

# Méthode d'Euler pour les équations différentielles du second ordre

Ce deuxième TP de calcul numérique a pour objectif d'appliquer la méthode d'Euler à des équations différentielles du second ordre. L'oscillateur de Wien constituera le premier exemple, et le pendule simple sera le second.

## I - Oscillateur quasi-sinusoïdal à pont de Wien

### I.A - Rappels du cours de physique

- Oscillateur de Wien

Considérons l'oscillateur de Wien de la figure 1, constitué d'un bloc amplificateur non-inverseur à ALI de gain  $A$  et d'un filtre passe-bande de fonction de transfert

$$H = \frac{H_0}{1 + jQ \left( \frac{\omega}{\omega_0} - \frac{\omega_0}{\omega} \right)}.$$

Dans tout ce TP, on travaillera avec les formes canoniques sans chercher à les relier aux valeurs des composants.

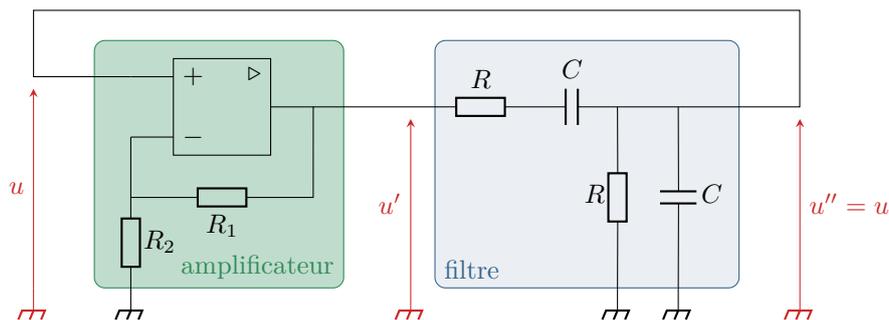


Figure 1 – Oscillateur de Wien.

Nous avons montré dans le cours sur les oscillateurs que :

▷ si  $|u| < V_{\text{sat}}/A$ , alors l'ALI fonctionne en régime linéaire et  $u_1$  vérifie l'équation différentielle

$$\frac{d^2u}{dt^2} + \frac{\omega_0}{Q}(1 - H_0A)\frac{du}{dt} + \omega_0^2u = 0,$$

▷ si  $|u| > V_{\text{sat}}/A$ , alors l'ALI fonctionne en régime de saturation et  $u$  vérifie l'équation différentielle

$$\frac{d^2u}{dt^2} + \frac{\omega_0}{Q}\frac{du}{dt} + \omega_0^2u = 0.$$

Le changement d'équation différentielle lorsque l'ALI change de régime (deux fois par période) rend la résolution analytique franchement fastidieuse ... mais une résolution numérique est parfaitement adaptée !

- Schéma d'Euler appliqué à une équation différentielle d'ordre 2

La méthode (à connaître!) consiste à traiter la dérivée première comme une deuxième fonction inconnue et « indépendante » de  $u$ . Ce faisant, l'équation différentielle d'ordre 2 devient un système différentiel de deux équations couplées du premier ordre. Par exemple, l'équation du régime linéaire devient

$$\begin{cases} \frac{du}{dt} = v \\ \frac{dv}{dt} + \frac{\omega_0}{Q}(1 - H_0A)v + \omega_0^2u = 0 \end{cases}$$

## I.B - Variables globales

Dans toute cette partie, les caractéristiques du filtre ( $H_0$ ,  $Q$  et  $\omega_0$ ) seront considérées comme des variables globales. Seul le gain de l'amplificateur sera amené à être modifié. De même, tous les calculs s'effectueront sur une même durée et un même nombre de pas de temps, que l'on définira au début du programme comme des variables globales.

```

1 | Vsat = 15      # V
3 | H0 = 1/3
4 | Q = 1/3
5 | w0 = 1e4      # rad.s-1
6 | T0 = 2*np.pi/w0      # s
8 | Nper = 50      # nbres de périodes propres
9 | Npts = 1000    # nombre de pts par période
10 | dt = T0/Npts
11 | Ntot = Nper * Npts
13 | t = np.linspace(0, Nper*T0, Ntot)

```

1 - Expliquer à quoi correspondent les variables  $T_0$ ,  $N_{per}$ ,  $N_{pts}$ ,  $dt$ ,  $N_{tot}$  et  $t$ .

