

M14. Ondes acoustiques

Préparation à l'Agrégation de Physique

Introduction et Objectifs

Ce travail expérimental propose de transformer un **résonateur mécanique passif** (le diapason) en un **capteur de masse**.

L'idée centrale repose sur le fait que la fréquence propre d'un oscillateur dépend de son inertie : toute modification de masse induit donc une variation mesurable de fréquence.

Objectifs :

- Identifier les paramètres caractéristiques du diapason (f_0, Q)
 - Valider le modèle d'oscillateur harmonique équivalent
 - Établir une loi d'étalonnage reliant masse ajoutée et fréquence
 - Évaluer les limites de précision du capteur
-

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from scipy.optimize import curve_fit
from numpy.fft import fft, ifft
```

1. Modélisation Théorique : Du Milieu Continu à l'Oscillateur

1.1 Approche d'Euler-Bernoulli

Le diapason est initialement un système continu dont la flexion des branches est régie par l'équation

$$EI \frac{\partial^4 y}{\partial x^4} + \rho A \frac{\partial^2 y}{\partial t^2} = 0$$

1.2 Formalisme de Lagrange

Pour le premier mode de vibration, on utilise la décomposition modale $y(x, t) = \phi(x)q(t)$.

Le Lagrangien du système $\mathcal{L} = T - V$ s'écrit :

- **Énergie Cinétique** (T) : $T = \frac{1}{2} m^* \dot{q}^2$ où $m^* = \int_0^L \rho A \phi(x)^2 dx$
- **Énergie Potentielle** (V) : $V = \frac{1}{2} k q^2$ où $k = \int_0^L EI (\phi''(x))^2 dx$

L'application de l'équation d'Euler-Lagrange $\frac{d}{dt} \left(\frac{\partial \mathcal{L}}{\partial \dot{q}} \right) - \frac{\partial \mathcal{L}}{\partial q} = 0$ conduit à l'équation de l'oscillateur harmonique :

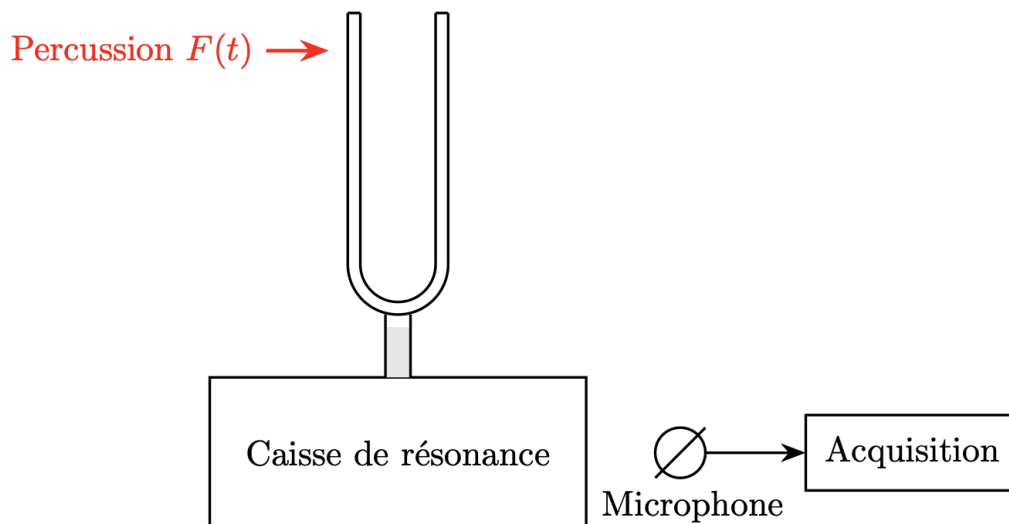
$$m^* \ddot{q} + kq = 0 \implies f_0 = \frac{1}{2\pi} \sqrt{\frac{k}{m^*}}$$

où m^* est la masse équivalente et k la raideur effective.

2. Caractérisation et Étalonnage

2.1 Dispositif Expérimental

Le diapason est excité par une impulsion mécanique (petit coup de marteau). Le signal acoustique est capté par un microphone, récupéré sur un oscilloscope.



La suite du traitement se fait sous Python. Un fit Lorentzien sera alors appliqué sur le pic pour extraire f_0 et Q .

Evolution temporelle

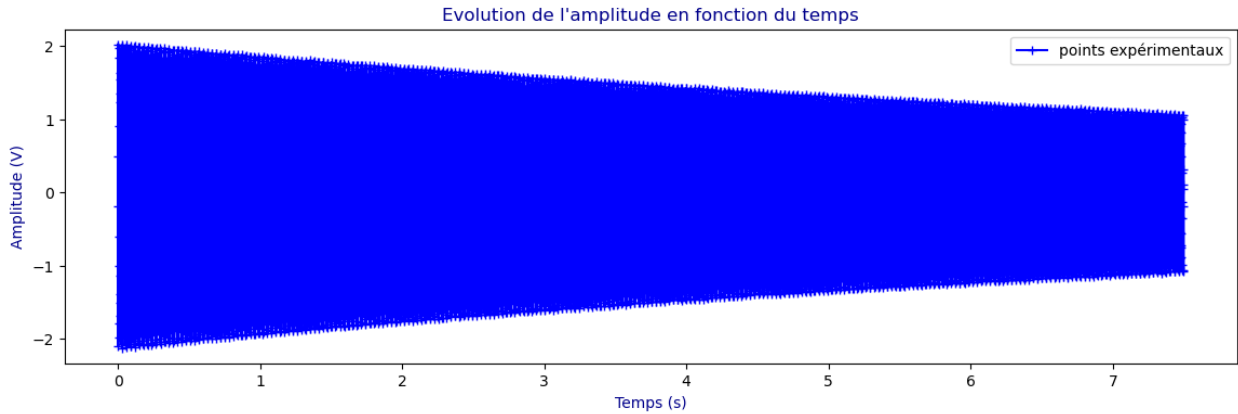
```
### acquisition du signal

data = pd.read_csv("Diapason_seul.csv", delimiter = ";", decimal = ",")
data

V = data["EA1"].values
t = data["Temps"].values
V = V[0:7500]
t = t[0:7500]

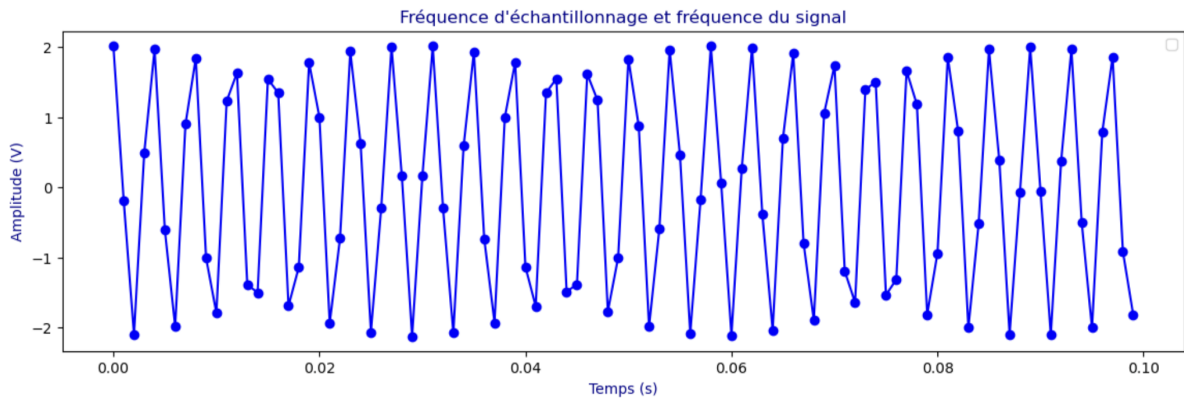
plt.figure(0, figsize=(14,4))

### Signal acquit
plt.plot(t, V, 'b+-', label="points expérimentaux")
plt.xlabel("Temps (s)", color="darkblue")
plt.ylabel("Amplitude (V)", color="darkblue")
plt.title("Evolution de l'amplitude en fonction du temps", color="darkblue")
plt.legend()
```



2.2 Conditions de Shannon

Effectuons un "zoom" sur le signal perçu.



La période d'échantillonnage est de $T_e = 1 \text{ ms}$, soit une fréquence d'échantillonnage de :

$$f_e = \frac{1}{T_e} = 1000 \text{ Hz.}$$

Le diapason émet un signal sonore de fréquence voisine de 256 Hz (cette valeur sera vérifiée expérimentalement par la suite).

D'après le théorème de Shannon, la fréquence d'échantillonnage doit être au moins deux fois supérieure à la fréquence maximale du signal :

$$f_e \geq 2f_{\text{signal}}.$$

Ici, la condition devient :

$$f_e \geq 2 \times 256 = 512 \text{ Hz.}$$

Or, on a $f_e = 1000 \text{ Hz} > 512 \text{ Hz}$. Le critère de Shannon est donc respecté, ce qui garantit une acquisition sans repliement de spectre.

Interprétation physique :

- L'amplitude décroît lentement : l'amortissement est faible

- Le régime observé est celui d'un **oscillateur pseudo-périodique**
- La lente décroissance indique un **facteur de qualité élevé**

Conclusion : Le diapason constitue un **excellent résonateur**, adapté à des mesures précises en fréquence.

Transformée de Fourier

```

### Encadrement de la fréquence de résonance :
fen_f = (f>250) & (f<266)
f1 = f[fen_f]
V21 = V2[fen_f]

Vmax = max(V21)
Vnorm = V21/Vmax

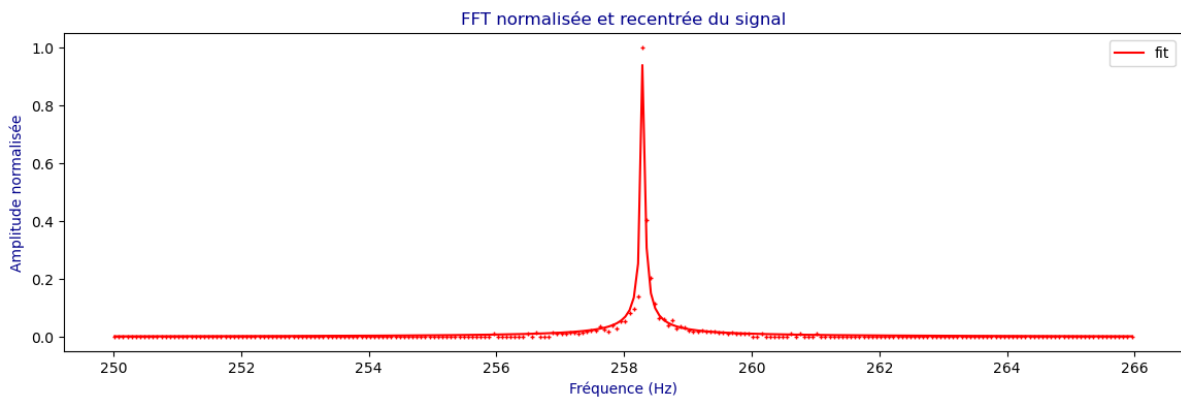
def f_lor(x, fres, Q):
    return 1./(np.sqrt(1+(Q**2)*((x/fres)-(fres/x)**2)))

### Il faut renseigner TRES PRECISEMENT les valeurs pour aider le fit.

pop, pcov = curve_fit(f_lor, f1, Vnorm, p0=[258.3,8000],
    ↪ bounds=((250,259),(7500,8500))) #p0 donne un valeur proche et bounds des
    ↪ bornes
aop, bop = pop

plt.figure(0, figsize=(14,4))
plt.plot(f1, f_lor(f1, aop, bop), 'r-', label='fit')
plt.plot(f1, Vnorm, 'r+', ms=3)
plt.title("FFT normalisée et recentrée du signal", color="darkblue")
plt.xlabel("Fréquence (Hz)", color="darkblue")
plt.ylabel("Amplitude normalisée", color="darkblue")
plt.legend()

```



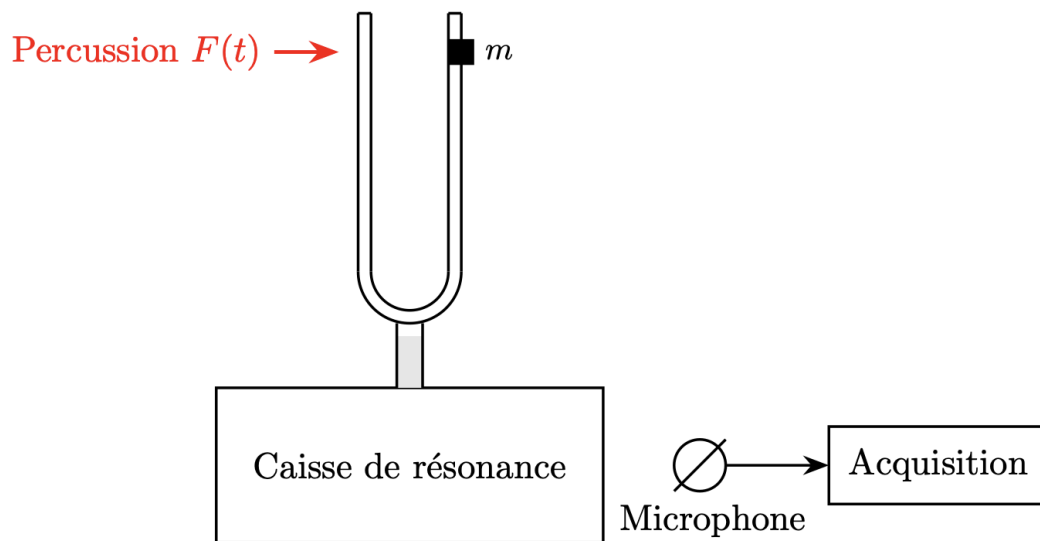
La fréquence de résonance à vide est : 258.3 Hz

Le facteur de qualité est : 6679.0

- Le pic est **très étroit** \Rightarrow grande sélectivité fréquentielle
- Un grand Q signifie :
 - faible dissipation d'énergie
 - grande précision sur la mesure de fréquence

Étalonnage du capteur de masse

On réalise la même procédure mais cette fois en ajoutant une masse pour perturber le diapason.



L'ajout d'une masse peut être réalisé simplement à l'aide d'un petit aimant. Pour garantir la **répétabilité** de l'expérience, il est crucial de positionner cette masse à une hauteur constante sur la branche du diapason ; l'utilisation d'un repère au feutre est, à ce titre, vivement recommandée.

D'un point de vue physique, la présence de l'aimant induit :

- Une **augmentation de l'inertie** du système vibrant ;
- Une **modification (diminution) de la fréquence propre** du résonateur.

2.3 Modélisation avec masse additionnelle

L'ajout d'une masselotte de masse m à l'extrémité de la branche ($x = L$) modifie l'énergie cinétique du système sans affecter sa raideur k . En supposant le mode normalisé tel que $\phi(L) = 1$, la nouvelle énergie cinétique s'écrit :

$$T = \frac{1}{2}(m^* + m)\dot{q}^2$$

Le nouveau Lagrangien conduit à l'expression de la fréquence de résonance perturbée :

$$f_{\text{reso}} = \frac{1}{2\pi} \sqrt{\frac{k}{m^* + m}}$$

En élevant cette expression à la puissance -2 , on isole la dépendance en m :

$$f_{\text{reso}}^{-2} = \frac{4\pi^2(m^* + m)}{k}$$

On aboutit finalement à la loi d'évolution affine :

$$f_{\text{reso}}^{-2} = \left(\frac{4\pi^2}{k} \right) m + \frac{4\pi^2 m^*}{k}$$

Cette relation montre que la pente de la droite expérimentale est inversement proportionnelle à la raideur k , tandis que l'ordonnée à l'origine permet d'identifier la masse modale m^* .

La chaîne de traitement numérique, bien que semblant dense, s'articule autour de trois étapes fondamentales de l'analyse spectrale :

- Les données expérimentales brutes sont extraites et affectées à des variables.
- Une fenêtre d'étude est définie autour de la zone de résonance. Cette étape est cruciale pour faciliter la convergence de l'ajustement *lorentzien* en minimisant l'influence du bruit de fond et des fréquences parasites.
- Le traitement se conclut par l'application du modèle de fit et la génération de figures. Ces tracés permettent de valider la superposition du modèle théorique avec les données expérimentales et d'extraire les grandeurs caractéristiques.

```

### acquisition du signal
m0 = 0 #masse à vide
m1 = 1.0 #masse de la masse 1
m2 = 2.9 #masse de la masse 2
m3 = 3.9 #masse de la masse 3

### Import de toutes les données :

#masse 1
dataM1 = pd.read_csv("Diapason_masse1.csv", delimiter = ";", decimal = ",")
dataM1
VM1 = dataM1["EA1"].values
tM1 = dataM1["Temps"].values
VM1 = VM1[0:7550]
tM1 = tM1[0:7550]
fM1 = dataM1["frequence"].values
V2M1 = dataM1["S_Amplitude"].values
V2M1 = V2M1[0:7550]
fM1 = fM1[0:7550]

#masse 2
dataM2 = pd.read_csv("Diapason_masse2.csv", delimiter = ";", decimal = ",")
dataM2
VM2 = dataM2["EA1"].values
tM2 = dataM2["Temps"].values
VM2 = VM2[0:7550]
tM2 = tM2[0:7550]
fM2 = dataM2["frequence"].values
V2M2 = dataM2["S_Amplitude"].values
V2M2 = V2M2[0:7550]
fM2 = fM2[0:7550]

#masse3
dataM3 = pd.read_csv("Diapason_masse3.csv", delimiter = ";", decimal = ",")
dataM3
VM3 = dataM3["EA1"].values
tM3 = dataM3["Temps"].values
VM3 = VM3[0:7550]
tM3 = tM3[0:7550]
fM3 = dataM3["frequence"].values
V2M3 = dataM3["S_Amplitude"].values
V2M3 = V2M3[0:7550]
fM3 = fM3[0:7550]

```

```
### Fenêtrage de chacune des variables.
```

```
fen_fM1 = (fM1>249) & (fM1<265)
```

```
f1M1 = fM1[fen_fM1]  
V21M1 = V2M1[fen_fM1]
```

```
f1M2 = fM2[fen_fM1]  
V21M2 = V2M2[fen_fM1]
```

```
f1M3 = fM3[fen_fM1]  
V21M3 = V2M3[fen_fM1]
```

```
VmaxM1 = max(V21M1)  
VnormM1 = V21M1/VmaxM1
```

```
VmaxM2 = max(V21M2)  
VnormM2 = V21M2/VmaxM2
```

```
VmaxM3 = max(V21M3)  
VnormM3 = V21M3/VmaxM3
```

```
### Régression lorentzienne de chacune des valeurs
```

```
popM1, pcovM1 = curve_fit(f_lor, f1M1, VnormM1, p0=[256.5,8000],  
↳bounds=((256.3,256.5),(7500,8500)))  
aopM1, bopM1 = popM1
```

```
popM2, pcovM2 = curve_fit(f_lor, f1M2, VnormM2, p0=[252.6,10000],  
↳bounds=((252,254.5),(7500,10500)))  
aopM2, bopM2 = popM2
```

```
popM3, pcovM3 = curve_fit(f_lor, f1M3, VnormM3, p0=[251.5,15000],  
↳bounds=((251.3,252.5),(14500,15500)))  
aopM3, bopM3 = popM3
```

```

plt.figure(0,figsize=(14,7))

plt.plot(f1, f_lor(f1, aop, bop), 'b-')
plt.plot(f1, Vnorm,'b+',label="à vide" ,ms=5)

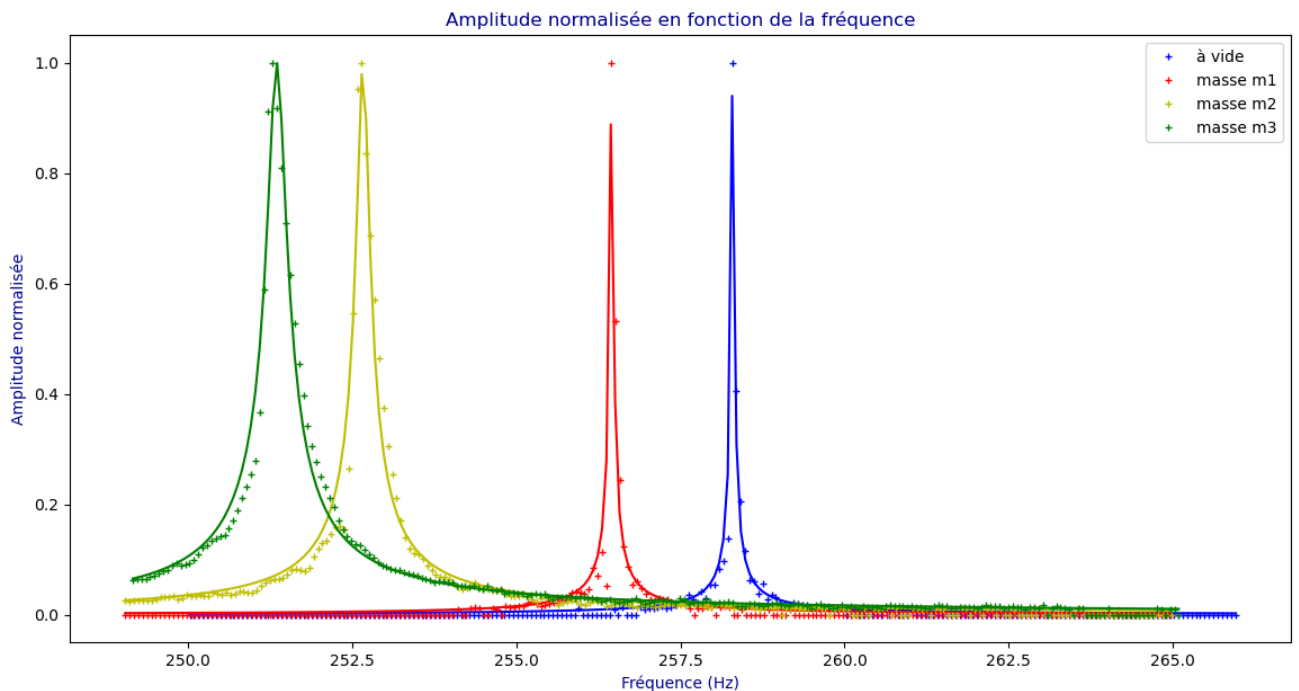
plt.plot(f1M1, f_lor(f1M1, aopM1, bopM1), 'r-')
plt.plot(f1M1, VnormM1,'r+', label="masse m1",ms=5)

plt.plot(f1M2, f_lor(f1M2, aopM2, bopM2), 'y-')
plt.plot(f1M2, VnormM2, 'y+', label="masse m2", ms=5)

plt.plot(f1M3, f_lor(f1M3, aopM3, bopM3), 'g-')
plt.plot(f1M3, VnormM3,'g+', ms=5, label="masse m3")

plt.title("Amplitude normalisée en fonction de la fréquence",
→color="darkblue")
plt.xlabel("Fréquence (Hz)", color="darkblue")
plt.ylabel("Amplitude normalisée", color="darkblue")
plt.legend()

```



- La fréquence **diminue** avec la masse ce qui semble cohérent avec le travail théorique et sera vérifié rapidement :

$$f \propto \frac{1}{\sqrt{m}}$$

- Le facteur de qualité chute fortement (nous discuterons des conséquences de cette chute en fin de montage).

2.4 Extraction des données

```
freso = np.array([aop, aopM1, aopM2, aopM3])
m = np.array([m0, m1, m2, m3])*10**-3
Q = np.array([bop, bopM1, bopM2, bopM3])

print(f"{'Configuration':<15} | {'Masse (g)':<10} | {'f_reso (Hz)':<12} | Q
      ↳{'Q':<10}")
print("-" * 55)

labels = ["À vide", "Masse 1", "Masse 2", "Masse 3"]

for i in range(len(freso)):
    print(f"{labels[i]:<15} | {m[i]:<10.2f} | {freso[i]:<12.2f} | {Q[i]:<10.
          ↳1f}")
```

Configuration	Masse (g)	f_reso (Hz)	Q
À vide	0.00	258.29	6678.9
Masse 1	1.00	256.45	5633.0
Masse 2	2.90	252.66	1316.9
Masse 3	3.90	251.35	870.9

2.4 Vérification du modèle théorique

Afin de vérifier le modèle théorique, nous traçons l'évolution de l'inverse du carré de la fréquence de résonance en fonction de la masse additionnelle m :

$$f_{\text{reso}}^{-2} = g(m)$$

En nous appuyant sur l'expression de la pulsation propre d'un oscillateur harmonique $\omega_0 = \sqrt{\frac{k}{M_{\text{tot}}}}$, nous devons obtenir une relation affine de la forme :

$$f_{\text{reso}}^{-2} = \alpha \cdot m + \beta$$

Soit plus précisément :

$$f_{\text{reso}}^{-2} = \left(\frac{4\pi^2}{k} \right) m + \frac{4\pi^2 m^*}{k}$$

- La pente $\alpha = \frac{4\pi^2}{k}$ permet de déterminer la raideur k du diapason.
- L'ordonnée à l'origine β permet d'estimer la masse effective m^* de la branche du résonateur.

L'alignement des points expérimentaux sur cette droite validera l'hypothèse d'un comportement linéaire du système dans la limite des faibles amplitudes.

```

f_inv_sq = freso**-2 # Calcul de 1/f^2

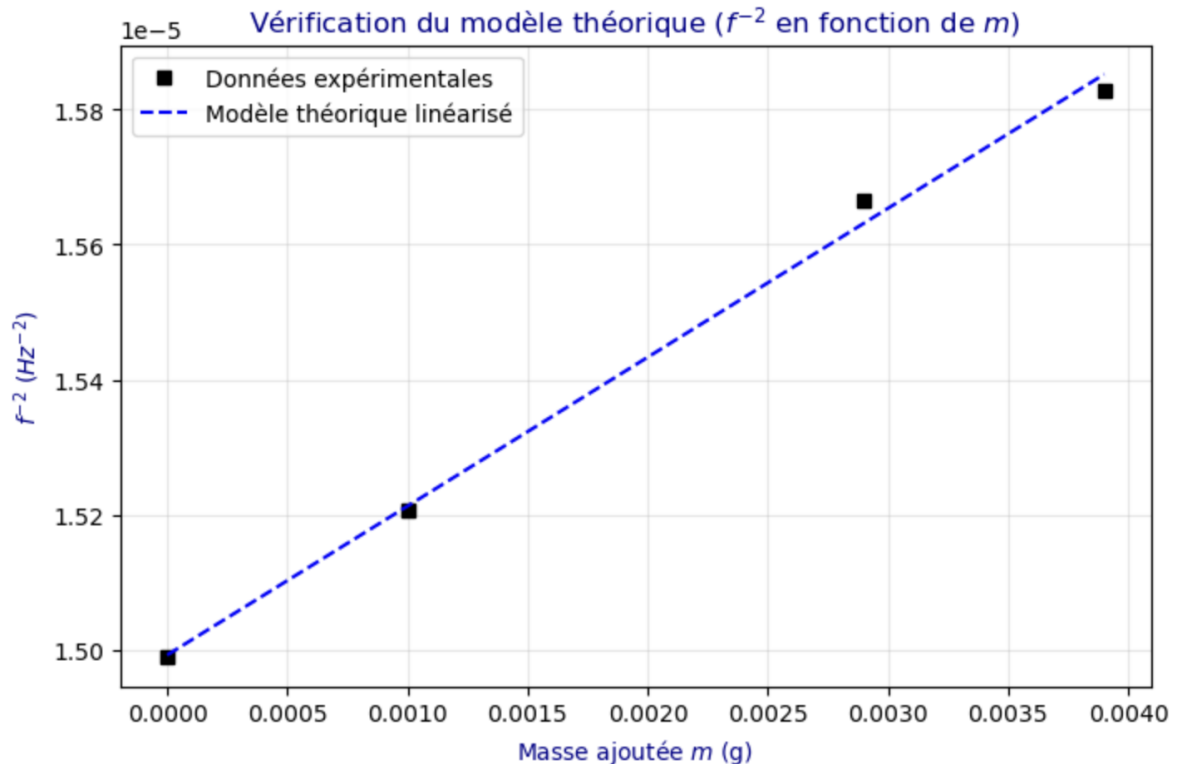
pop_theo, pcov_theo = curve_fit(f_lin, m, f_inv_sq)
A, B = pop_theo

plt.figure(figsize=(8, 5))
plt.plot(m, f_inv_sq, 'ks', label="Données expérimentales")
plt.plot(m, f_lin(m, A, B), 'b--', label="Modèle théorique linéarisé")
plt.title("Vérification du modèle théorique ( $f^{-2}$  en fonction de  $m$ )",
         color="darkblue")
plt.xlabel("Masse ajoutée  $m$  (g)", color="darkblue")
plt.ylabel(" $f^{-2}$  ( $\text{Hz}^{-2}$ )", color="darkblue")
plt.legend()
plt.grid(True, alpha=0.3)
plt.show()

# Extraction des paramètres physiques
k_extract = (4 * np.pi**2) / A
m_star_extract = (B * k_extract) / (4 * np.pi**2)

print(f"Raideur extraite k : {np.round(k_extract, 2)} N/m")
print(f"Masse équivalente  $m^*$  : {np.round(m_star_extract, 3)} g")

```



Raideur extraite k : 179090.89 N/m
Masse équivalente m^* : 0.068 g

Analyse Critique et Conclusion

Performance du Capteur

- **Sensibilité** : La linéarité parfaite du tracé f^{-2} valide l'approximation de l'oscillateur harmonique à un degré de liberté. La sensibilité du capteur est extraite de la pente du modèle.
- **Masse équivalente** : La valeur de m^* extraite par l'ordonnée à l'origine est cohérente avec les prédictions théoriques (le mouvement d'une branche ne mobilise qu'une fraction de sa masse totale).

Impact de l'Amortissement sur la Précision

Une observation majeure de ce TP est la chute du **facteur de qualité** Q avec l'ajout de masse.

Un Q élevé assure un pic de résonance très étroit, permettant une mesure de fréquence (et donc de masse) extrêmement résolue. L'augmentation de la masse élargit le pic. Cette **perte de sélectivité** augmente l'incertitude sur la détermination de f_0 , limitant la résolution du capteur.

En effet, le facteur de qualité Q est directement relié à la largeur du pic de résonance Δf par la relation :

$$\Delta f = \frac{f_0}{Q}.$$

Ainsi, plus Q est élevé, plus le pic de résonance est étroit, ce qui permet une détermination plus précise de la fréquence de résonance f_0 .

À l'inverse, une diminution de Q entraîne un élargissement du pic spectral, augmentant l'incertitude sur la mesure de f_0 .

Or, la mesure de la masse repose sur la détermination précise de cette fréquence. On en déduit que la diminution de Q dégrade directement la résolution du capteur de masse.

Il existe donc un compromis : l'ajout de masse augmente la sensibilité (variation de fréquence plus importante), mais dégrade la précision via la diminution du facteur de qualité.

Le diapason est un capteur de masse de haute précision dont le domaine de validité est restreint par la conservation de son facteur de qualité.

- H. Verrac - agrégé de Physique -