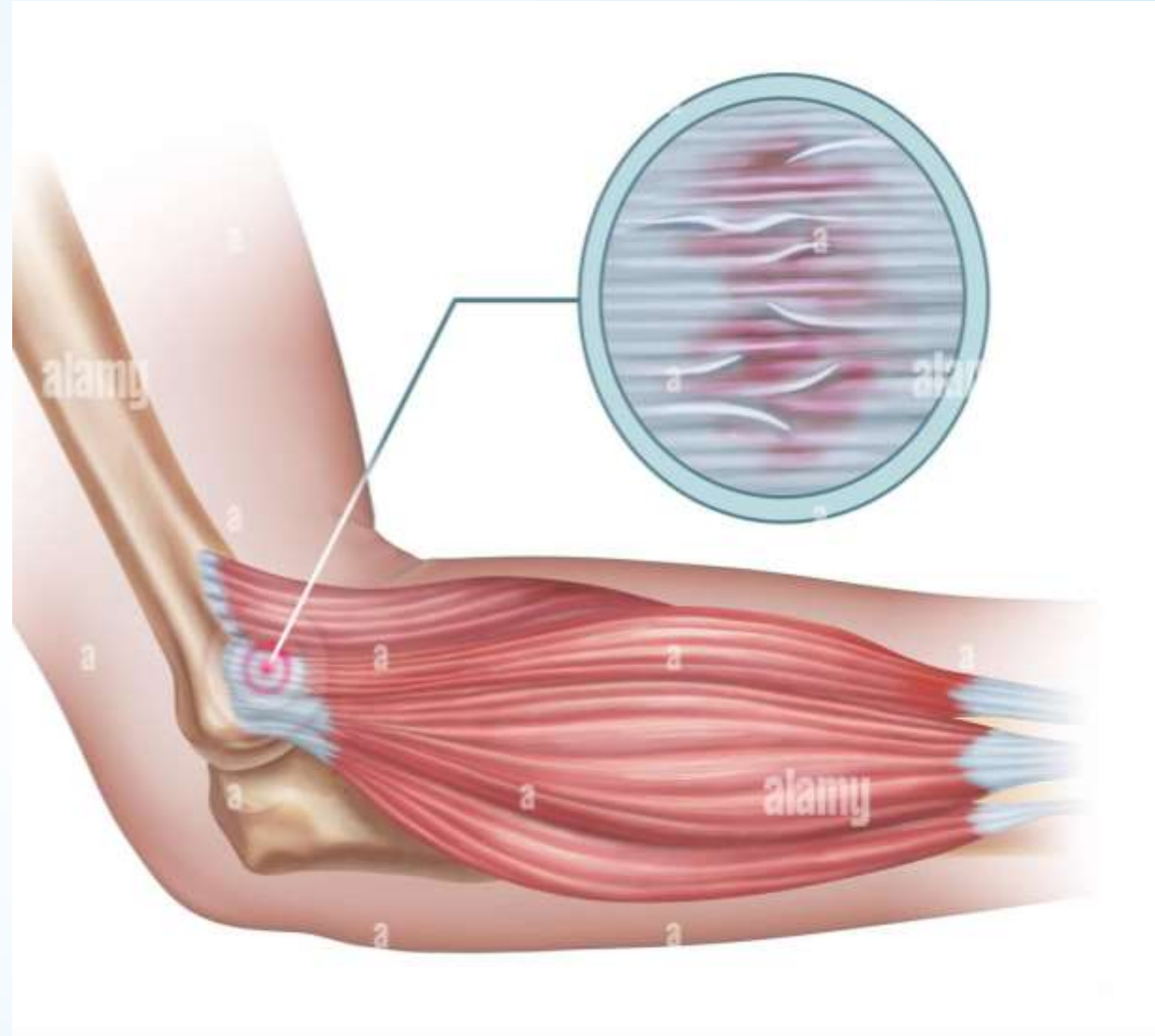




¹ Etude d'une raquette piézoélectrique

Ancrage du sujet :

➡ L'épicondylite



Prototype de la raquette piézoélectrique

► Principe global



**Comment fonctionne le système de
réduction de vibrations à
piézoélectricité dans une raquette ?**

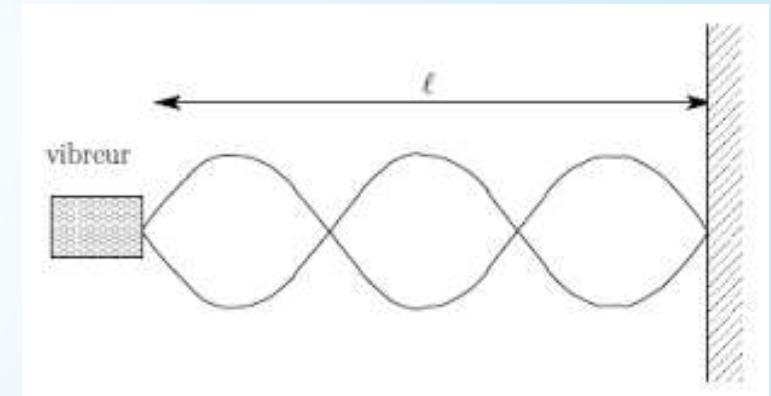
Plan de l'étude :

- I) Modélisation des cordes par la corde de MELDE
- II) Compréhension du capteur piézoélectrique
- III) Reproduction du système de réduction de vibrations

I) Modélisation des cordes par la corde de MELDE

A) *Pourquoi souhaite-on modéliser le phénomène des cordes vibrantes par celui de la Corde de Melde ?*

- Cadre relativement similaire : ondes stationnaires
- Commodité des résultats de la corde de Melde



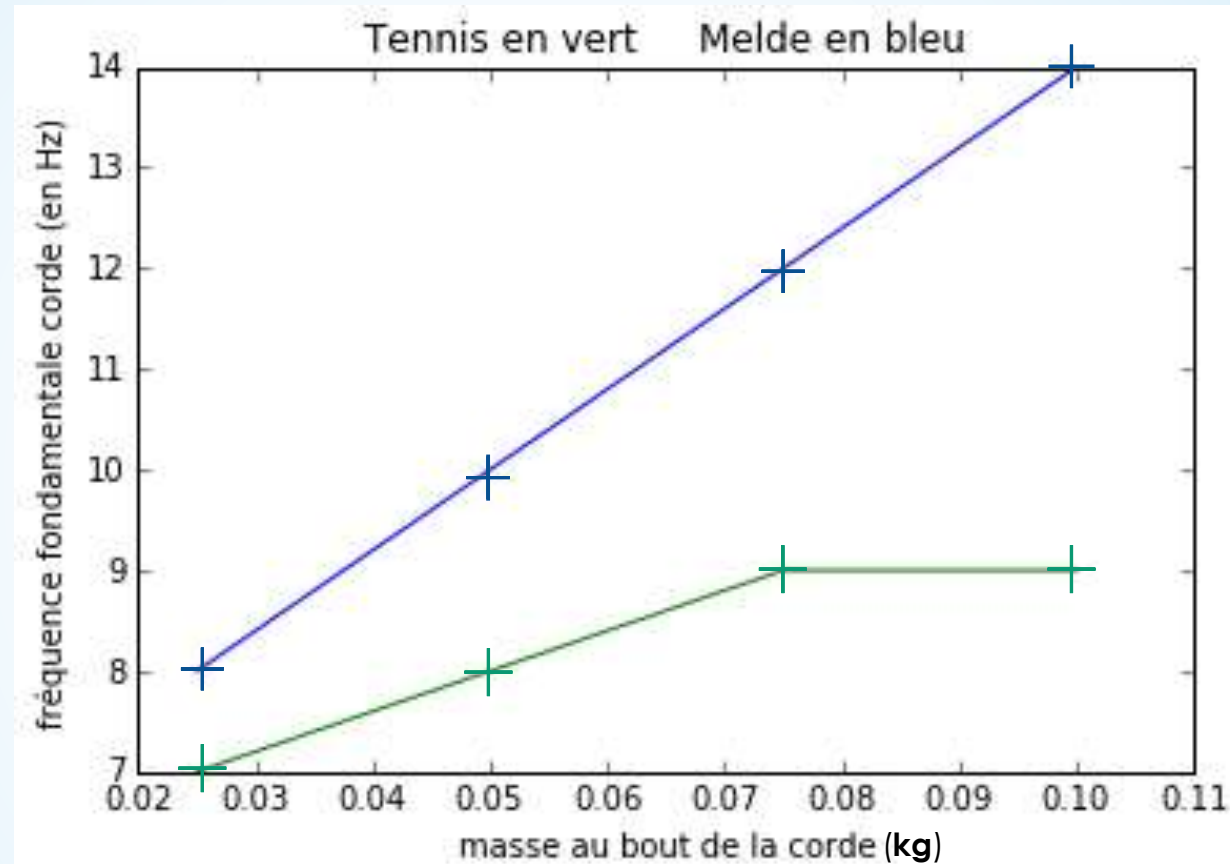
B) Vérification expérimentale

Protocole :

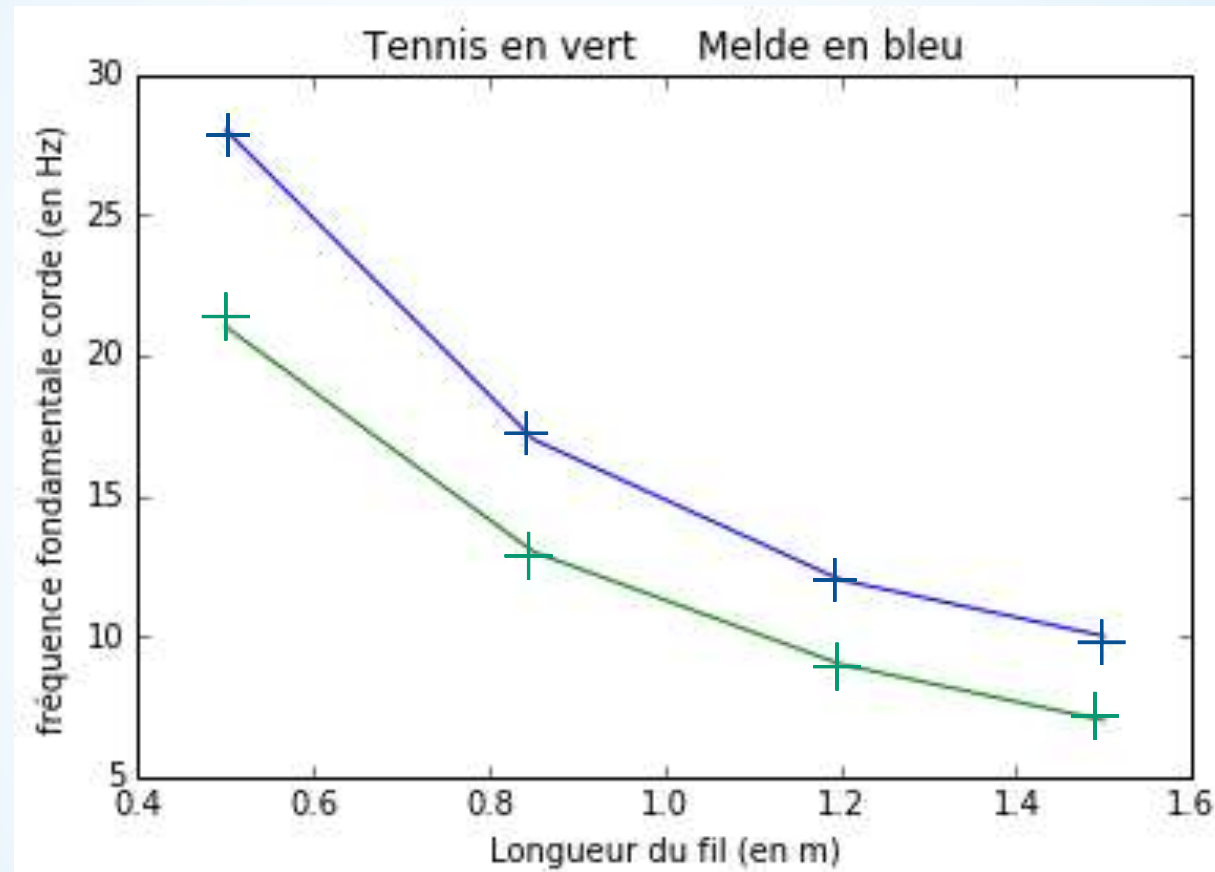
- **Expérimentation à longueur constante**
- **Expérimentation à masse constante**



C) Résultats et conclusion



- ➡ Modélisation réfutée pour les essais à longueur constante (120 cm)



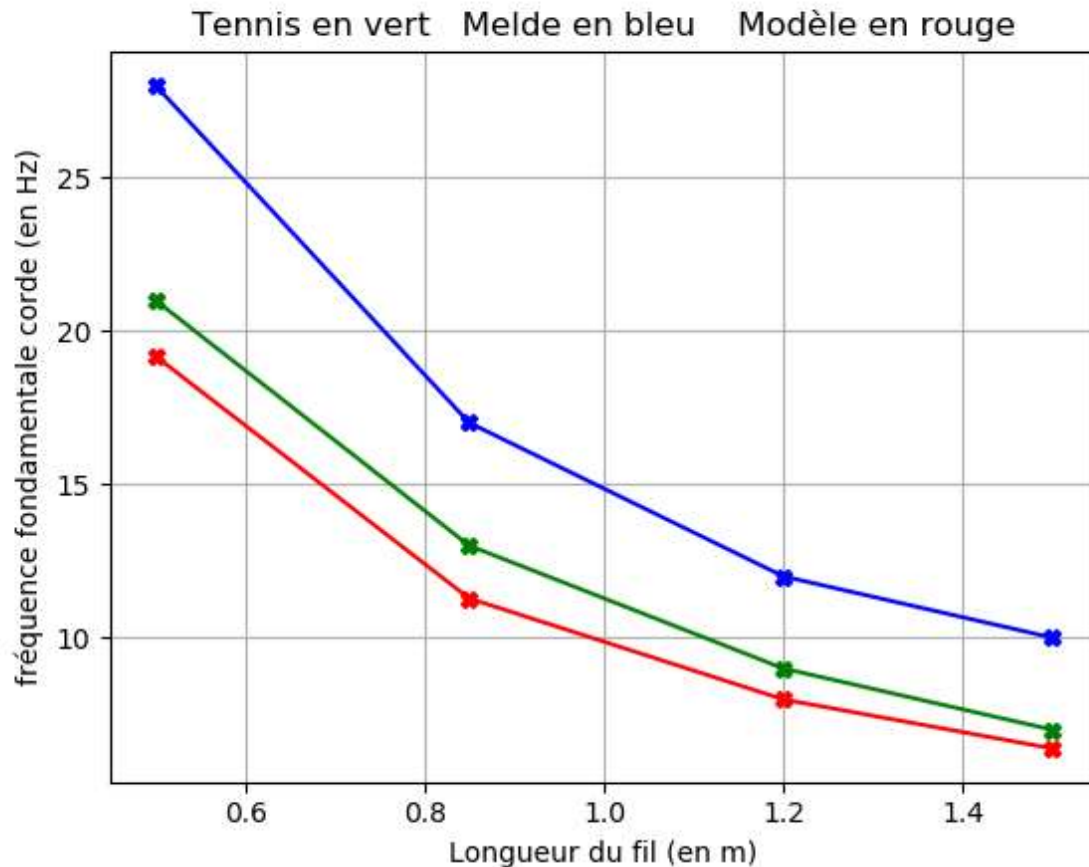
Mesure du
mode propre
 $n=1$

- ➡ La modélisation semble valide pour les essais à masse constante (75 g)

Exploitation des courbes

- ➔ Tendance en $f = K \frac{1}{L}$
- ➔ Modèle de la corde de Melde : $f = \frac{nc}{2} \cdot \frac{1}{L}$
- ➔ $C = \sqrt{\frac{T}{\mu}}$ avec $\mu = \frac{\text{masse de la corde}}{\text{longueur de la corde}}$

Vérification du modèle $f = \frac{nc}{2} \cdot \frac{1}{L}$

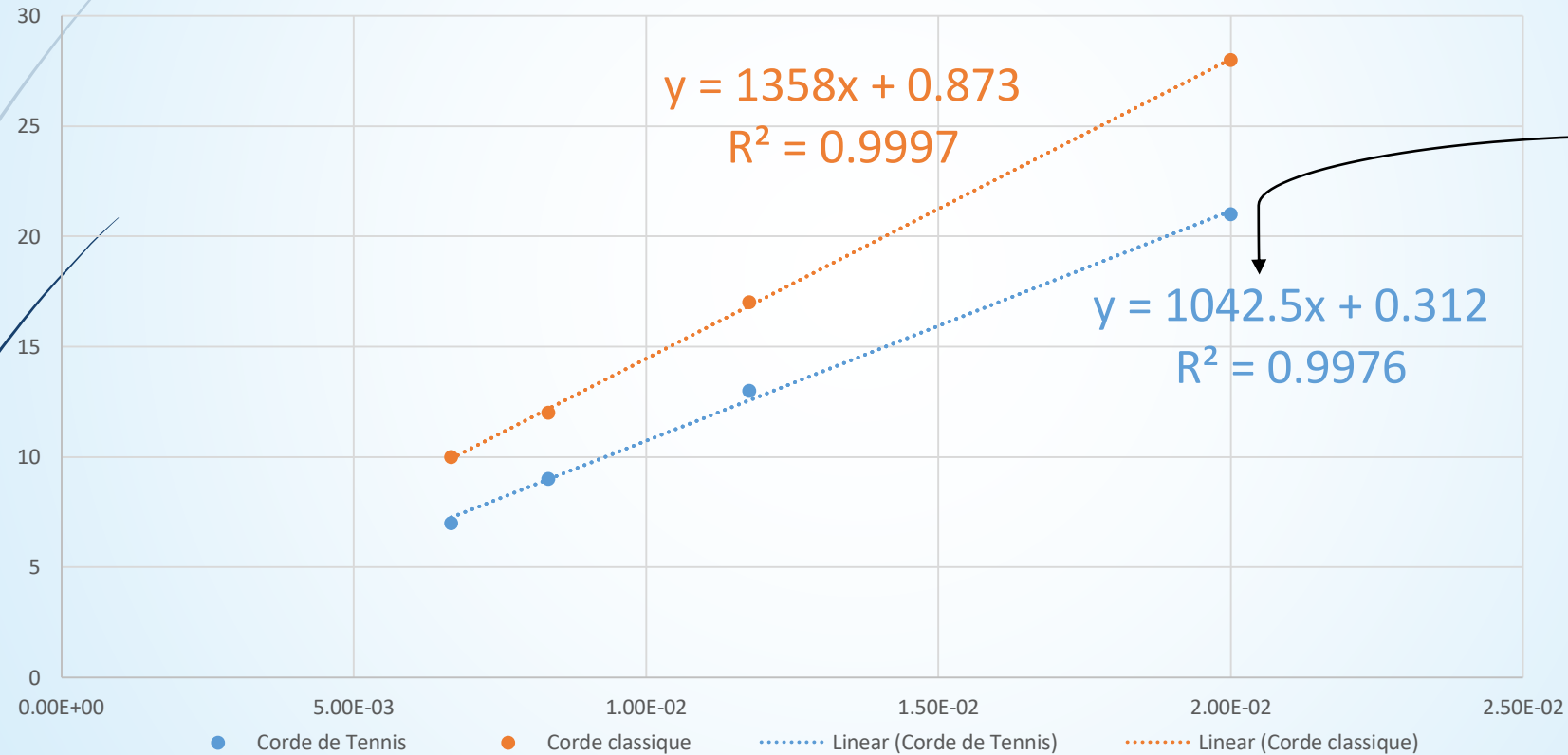


Courbe modèle et courbe corde de Tennis très proches (écarts expliqués par les incertitudes de mesure)

⇒ Modélisation vérifiée pour les essais à masse constante (75 g)

Vérification de la linéarité de $f = K \times \frac{1}{L}$

Fréquence (en Hz)



$$1042.5 = \frac{nc}{2}$$

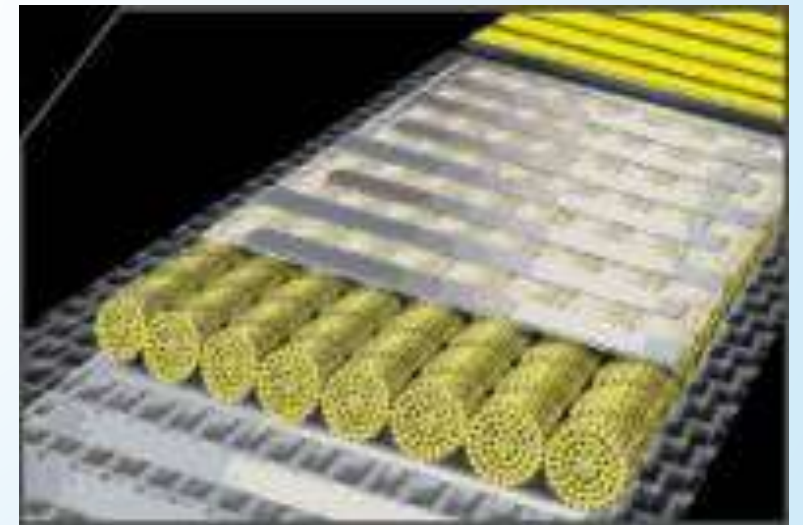
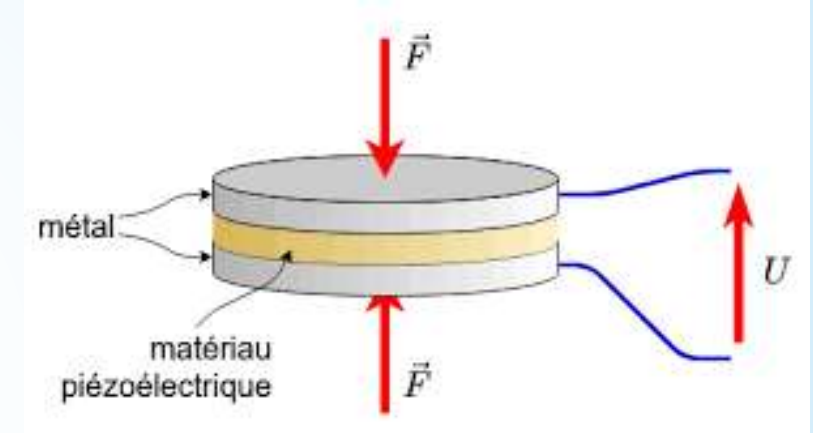
$$\frac{1}{L} \text{ (en } m^{-1} \text{)}$$

II) Compréhension du capteur piézoélectrique

13

A) Partie théorique

➔ Principe du capteur



B) Mesure des vibrations sur la raquette



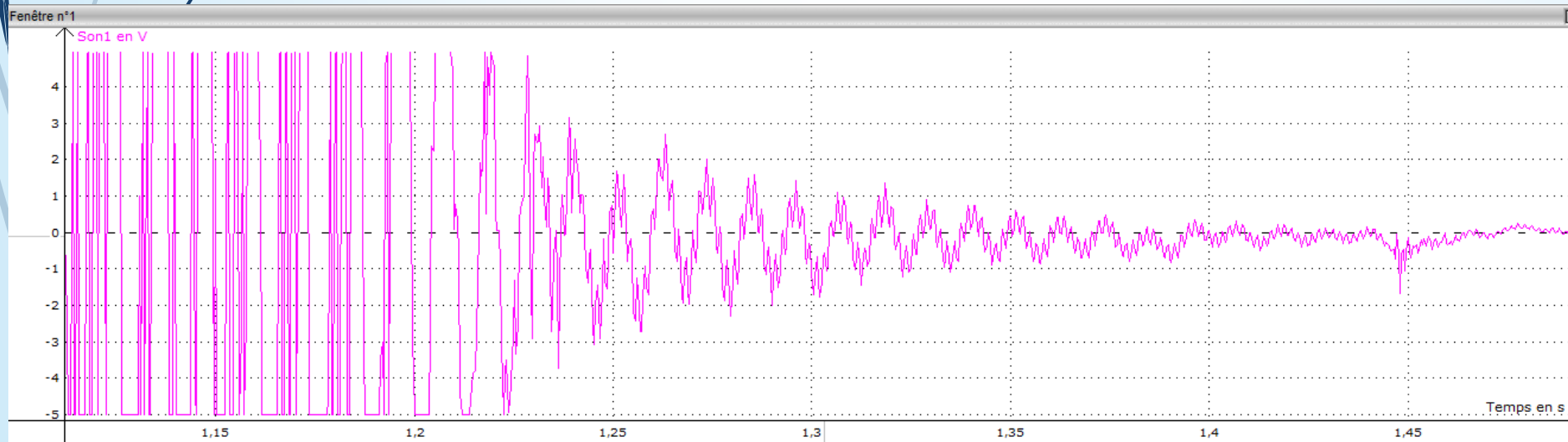
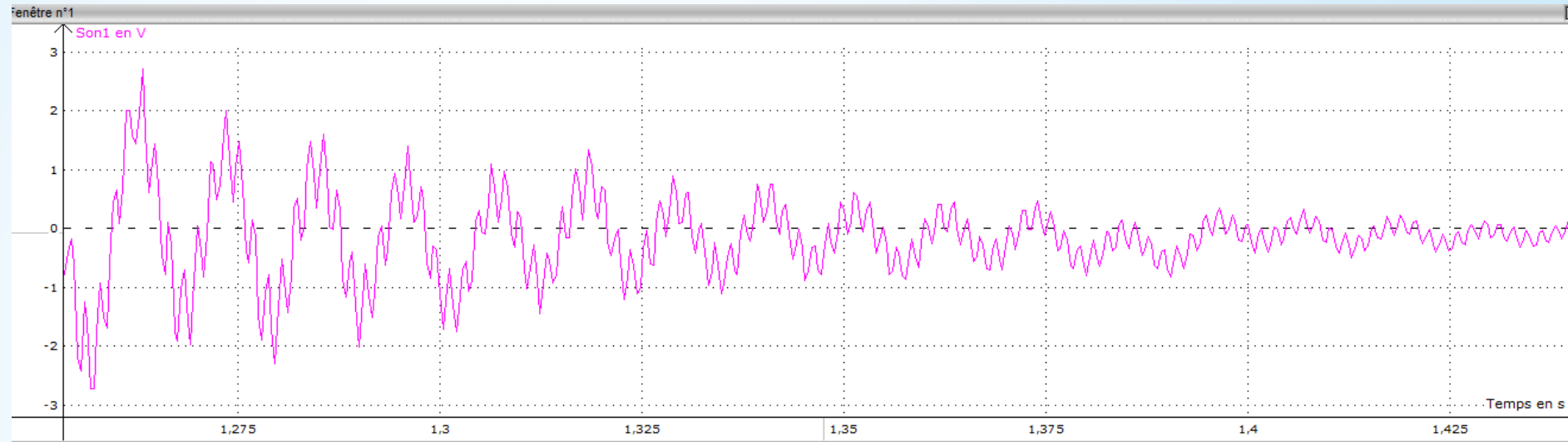
Choix du capteur permis
par les ordres de
grandeurs de fréquence
mesurés précédemment

Amplificateur branché à la carte d'acquisition

Capteur piézoélectrique

C) Tests réalisés et résultats

- Test avec capteur piézoélectrique posé sur la raquette

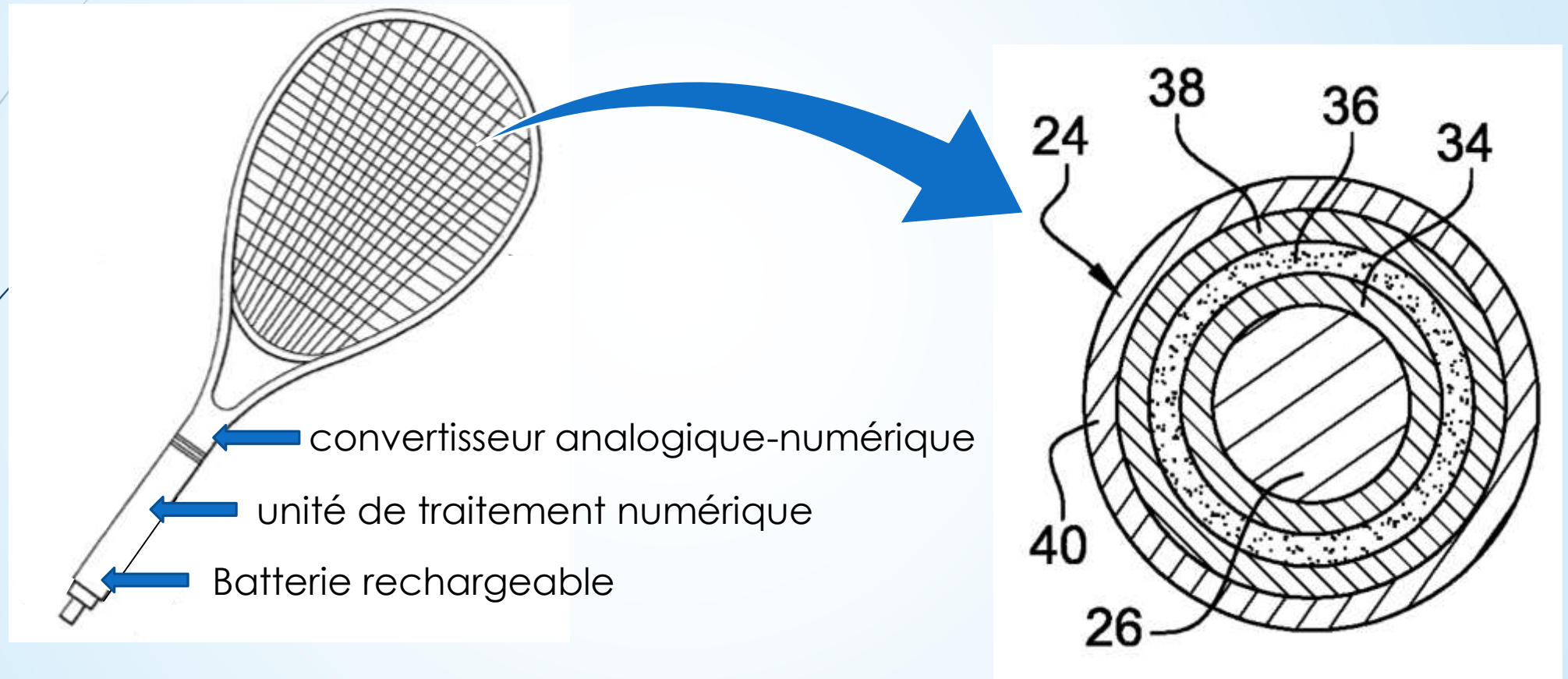


Test avec capteur scotché à la raquette qui est tenue dans la main

III) Reproduction du système de réduction de vibration

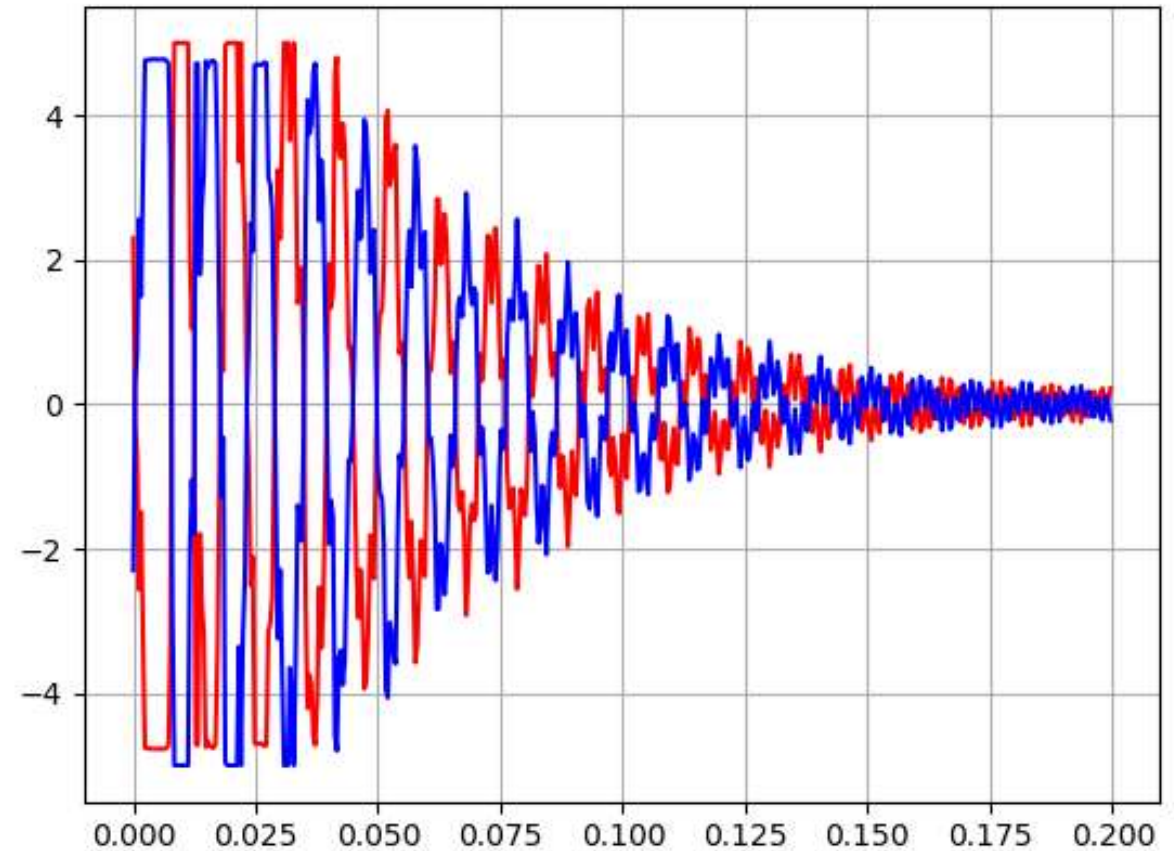
16

➔ A) Principe de la raquette piézoélectrique



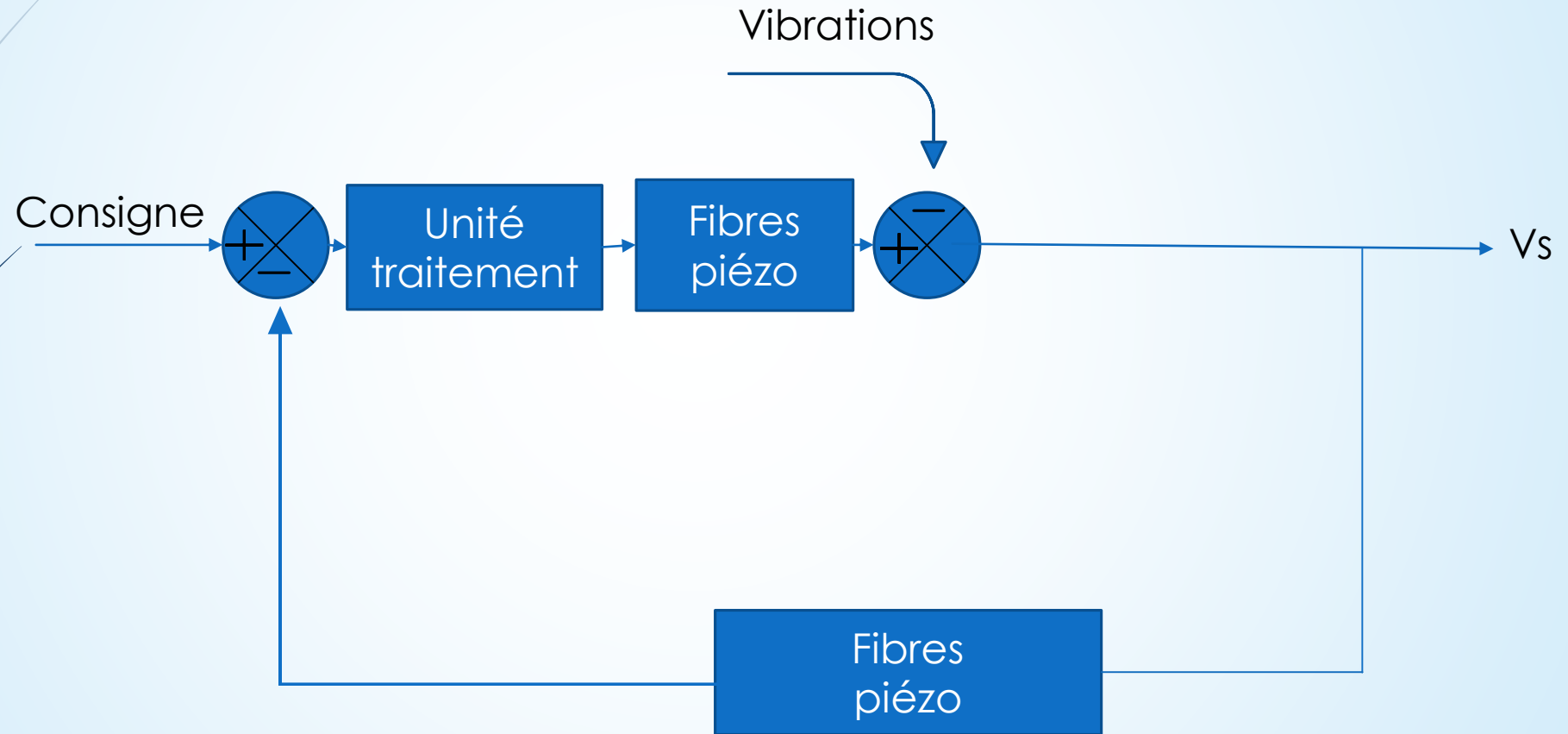
B) Reproduction du système

- Le signal « inversé » se superpose parfaitement
- Signal bruité (dû aux défauts du capteur)
- Tentative de lissage (échec)



➡ Problème de temps de réponse

C) Optimisation : système asservi



Critique et proposition d'amélioration de l'amortissement

- Est-ce que la fabrication des raquettes en matériaux plus souples n'est pas suffisante?
- Est-ce que le coût des nombreux composants en vaut la peine ?
- Pourquoi cette raquette ne s'est pas commercialisée ?

**Merci de votre attention,
avez-vous des questions ?**



Annexes

➤ Résultats corde de Melde

1	m en gramme	L en cm	fréquence fondamentale en Hz				m en gramme	L en cm	fréquence fondamentale en Hz
2	25	120	8				25	120	7
3	50	120	10				50	120	8
4	75	120	12				75	120	9
5	100	120	14				100	120	9
6									
7	75	50	28				75	50	21
8	75	85	17				75	85	13
9	75	120	12				75	120	9
10	75	150	10				75	150	7

Corde en tissu

Corde de tennis
(multi filament synthétique)

Programme python : tracés à masse et longueur constantes

```
import matplotlib.pyplot as plt
#m = 0.075 kg
L = [0.5,0.85,1.20,1.50]
f0_melde = [28,17,12,10]
f0_tennis = [21,13,9,7]

plt.plot(L,f0_melde)
plt.xlabel("Longueur du fil (en m)")
plt.ylabel("fréquence fondamentale corde (en Hz)")

plt.plot(L,f0_tennis)
plt.title("Tennis en vert      Melde en bleu")

plt.show()
```

```
import matplotlib.pyplot as plt
#import numpy as n
#L = 1.20 m
m = [0.025,0.05,0.075,0.1]
f0_melde = [8,10,12,14]
f0_tennis = [7,8,9,9]

plt.plot(m,f0_melde)
plt.xlabel("masse au bout de la corde")
plt.ylabel("fréquence fondamentale corde (en Hz)")

plt.plot(m,f0_tennis)
plt.title("Tennis en vert      Melde en bleu")

plt.show()
```

Programme python : modèle

```
import matplotlib.pyplot as plt
#m = 0.075 kg
L = [0.5,0.85,1.20,1.50]
f0_melde = [28,17,12,10]
f0_tennis = [21,13,9,7]

L_modele = [(1/0.5)*19.18/2,(1/0.85)*19.18/2,(1/1.20)*19.18/2,(1/1.50)*19.18/2]

plt.plot(L,f0_melde)
plt.xlabel("Longueur du fil (en m)")
plt.ylabel("fréquence fondamentale corde (en Hz)")

plt.plot(L,f0_tennis)
plt.plot(L,L_modele)
plt.grid()
plt.title("Tennis en vert   Melde en bleu   Modèle en orange")

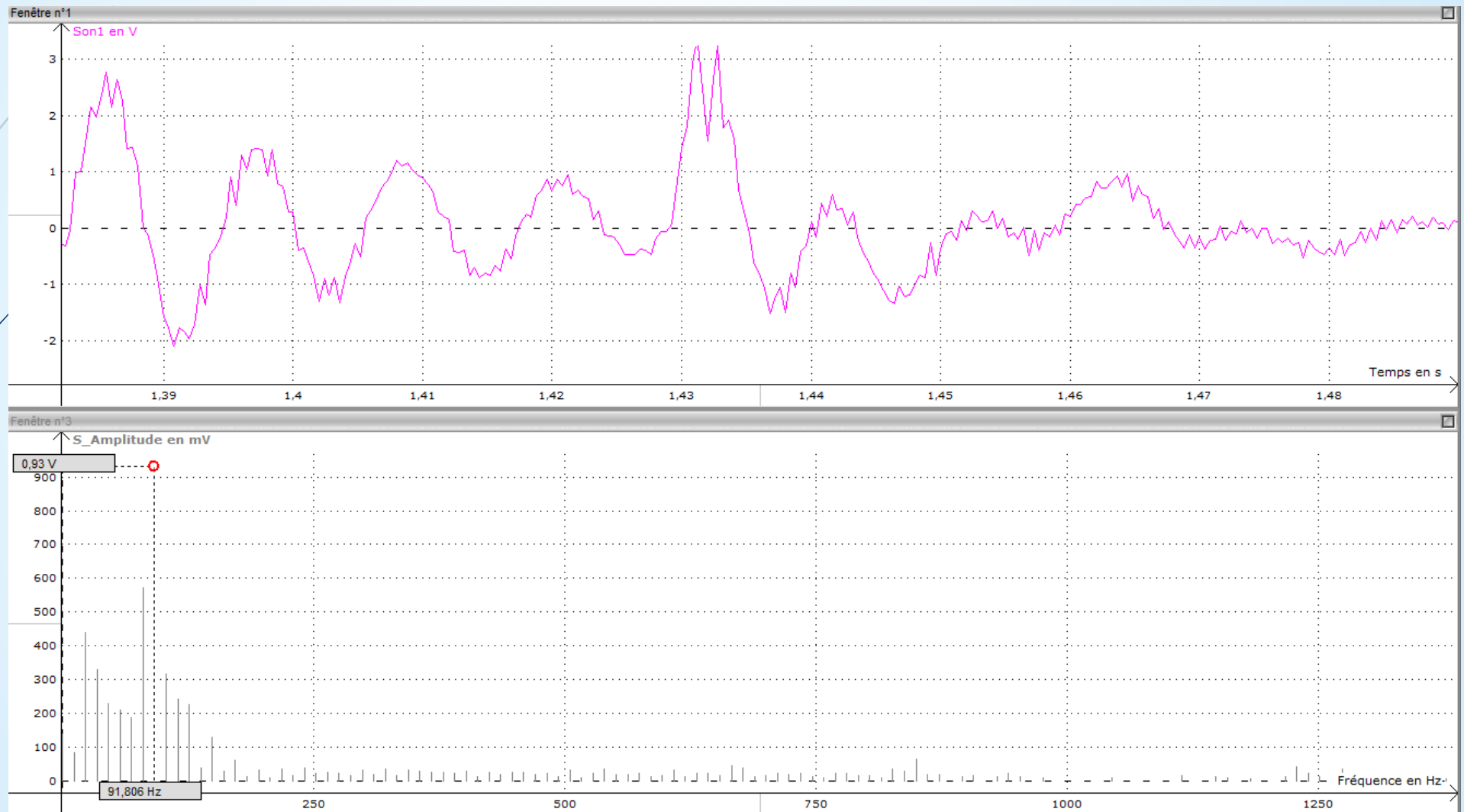
plt.show()
```

But : comparer la courbe expérimentale de la corde de tennis avec le modèle

Masse linéique et célérité

- ➡ $\mu_{Tennis} = 2.10^{-3} \text{ kg.m}^{-1}$ (mesure sur la corde de Tennis)
- ➡ $\mu_{Melde} = 1.2.10^{-3} \text{ kg.m}^{-1}$ (mesure sur la corde de Melde)
- ➡ $c_{Tennis} = 19.18 \text{ m.s}^{-1}$
- ➡ $c_{Melde} = 24.76 \text{ m.s}^{-1}$

Test avec capteur non fixé sur la raquette



Programme réduction de vibrations

```
def separation(nom_fichier,n):  
    fichier = open(nom_fichier,'r')  
    fichier.readline()  
    T = [ ]  
    V = [ ]  
    for i in range (n):  
        ligne = fichier.readline()  
        ligne = ligne.replace(",",".")  
        ligne = ligne.replace("'",'"')  
        ligne = ligne.strip()  
        ligne = ligne.split(';')  
        T.append(float(ligne[0]))  
        V.append(float(ligne[1]))  
    return T,V  
  
def inversion (temps, tension,n):  
    inv = [ ]  
    for i in range (n):  
        inv.append(-(float(tension[i])))  
    plt.plot(temps,inv,'r')  
    plt.plot(temps, tension,'b')  
    plt.grid()  
    plt.show()  
    return inv,temps
```



La fonction separation prend en paramètre un fichier csv ainsi que le nombre de valeurs relevé et renvoie deux listes T le temps (en s) et V la tension (en V)



Création et remplissage de la liste inv, l'opposé de la liste V

Fonction lissage et moyenne

```
def moyenne (V):  
    count = 0  
    somme = 0  
    for k in V:  
        count+=1  
        somme += k  
    return somme/count  
  
def lissage(V,T,p):  
    L_lisse = []  
    Temps = []  
    for k in range(p,len(V)-p):  
        L_valeur = []  
        for i in range(k-p,k+p):  
            L_valeur.append(V[i])  
        L_lisse.append(moyenne(L_valeur))  
  
    for k in range(0+p,len(T)-p):  
        Temps.append(T[k]/500)  
    return L_lisse,Temps  
  
def tracé (tension,tensioninv, temps):  
    plt.plot(temps,inv,'r')  
    plt.plot(temps, tension,'b')  
  
    plt.grid()  
    plt.show()  
  
T,V = separation('Test_raquette_tendue_et_piezo_colle.csv',500)  
inv, T = inversion(T,V,500)  
Llisse, T1 = lissage(V,T,5)  
Llisseinv,T1 = lissage(inv,T,5)  
tracé(Llisse,Llisseinv,T1)
```

**Programme permettant
de lisser la courbe afin
de limiter le bruit**

Sources des images

- Diapo 1 : <https://sportfoy.com/2021/02/16/mens-tennis-greats-of-this-generation/>
- Diapo 2 : <https://www.alamyimages.fr/photos-images/%C3%A9picondylite.html>
- Diapo 10 Capteur piézoélectrique : <https://si.blaisepascal.fr/1t-les-capteurs/>
- Diapo 10 Cordage : <https://engineering.purdue.edu/MSE/aboutus/gotmaterials/Sports/ruh.html>
- Diapo 13 Raquette : <https://patentimages.com/patent-image/us8449410-1-figure-1/>
- Diapo 13 Cordage :
<https://patentimages.storage.googleapis.com/91/cb/7c/b9491b7d6d2e9a/EP2937117B1.pdf>