## I.C - Résolution numérique

2 - Écrire pour les deux états de l'ALI le système d'équations différentielles équivalent, et les discrétiser par application du schéma d'Euler explicite.

3 - Coder une fonction `oscillateur` permettant de calculer la tension  $u$  en fonction de la valeur du gain  $A$ . Les oscillations démarrant sur le bruit de fond électronique, on prendra comme conditions initiales

$$u(0) = 5 \text{ mV} \quad \text{et} \quad \frac{du}{dt}(0) = 0.$$

4 - Afficher sur une figure la tension  $u$  calculée numériquement, ainsi que deux droites horizontales indiquant les valeurs seuils pour lesquels l'ALI passe en saturation (utiliser `plt.axhline`, cf. fiche outil Python).

5 - Observer l'allure de la courbe obtenue pour différentes valeurs du gain  $A$  (p.ex. 2,5, 2,95, 3,05, 3,5, 6 et 12) :  
 ▷ mettre en évidence une valeur seuil  $A^* = 3$  : que se passe-t-il pour  $A \leq A^*$  ? pour  $A \geq A^*$  ? ;  
 ▷ commenter la durée de démarrage des oscillations et la durée des phases non-linéaires une fois les oscillations établies.

## I.D - Influence du gain sur la période des oscillations

Des tracés issus des simulations précédentes montrent facilement que la période des oscillations n'est pas toujours égale à la période propre du filtre, et s'en éloigne beaucoup pour les grandes valeurs du gain. On cherche désormais à étudier cette dépendance. Une première étape évidente consiste à déterminer la période des oscillations à partir du résultat des calculs précédents. Pour ce faire, il est astucieux de repérer que les oscillations sont symétriques et de moyenne nulle, si bien que deux annulations successives sont séparées d'une demi-période.

6 - Étudier le signe du produit  $u[n] * u[n+1]$  en fonction des valeurs prises par  $u$  aux instants  $t[n]$  et  $t[n+1]$ . En déduire une façon de repérer une annulation de  $u$ .

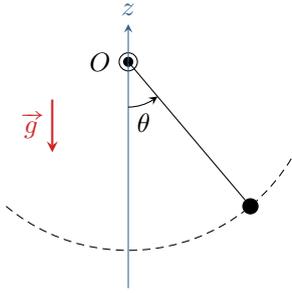
7 - Écrire une fonction `periode` prenant en argument une tension  $u$  et renvoyant sa période. On rappelle que le temps est défini ici comme une variable globale. 🚫🚫🚫 **Attention !** La période n'est définie qu'en régime établi ... et compte tenu des phases de démarrage des oscillations, la liste contenant les valeurs de  $u$  devra donc être parcourue **en partant de la fin**, ce qui se fait en utilisant des indices négatifs. Par ailleurs, inutile de parcourir toute la liste : il suffit d'identifier deux annulations successives.

8 - En utilisant les fonctions précédentes, représenter la période des oscillations en fonction du gain pour  $A$  allant de 3,05 à 50. On pourra commencer par définir une liste (ou un tableau `numpy`) de valeurs de gain pour lesquelles on calculera la période.

*En fonction des valeurs utilisées, on pourra constater des « fluctuations » de la période d'oscillations autour d'une tendance moyenne. Ces fluctuations sont dues aux imprécisions numériques du schéma d'Euler explicite, qui s'accumulent et peuvent devenir notables pour les grandes valeurs de  $n$ . Diminuer le pas de temps  $dt$  permet bien de constater que ces fluctuations sont d'origine numérique et non pas physique ... mais cela augmente notablement le temps de calcul allant jusqu'à l'affichage de la courbe !*

## II - Pendule simple

### II.A - Rappels du cours de physique



Un pendule simple est constitué d'une masse  $m$  suspendue à un fil idéal (masse nulle, inextensible, etc.) de longueur  $\ell$ . Cette masse oscille en formant un angle  $\theta(t)$  avec la verticale. Par application du théorème de la résultante cinétique, on montre que  $\theta$  vérifie l'équation différentielle

$$\frac{d^2\theta}{dt^2} + \Omega_0^2 \sin \theta = 0 \quad \text{avec} \quad \Omega_0 = \sqrt{\frac{g}{\ell}}.$$

Pour simplifier, on prendra par la suite  $\Omega_0 = 1 \text{ rad} \cdot \text{s}^{-1}$ , soit une période d'environ 6,3s, ce qui est tout à fait pertinent sur le plan expérimental. On supposera également le pendule lâché sans vitesse initiale d'un angle de  $45^\circ$ .

### II.B - Résolution numérique

**9** - Écrire les relations de récurrence issues du schéma d'Euler explicite appliqué à cette équation différentielle. On notera  $\omega$  la fonction dérivée.

**10** - Analyser l'équation différentielle pour proposer une valeur de pas de temps raisonnable a priori.

**11** - L'implémenter en Python pour résoudre l'équation sur une durée totale correspondant à trois oscillations. On supposera le pendule lâché sans vitesse initiale depuis un angle de  $45^\circ$ .

### II.C - Euler détail !

**12** - Vérifier en augmentant la durée totale de calcul que la solution numérique finit toujours par diverger, et ce quel que soit le pas de temps.

L'objectif de cette partie est d'en proposer une interprétation.

**13** - Du point de vue physique, identifier ce que représente la quantité

$$E = m\ell^2 \left( \Omega_0^2 (1 - \cos \theta) + \frac{\dot{\theta}^2}{2} \right).$$

Comment est-elle censée se comporter pour un pendule simple ?

**14** - En utilisant les calculs précédents issus de la méthode d'Euler, représenter  $E/m\ell^2$  en fonction du temps. Proposer une interprétation à la divergence constatée.

### II.D - Amélioration par l'utilisation d'une bibliothèque

Ce problème est intrinsèque au schéma d'Euler explicite, et ne peut être résolu qu'en passant à un schéma numérique plus élaboré, c'est-à-dire en améliorant le « protocole » permettant de transformer l'équation différentielle en relation de récurrence. La fonction `odeint`, de la bibliothèque `scipy`, permet d'utiliser un schéma plus performant.

**15** - (Re)lire le paragraphe IV de la fiche outil « Python pour la physique ».

Dans notre cas, il ne s'agit pas d'une unique équation différentielle mais d'un système de deux équations différentielles, que l'on peut réécrire sous forme vectorielle :

$$\frac{d\vec{u}}{dt} = \vec{f}(\vec{u}) \quad \text{avec} \quad \vec{u} = \begin{bmatrix} \theta \\ \omega = \frac{d\theta}{dt} \end{bmatrix}$$

**16** - Écrire le système différentiel sous la forme ci-dessus, et en déduire les deux composantes de la fonction  $\vec{f}$ .

**17** - Écrire la fonction Python  $f(u, t)$ , renvoyant une liste de deux éléments correspondant aux deux composantes de  $\vec{f}$ . Attention, même si la fonction  $\vec{f}$  ne dépend pas du temps dans notre cas,  $t$  doit obligatoirement apparaître dans les arguments de  $f$ .

**18** - En utilisant `odeint`, déterminer l'évolution de  $\theta(t)$  et comparer au résultat issu de la méthode d'Euler.

- ▷ La condition initiale doit être donnée sous forme d'une liste à deux éléments, contenant les deux composantes du vecteur  $\vec{u}(t=0)$ .
- ▷ La fonction renvoie un tableau `numpy` dont les éléments sont les deux composantes du vecteur  $\vec{u}$  au cours du temps. À vous de chercher, par exemple en faisant des essais dans la console, comment isoler  $\theta$ .