variées.
- Diminuer la valeur de ρ ; comment le système évolue-t-il ?
- Question subsidiaire : supprimer les termes non-linéaires du système différentiel et étudier le comportement du système (on pourra faire varier t entre 0 et 0,1). Quel est l'effet de la non-linéarité ?

Pour tracer des courbes en 3D :

```
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import Axes3D
#
fig=plt.figure()
ax=fig.gca(projection='3d')
ax.plot(X,Y,Z)
plt.show()
```

où X, Y et Z sont les listes (ou des tableaux *numpy* contenant les coordonnées des différents points.

3 Courbes et commentaires



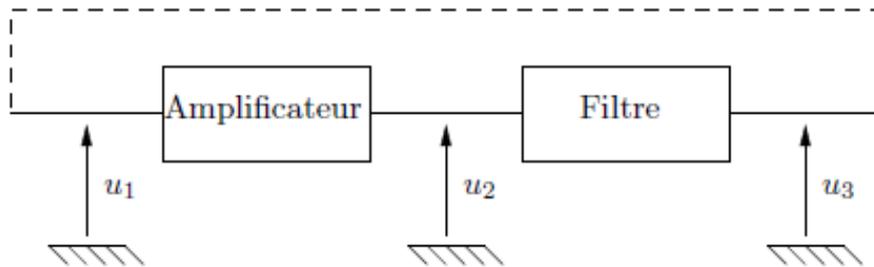
A l'oral :

- Présenter succinctement le système d'équations différentielles.
- vidéo projeter le programme Python et commenter la méthode de résolution.
- Commenter les courbes obtenues en répondant entre autre aux questions du paragraphe 2.

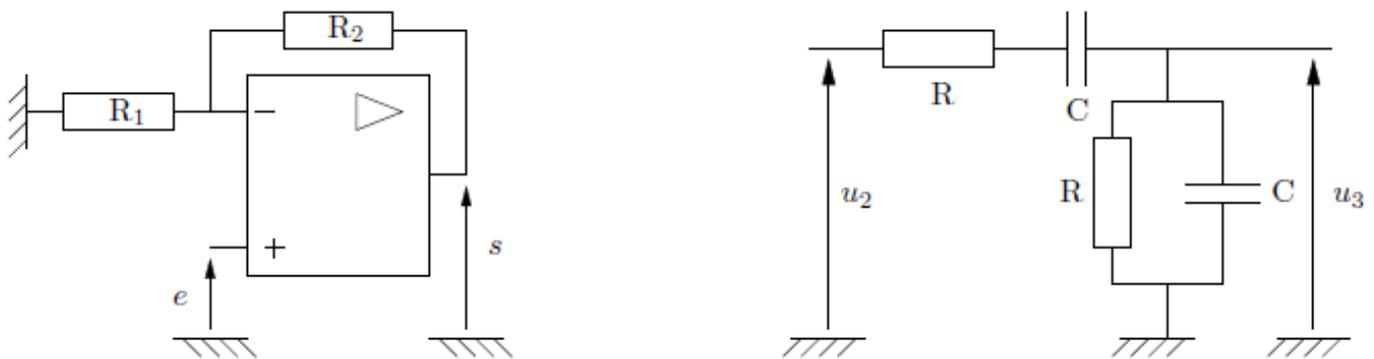
XII Oscillateur de Wien

1 Position du problème

L'oscillateur de Wien est un montage électronique constitué d'un filtre passe-bande (appelé filtre de Wien) et d'un montage amplificateur réalisé avec un AO/ALI :



On donne la structure détaillée du filtre de Wien et du montage amplificateur :



La particularité de ce montage est que la moindre fluctuation de tension dans ce montage peut y faire naître des oscillations (sous réserve que les paramètres du montage soient bien choisis). Le filtre est

caractérisé par l'équation différentielle : $(RC)^2 \frac{d^2 u_3}{dt^2} + 3RC \frac{du_3}{dt} + u_3 = RC \frac{du_2}{dt}$

et le montage amplificateur par la relation : $u_2 = Gu_1$ avec $G = 1 + \frac{R_2}{R_1}$

Quand le montage est bouclé, les différentes tensions vérifient l'équation différentielle :

$$(RC)^2 \frac{d^2 u}{dt^2} + (3 - G)RC \frac{du}{dt} + u = 0$$

2 Résolution numérique avec Python

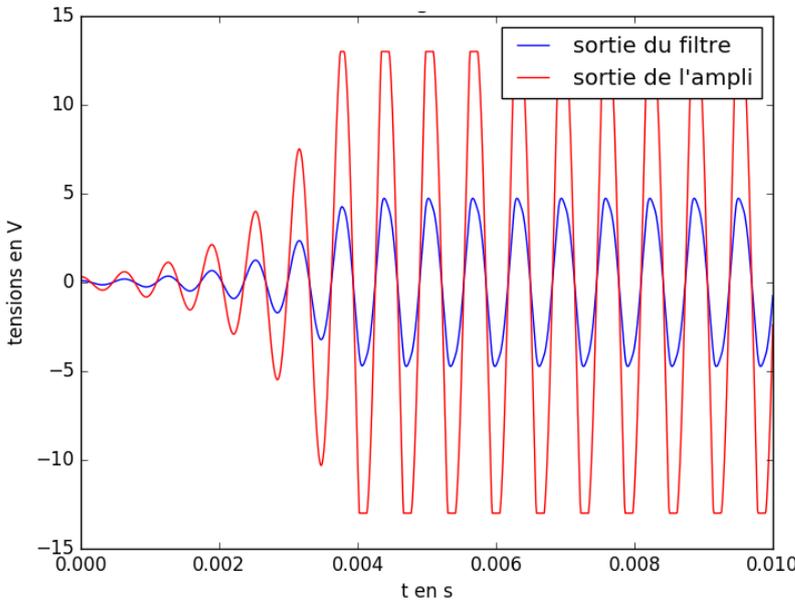
On cherche à simuler le comportement de ce montage.

- Programmer la résolution numérique de cette équation dans le cas où $G < 3$, $G = 3$ et $G > 3$. On pourra prendre les conditions initiales $u(0) = 10 \text{ mV}$ et $u'(0) = 0$ ainsi que les valeurs $R = 10 \text{ k}\Omega$ et $C = 10 \text{ nF}$.
- En fait, le montage amplificateur ne peut pas produire un signal de sortie infiniment grand ; la tension de sortie est limitée par une valeur appelée tension de saturation V_s

$$u_2 = \begin{cases} -V_s & \text{si } Gu_1 \leq -V_s \\ Gu_1 & \text{si } |Gu_1| \leq V_s \\ V_s & \text{si } Gu_1 \geq V_s \end{cases}$$

- Modifier le programme précédent pour simuler le comportement du montage. Tracer les courbes des tensions ainsi que le portrait de phase.

3 Courbes et commentaires



A l'oral :

- Présenter le principe de la démonstration de la première équation différentielle proposée.
- vidéo projeter le programme Python et commenter la méthode de résolution.
- Commenter les courbes obtenues en répondant entre autre aux questions du paragraphe 2.

XIII Etude d'un filtre passe-bas du premier ordre

1 Position du problème

On s'intéresse à un filtre passe-bas d'ordre 1 réalisé à l'aide d'un circuit RC. On cherche à simuler la réponse à une tension sinusoïdale. On rappelle que la tension d'entrée et la tension de sortie sont reliées par l'équation différentielle :

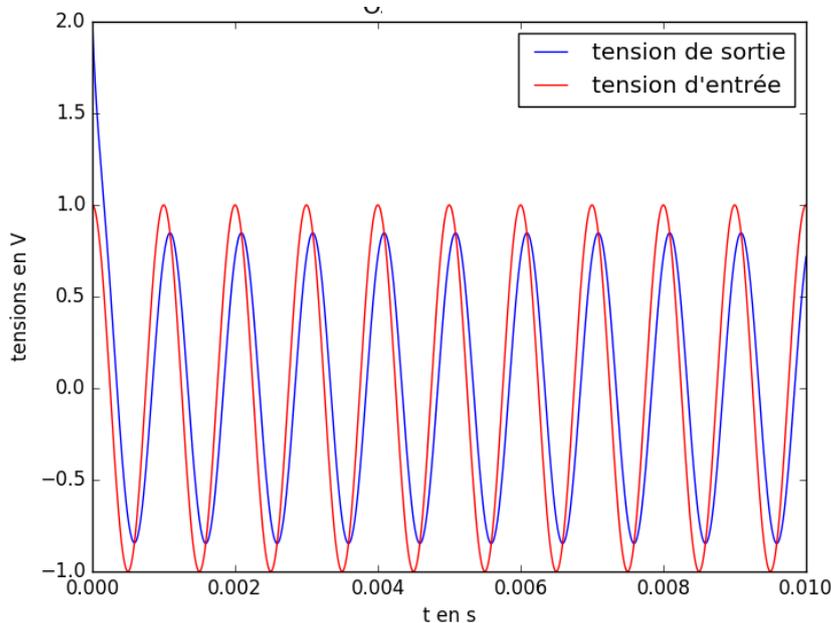
$$RC \frac{ds}{dt} + s = e$$

On supposera le condensateur initialement chargé : $s(0^+) \neq 0$. Dans un premier temps, on pourra prendre $R = 1 \text{ k}\Omega$ et $C = 100 \text{ nF}$. La tension d'entrée peut être écrite sous la forme : $e(t) = e_0 \cos(2\pi ft)$ avec par exemple $f = 1 \text{ kHz}$ et $e_0 = 1 \text{ V}$.

2 Résolution numérique avec Python

- Résoudre l'équation différentielle avec Python.
- Etudier l'effet du régime transitoire : que ce passe-t-il en fonction de $s(0)$? Observe-t-on ce régime transitoire en TP d'électronique ?
- En régime établi, étudier le comportement du filtre en fonction de la fréquence d'entrée.

3 Courbes et commentaires



A l'oral :

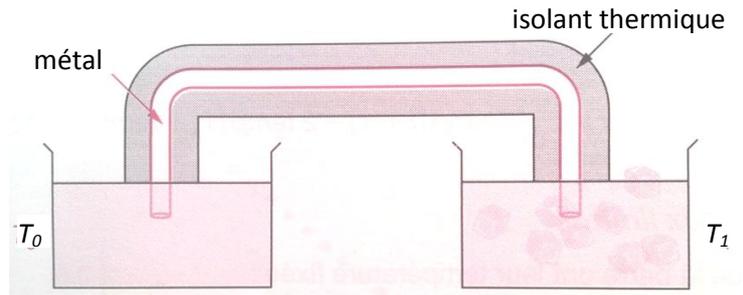
- Démontrer l'équation différentielle proposée.
- vidéo projeter le programme Python et commenter la méthode de résolution.
- Commenter les courbes obtenues en répondant entre autre aux questions du paragraphe 2.

XIV Propagation de la chaleur

1 Position du problème

On considère une barre métallique homogène de longueur L et de section S , dont les parois latérales sont calorifugées.

Ses extrémités sont plongées dans deux bains liquides, dont les températures sont maintenues à des valeurs fixées T_0 et T_1 (bain thermostaté pour ce qui concerne T_0 et mélange eau-glace pour ce qui concerne T_1).



La barre est initialement à la température T_1 ; on souhaite simuler l'évolution au cours du temps du profil de température $T(x,t)$.

La température $T(x,t)$ vérifie l'équation différentielle $\frac{\partial^2 T}{\partial x^2} - \frac{1}{\kappa} \frac{\partial T}{\partial t} = 0$ avec κ une constante appelée diffusivité thermique (vu en spé).

On introduira la grandeur adimensionnée $\theta(x,t) = \frac{T(x,t) - T_1}{T_0 - T_1}$ qui mesure l'écart de température ramené à l'écart maximal et qui vérifie la même équation différentielle que $T(x,t)$.

Déterminer $\theta(x,0)$ ainsi que les conditions aux limites $\theta(0,t)$ et $\theta(L,t)$ aux extrémités.

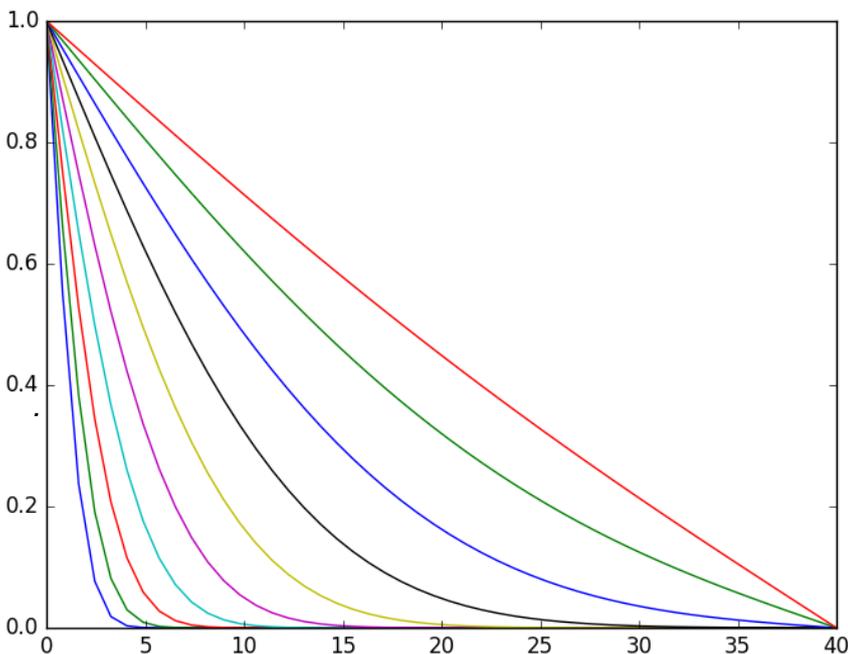
2 Résolution numérique avec Python

Après avoir effectué une discrétisation spatiale de la tige ainsi qu'une discrétisation temporelle, montrer

$$\theta(x_i, t + \Delta t) = \theta(x_i, t) + \Delta t \times \frac{\theta(x_{i+1}, t) - 2\theta(x_i, t) + \theta(x_{i-1}, t))}{\Delta x^2}$$

Déterminer les valeurs de $\theta(x_i, t_j)$ en différents points, à différents instants puis tracer les profils de température $\theta = f(x)$ à différentes dates t .

3 Courbes et commentaires



A l'oral :

- commenter les profils de température obtenus (conditions aux limites et évolution au cours du temps).
- interpréter la courbe obtenue en régime stationnaire.
- vidéo projeter le programme Python et commenter la méthode de résolution

Travail à faire et à présenter à l'oral le (précisé ultérieurement)

- Vous allez former ... groupes de 3 ou 4 élèves.
- Votre travail consiste à résoudre des équations différentielles, à tracer des courbes avec Python puis à faire une présentation à l'oral (sur beamer). Ce travail se fera en binôme en informatique mais par groupe de 3 ou 4 pour la présentation à l'oral.
- Précision : vous devez résoudre 2 des 14 cas proposés précédemment (un cas imposé parmi 12 et un cas à choisir parmi les 14 cas proposés), puis compléter le tableau ci-dessous (affiché dans la salle de cours) en indiquant le numéro du deuxième cas traité.
- Vous devrez ensuite préparer un beamer pour le premier cas proposé et vous le présenterez à l'oral à une date précisée ultérieurement.

groupe	Cas étudiés	Noms	
1	I et ...	-	-
2	II et ...	-	-
3	III et ...	-	-
4	IV et ...	-	-
5	V et ...	-	-
6	VI et ...	-	-
7	VII et ...	-	-
8	VIII et ...	-	-
9	IX et ...	-	-
10	X et ...	-	-
11	XI et ...	-	-
12	XII et ...	-	-