Résonateur de Tesla

0.1 Résonance

[1]:
```
import numpy as np
import matplotlib.pyplot as plt
L1,L2=20e-6,50e-3
M=np.sqrt(L1*L2)*0.7
R1,C2,R2=2,5e-12,3
w0=1/np.sqrt(C2*L2)
f0=w0/2/np.pi
C1=1/(L1*w0**2)
print(f0)
print(C1)
```

```
318309.8861837907
1.25e-08
```

Lois des mailles + des bobines:

$$u_1 + R_1 i_1 + L_1 \frac{di_1}{dt} + M\frac{di_2}{dt} = e$$
$$u_2 + R_2 i_2 + L_2 \frac{di_2}{dt} + M\frac{di_1}{dt} = 0$$

Lois des condensateurs:

$$i_1 = C_1 \frac{du_1}{dt}$$
$$i_2 = C_2 \frac{du_2}{dt}$$

Sous forme matricielle:

$$\begin{pmatrix} 1 & 0 & R_1 + jL_1\omega & jM\omega \\ 0 & 1 & jM\omega & R_2 + jL_2\omega \\ jC_1\omega & 0 & -1 & 0 \\ 0 & jC_2\omega & 0 & -1 \end{pmatrix} \cdot \begin{pmatrix} \underline{u_1} \\ \underline{u_2} \\ \underline{i_1} \\ \underline{i_2} \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

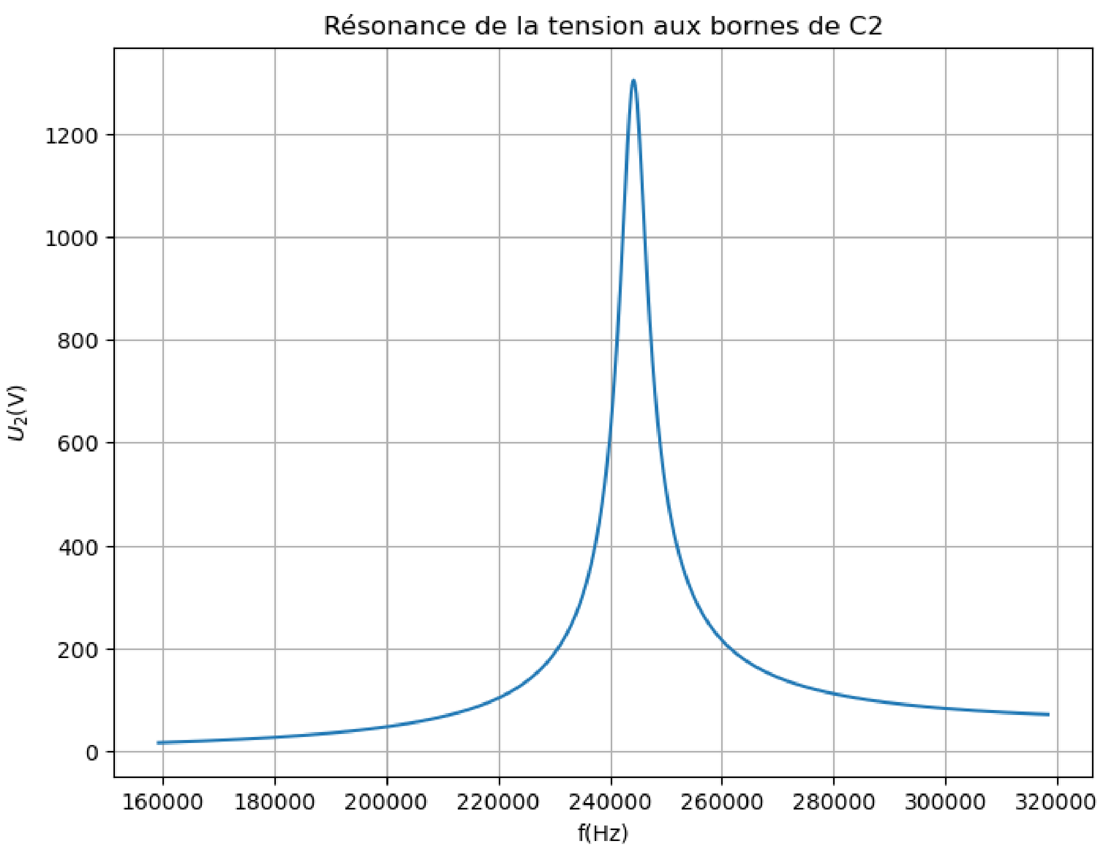
[2]:
```
import numpy.linalg as npl
f=100e3
w=2*np.pi*f
A=np.
 ↪array([[1,0,R1+1j*L1*w,1j*M*w],[0,1,1j*M*w,R2+1j*L2*w],[1j*C1*w,0,-1,0],[0,1j*C2*w,0,-1]])
```

```
B=np.array([1,0,0,0])
X=npl.solve(A,B)
print(X)
```

```
[1.11571825e+00-1.95597789e-02j 4.27613070e+00-7.50100441e-02j
 1.53622144e-04+8.76283068e-03j 2.35651003e-07+1.34338608e-05j]
```

[3]:

```
tab_f=np.linspace(0.5*f0,1*f0,1000)
tab_u2=np.zeros_like(tab_f)
for i in range(len(tab_f)):
    w=tab_f[i]*2*np.pi
    A=np.
 ↪array([[1,0,R1+1j*L1*w,1j*M*w],[0,1,1j*M*w,R2+1j*L2*w],[1j*C1*w,0,-1,0],[0,1j*C2*w,0,-1]])
    B=np.array([1,0,0,0])
    X=npl.solve(A,B)
    tab_u2[i]=abs(X[1])
plt.figure(figsize=(8,6))
plt.plot(tab_f,tab_u2)
plt.xlabel("f(Hz)")
plt.ylabel(r"$U_2$(V)")
plt.title("Résonance de la tension aux bornes de C2")
plt.grid()
plt.show()
```



Résonance à $244300 Hz$ d'amplitude $1300 V$

Largeur à $-3dB$: $4700 Hz$

$Q = \frac{244300}{4700} = 52$

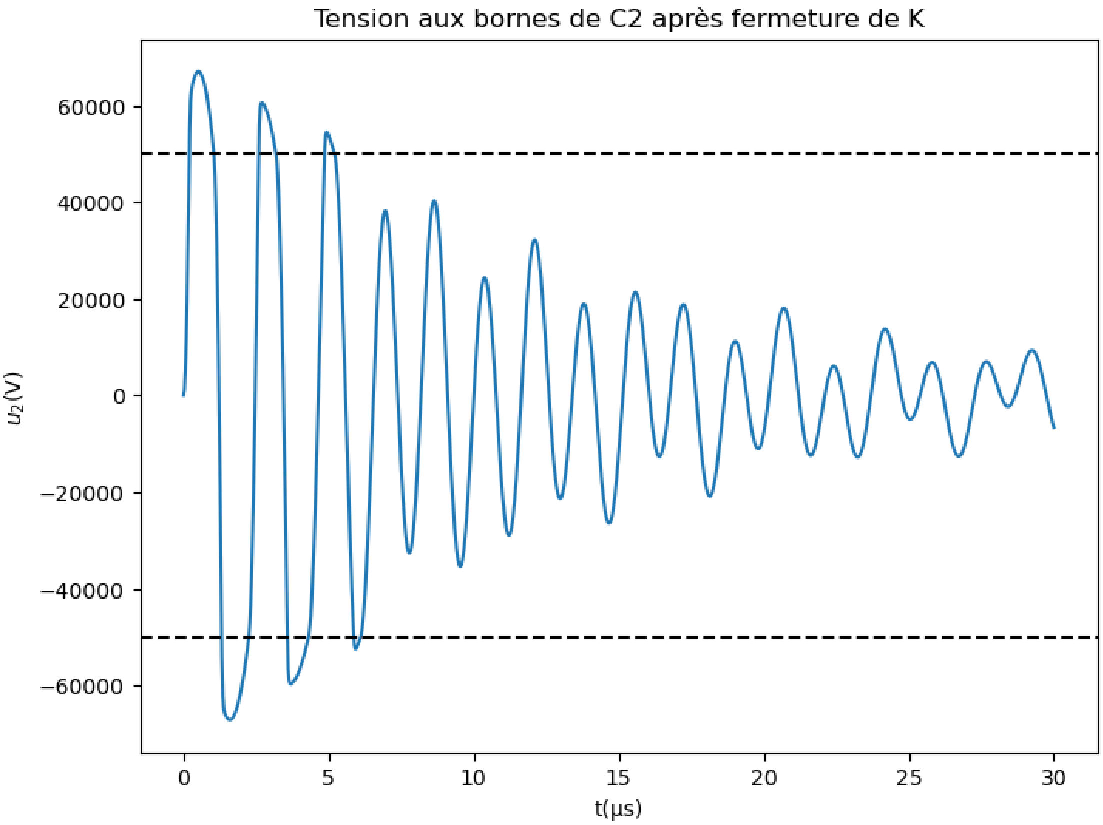
0.2 Étude dynamique de l'éclair

Le système à résoudre est:

$$\begin{cases} \frac{di_1}{dt} = \frac{Mu_2+MR_2i_2-L_2u_1-L_2R_1i_1}{L_1L_2-M^2} \\ \frac{di_2}{dt} = \frac{Mu_1+MR_1i_1-L_1u_2-L_1R_2i_2}{L_1L_2-M^2} \\ \frac{du_1}{dt} = \frac{i_1}{C_1} \\ \frac{du_2}{dt} = \begin{cases} \frac{i_2}{C_2} \text{ si } |u_2| < U_e \\ \frac{i_2}{C_2} - \frac{u_2}{R_eC_2} \text{ sinon avec } R_e = \frac{D}{|u_2|-U_e} \end{cases} \end{cases}$$

[4]:
```
from scipy.integrate import odeint
R2=3
R1=2
D,Ue=3e8,5e4
def Re(u2): # renvoie Re lorsque u2>Ue
    return D/(abs(u2)-Ue)
def derivee(inconnues,t):
    u1,u2,i1,i2=inconnues
    di1dt=(-L2*u1-L2*R1*i1+M*u2+M*R2*i2)/(L1*L2-M**2) ##
    di2dt=(-L1*R2*i2-L1*u2+M*u1+M*R1*i1)/(L1*L2-M**2) ##
    du1dt=i1/C1
    if abs(u2)<Ue:
        du2dt=i2/C2
    else:
        du2dt=i2/C2-u2/Re(u2)/C2
    return [du1dt,du2dt,di1dt,di2dt]
U0=10000
ci=[U0,0,0,0]
tab_t=np.linspace(0,3e-5,1000)
sol=odeint(derivee,ci,tab_t)
tab_u1=sol[:,0]
tab_u2=sol[:,1]
tab_i1=sol[:,2]
tab_i2=sol[:,3]
plt.figure(figsize=(8,6))
plt.plot(tab_t*1e6,tab_u2)
plt.axhline(Ue,ls="--",color="black")
plt.axhline(-Ue,ls="--",color="black")
plt.title("Tension aux bornes de C2 après fermeture de K")
```

```python
plt.xlabel("t(µs)")
plt.ylabel(r"$u_2$(V)")
plt.show()
```



On observe 6 éclairs durant environ $0,8\mu s$.

[ ]:

```python

```

[ ]:

```python

```