

# ajustement

September 28, 2022

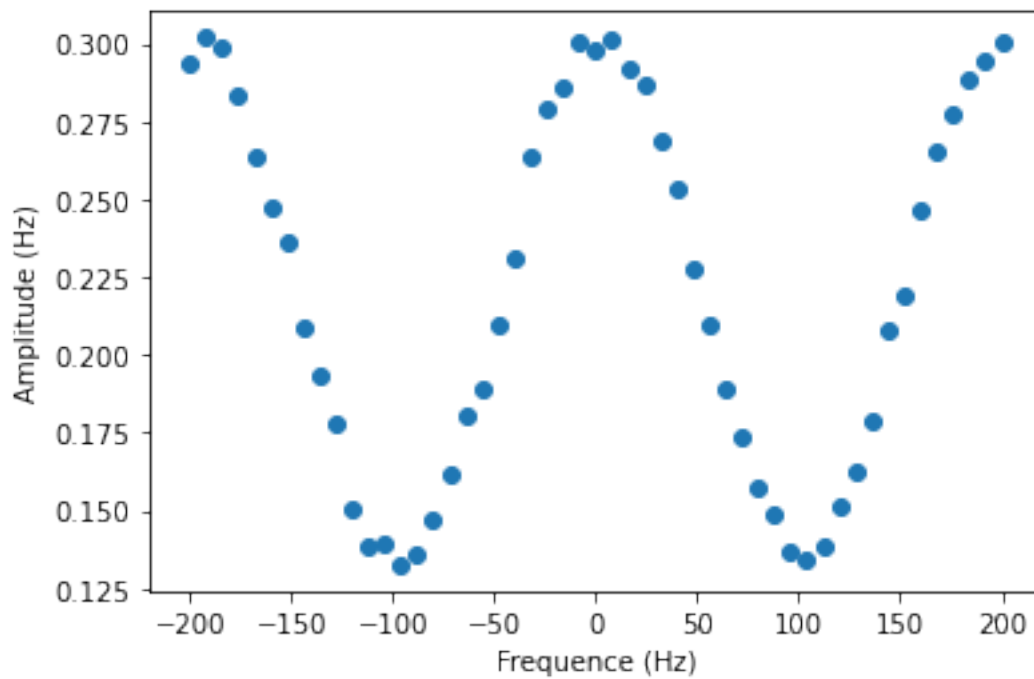
## 0.0.1 Ajustement

Ajuster les franges d'un interféromètre atomique \* Faire un modèle avec les bons paramètres (ni trop, ni pas assez) \* Trouver des paramètres initiaux approximatif avant de faire le fit \* Vérifier que le fit à bien convergé !

```
[1]: import numpy as np
import matplotlib.pyplot as plt
from scipy.optimize import curve_fit
freq, amp = np.loadtxt('fit_sinus.dat', unpack=True)
```

```
[2]: plt.plot(freq, amp, 'o')
plt.xlabel('Frequence (Hz)')
plt.ylabel('Amplitude (Hz)')
```

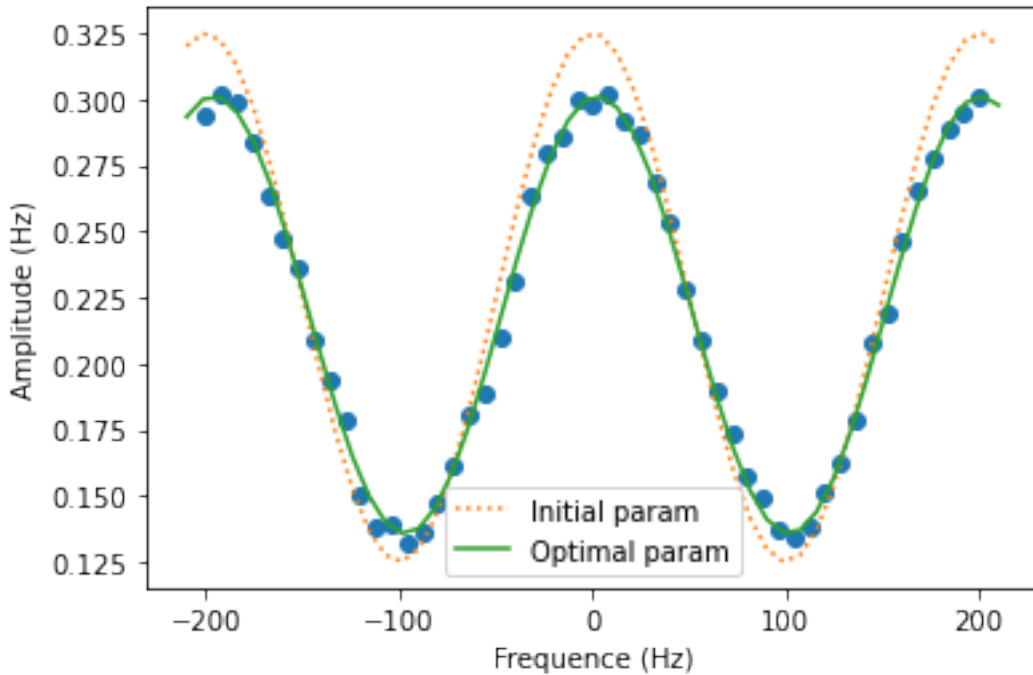
```
[2]: Text(0, 0.5, 'Amplitude (Hz)')
```



```
[3]: def modele(x, amplitude, offset, position, largeur):  
      return offset + amplitude*(1+np.cos(2*np.pi*(x-position)/largeur))/2
```

```
[8]: x_plot = np.linspace(-210, 210)  
  
p0 = (0.2, 0.125, 0, 200)  
plt.plot(freq, amp, 'o')  
  
# Commenter cette ligne une fois le fit effectué  
plt.plot(x_plot, modele(x_plot, *p0), ':', label='Initial param')  
  
popt, pcov = curve_fit(modele, freq, amp, p0) # toujours mettre p0  
plt.plot(x_plot, modele(x_plot, *popt), label='Optimal param')  
  
plt.xlabel('Frequence (Hz)')  
plt.ylabel('Amplitude (Hz)')  
plt.legend()
```

[8]: <matplotlib.legend.Legend at 0x7fb3e258cbb0>



[ ]: