Glissé-décolé (stick-slip)

[1]:
```
import numpy as np
import matplotlib.pyplot as plt
```

La corde est soumise à:

- le rappel du ressort $+k.(x_F - x).\vec{e_x} = +k(v_a t - x)\vec{e_x}$
- la force de frottements solides $T\vec{e_x}$
- la force de frottements fluides $-\lambda.\dot{x}.\vec{e_x}$

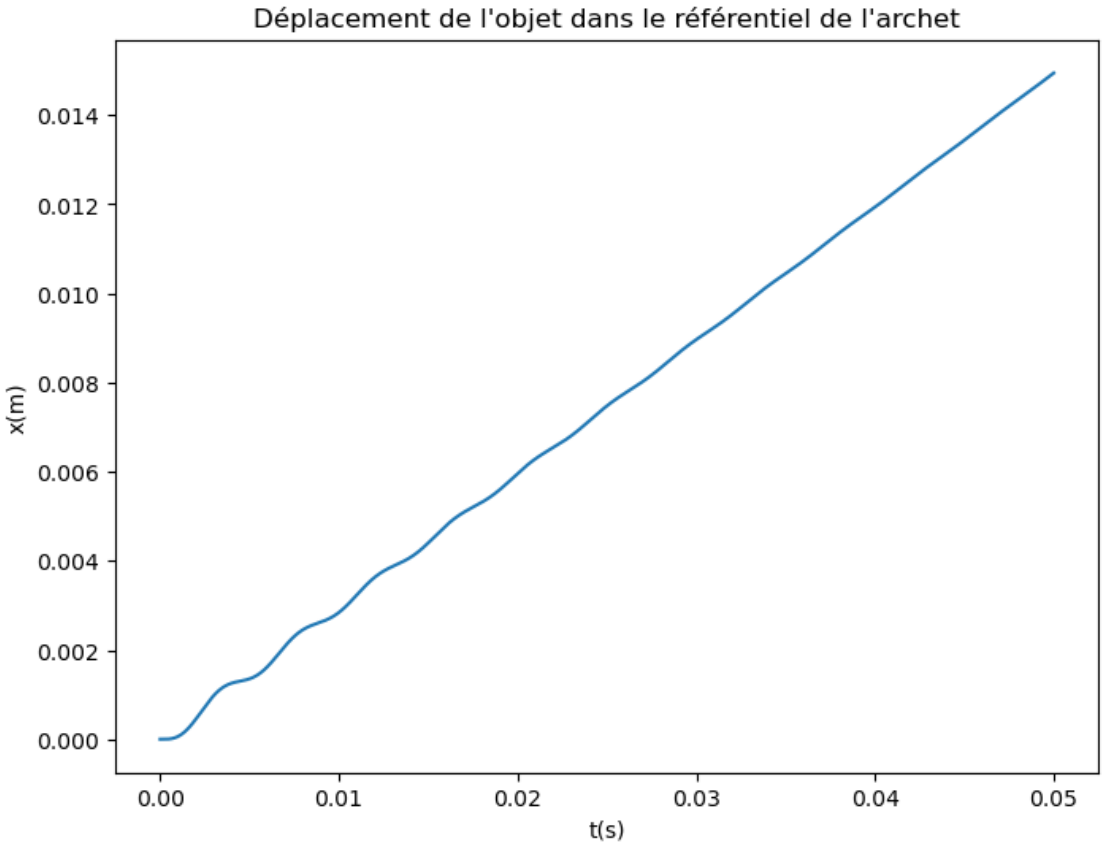
Donc après projection du PFD sur $\vec{e_x}$ on trouve

$$\ddot{x} = \frac{k(v_a t - x) + T - \lambda.\dot{x}}{m}$$

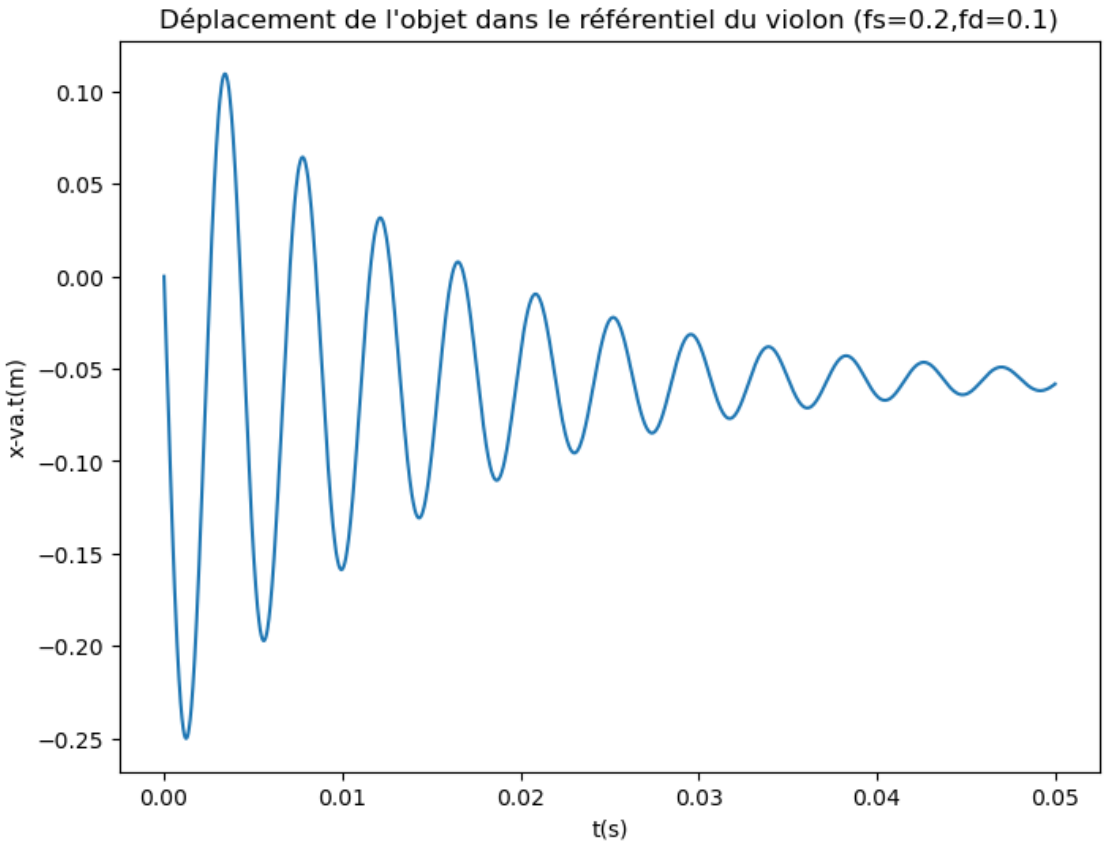
[2]:
```
va=30e-2
k=2500
fs=0.2
fd=0.1
m=1.2e-3
g=9.8
R=0.8
lam=0.2
def acceleration(x,v,t):
    xF=va*t
    Fressort=k*(xF-x)
    Fair=-lam*v
    if v>0:
        T=-fd*R
        return (Fressort+Fair+T)/m
    elif v<0:
        T=+fd*R
        return (Fressort+Fair+T)/m
    else:
        if abs(Fressort+Fair)<fs*R:
            return 0
        else:
            T=-np.sign(Fressort+Fair)*fs*R
            return (Fressort+Fair+T)/m
```

[3]:
```
pas=1e-5
tmax=0.05
N=int(tmax/pas)
tab_t=np.zeros(N+1)
tab_x=np.zeros(N+1)
tab_v=np.zeros(N+1)
tab_t[0],tab_x[0],tab_v[0]=0,0,0
for n in range(N):
    tab_t[n+1]=tab_t[n]+pas
    tab_x[n+1]=tab_x[n]+pas*tab_v[n]
    tab_v[n+1]=tab_v[n]+pas*acceleration(tab_x[n],tab_v[n],tab_t[n])
    if tab_v[n]*tab_v[n+1]<0:
        tab_v[n+1]=0
```

[4]:
```
plt.figure(figsize=(8,6))
plt.plot(tab_t,tab_x)
plt.xlabel("t(s)")
plt.ylabel("x(m)")
plt.title("Déplacement de l'objet dans le référentiel de l'archet")
plt.show()
```



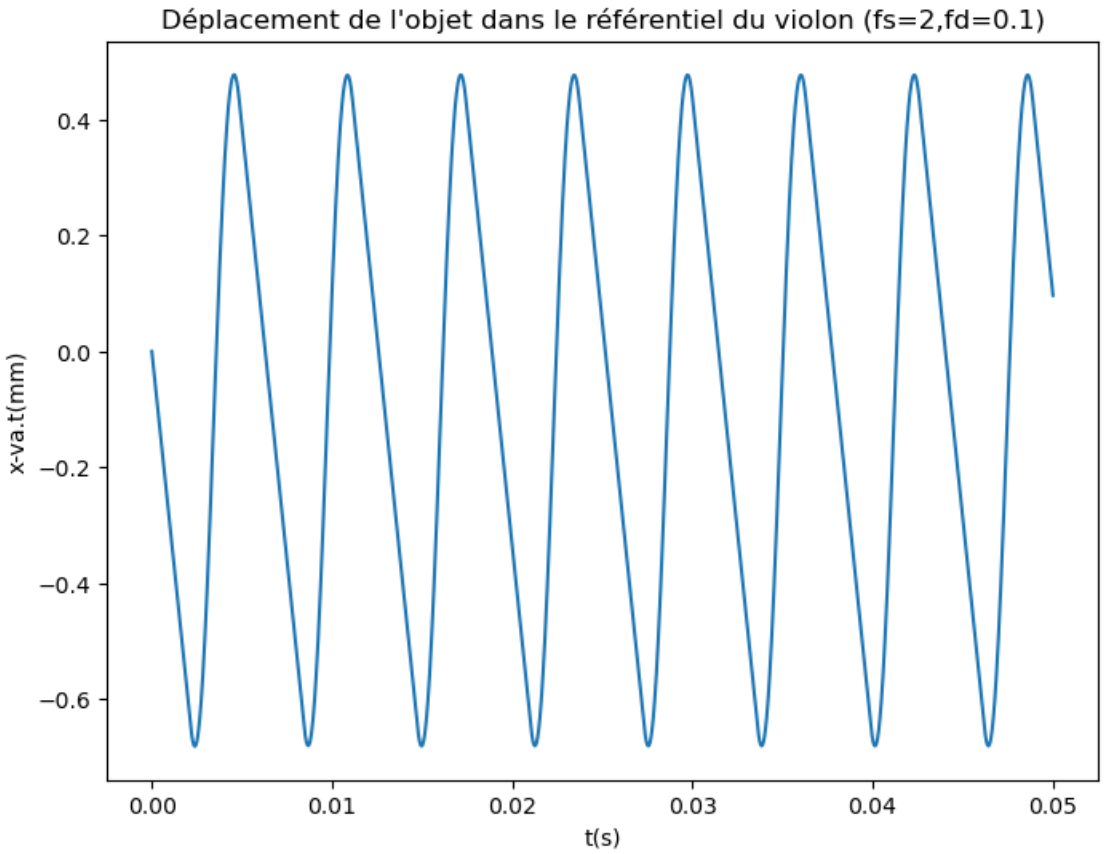
[5]:
```
plt.figure(figsize=(8,6))
plt.plot(tab_t,1000*(tab_x-va*tab_t))
plt.xlabel("t(s)")
plt.ylabel("x-va.t(m)")
plt.title("Déplacement de l'objet dans le référentiel du violon (fs=0.2,fd=0.
↪1)")
plt.show()
```



[6]:
```
def trace_oscillations(va,fs,fd,k,lam,R):
    def acceleration(x,v,t):
        xF=va*t
        Fressort=k*(xF-x)
        Fair=-lam*v
        if v>0:
            T=-fd*R
            return (Fressort+Fair+T)/m
        elif v<0:
            T=+fd*R
            return (Fressort+Fair+T)/m
        else:
```

```
            if abs(Fressort+Fair)<fs*R:
                return 0
            else:
                T=-np.sign(Fressort+Fair)*fs*R
                return (Fressort+Fair+T)/m
    pas=1e-5
    tmax=0.05
    N=int(tmax/pas)
    tab_t=np.zeros(N+1)
    tab_x=np.zeros(N+1)
    tab_v=np.zeros(N+1)
    tab_t[0],tab_x[0],tab_v[0]=0,0,0
    for n in range(N):
        tab_t[n+1]=tab_t[n]+pas
        tab_x[n+1]=tab_x[n]+pas*tab_v[n]
        tab_v[n+1]=tab_v[n]+pas*acceleration(tab_x[n],tab_v[n],tab_t[n])
        if tab_v[n]*tab_v[n+1]<0:
            tab_v[n+1]=0
    plt.plot(tab_t,1000*(tab_x-va*tab_t))
    plt.xlabel("t(s)")
    plt.ylabel("x-va.t(mm)")
    plt.title("Déplacement de l'objet dans le référentiel du violon␣
↪(fs="+str(fs)+",fd="+str(fd)+")")
```

[7]:
```
plt.figure(figsize=(8,6))
trace_oscillations(30e-2,2,0.1,2500,0.2,0.8)
plt.show()
```


Déplacement de l'objet dans le référentiel du violon (fs=2,fd=0.1)
x-va.t(mm)
0.4
0.2
0.0
−0.2
−0.4
−0.6
0.00
0.01
0.02
0.03
0.04
0.05
t(s)

[ ]:

[ ]: