Themodynamique
Corrigé Série Supplémentaire 1:
Déphasage Thermique

S. Guinchard*
Section de Physique, Ecole Polytechnique Fédérale de Lausanne, Lausanne, Suisse
(Supervised by Prof. J.P. Ansermet)†
(Dated: April 6, 2022)

I. EXERCICE 1: DÉPHASAGE THERMIQUE ENTRE DEUX ISOLANTS

A. Questions analytiques

1. Pour une puissance thermique P_Q quelconque, déterminer le système d'équations différentielles couplées qui décrit l'évolution des températures T_i $i = 1, 2$ des deux sous-systèmes.

   Solution: En appliquant le premier principe aux deux sous-systèmes en l'absence d'action mécanique, $P_W = 0$, compte tenu du fait que le transfert de chaleur vers le sous système 1 est décrit par la somme de la puissance thermique P_Q exercée par l'environnement et la puissance thermique exercée par le sous système 2 $P_Q^{(21)}$, et que le transfert de chaleur vers le sous système 2 est décrit par $P_Q^{(12)}$, on a que les dérivées temporelles des énergies internes des deux sous-systèmes s'écrivent:

$$\begin{aligned} \dot{U}_1 &= C_1\dot{T}_1 = P_Q^{(21)} + P_Q, \\ \dot{U}_2 &= C_2\dot{T}_2 = P_Q^{(12)}. \end{aligned} \tag{1}$$

   Compte tenu de la loi de Fourier discrète, les puissances thermiques décrivant le transfert de chaleur à travers la paroi diatherme de surface A, d'épaisseur l et de conductivité thermique κ entre les deux sous systèmes s'écrivent:

$$P_Q^{(12)} = P_Q^{(21)} = -\kappa\frac{A}{l}(T_2 - T_1) \tag{2}$$

   Par conséquent, le système d'équations différentielles couplées décrivant l'évolution de température s'écrit

$$\begin{aligned} \dot{T}_1 &= \frac{P_Q}{C_1} + \frac{\kappa}{C_1}\frac{A}{l}(T_2 - T1), \\ \dot{T}_2 &= -\frac{\kappa}{C_2}\frac{A}{l}(T_2 - T1). \end{aligned} \tag{3}$$

2. Dans le cas particulier où il n'y a pas de transfert de chaleur avec l'environnement ($P_Q \equiv 0$), déterminer explicitement l'évolution temporelle de la différence de température $T_2(t) - T_1(t)$.

   Solution: En absence de transfert de chaleur périodique avec l'environnement, $P_Q \equiv 0$, la différence entre les deux équations d'évolution des températures Eq.(3) s'écrit,

* salomon.guinchard@epfl.ch
† Laboratoire de Physique des Matériaux Nanostructurés, Ecole Polytechnique Fédérale de Lausanne, Lausanne, Suisse

$$\dot{T}_2 - \dot{T}_1 = -\kappa\Big(\frac{1}{C_1} + \frac{1}{C_2}\Big)\frac{A}{l}(T_2 - T_1). \tag{4}$$

En multipliant Eq.(4) par dt et en la divisant par $T_2 - T_1$, on obtient

$$\frac{d\Big(T_2 - T_1\Big)}{T_2 - T_1} = -\kappa\Big(\frac{1}{C_1} + \frac{1}{C_2}\Big)\frac{A}{l}dt, \tag{5}$$

qui est une équation logarithmique. En définissant le temps d'amortissement τ comme

$$\tau = \frac{l}{\kappa A}\Big(\frac{1}{C_1} + \frac{1}{C_2}\Big)^{-1} = \frac{l}{\kappa A}\Big(\frac{C_1 C_2}{C_1 + C_2}\Big) \tag{6}$$

et en intégrant sur la période $t' \in [0, t]$, on obtient:

$$\int_{T_2(0)-T_1(0)}^{T_2(t)-T_1(t)} \frac{d\Big(T_2(t') - T_1(t')\Big)}{T_2(t') - T_1(t')} = -\frac{1}{\tau}\int_0^t dt'. \tag{7}$$

En prenant l'exponentielle du résultat précédent, on obtient le résultat désiré:

$$T_2(t) - T_1(t) = \Big(T_2(0) - T_1(0)\Big)\exp\Big(-\frac{t}{\tau}\Big). \tag{8}$$

Ainsi, en l'absence de transfert de chaleur avec l'environnement, la différence de température entre les deux sous-systèmes décroit exponentiellement. Les deux sous systèmes se thermalisent.

3. Toujours dans le cas sans transfert de chaleur ($P_Q \equiv 0$), déterminer le taux de production d'entropie Π_S.

Solution: Compte tenu de la relation de Gibbs, et de l'équation d'évolution appliquée à chaque sous système fermé, de volume constant, en absence d'action mécanique $P_W \equiv 0$, la dérivée temporelle de l'entropie de chaque sous-système s'écrit,

$$\begin{aligned} \dot{S}_1 &= \frac{\dot{U}_1}{T_1} = \frac{P_Q^{(21)}}{T_1} = \kappa\frac{A}{l}\frac{T_2 - T_1}{T_1}, \\ \dot{S}_2 &= \frac{\dot{U}_2}{T_2} = \frac{P_Q^{(12)}}{T_2} = -\kappa\frac{A}{l}\frac{T_2 - T_1}{T_2} \end{aligned} \tag{9}$$

En tenant compte du deuxième principe pour un système adiabatiquement fermé et de l'extensivité de l'entropie, le taux de production d'entropie est donné par

$$\Pi_S = \dot{S}_1 + \dot{S}_2 = \kappa\frac{A}{l}\Big(\frac{1}{T_1} - \frac{1}{T_2}\Big)(T_2 - T_1) = \kappa\frac{A}{l}\frac{(T_2 - T_1)^2}{T_1 T_2} \geq 0. \tag{10}$$

4. Dans le régime harmonique dû à un transfert de chaleur périodique avec l'ensemble de l'environnement, en écrivant le système d'équations d'évolution couplées sous forme matricielle, en déduire le module du rapport des amplitudes complexes des oscillations de températures: $|\Delta T_2/\Delta T|$ et l'angle de déphasage $\Delta\phi = \phi_2 - \phi_1$ entre ces amplitudes complexes ΔT_i.

Solution: Comme les températures complexes des deux sous-systèmes s'écrivent

$$T_j(t) = \Delta T_j \exp(i\omega t) + T_0 \qquad j = 1, 2, \tag{11}$$

leurs dérivées temporelles sont données par

$$\dot{T}_j(t) = i\omega \Delta T_j \exp(i\omega t) \qquad j = 1, 2. \tag{12}$$

Pour une puissance thermique périodique (et complexe),

$$P_Q(t) = P_0 \exp(i\omega t), \tag{13}$$

le système d'équations Eq.(8) devient

$$\begin{aligned} i\omega \Delta T_1 \exp(i\omega t) &= \frac{P_0}{C_1} \exp(i\omega t) + \frac{\kappa}{C_1}\frac{A}{l}(\Delta T_2 - \Delta T_1)\exp(i\omega t), \\ i\omega \Delta T_2 \exp(i\omega t) &= -\frac{\kappa}{C_2}\frac{A}{l}(\Delta T_2 - \Delta T_1)\exp(i\omega t). \end{aligned} \tag{14}$$

Le système peut ainsi se réécrire

$$\begin{aligned} \left(i\omega + \frac{\kappa}{C_1}\frac{A}{l}\right)\Delta T_1 - \frac{\kappa}{C_1}\frac{A}{l}\Delta T_2 &= \frac{P_0}{C_1}, \\ -\frac{\kappa}{C_2}\frac{A}{l}\Delta T_1 + \left(i\omega + \frac{\kappa}{C_2}\frac{A}{l}\right)\Delta T_2 &= 0. \end{aligned} \tag{15}$$

En mettant le résultat précédent en forme matricielle on obtient

$$\begin{pmatrix} i\omega + \frac{\kappa}{C_1}\frac{A}{l} & -\frac{\kappa}{C_1}\frac{A}{l} \\ -\frac{\kappa}{C_2}\frac{A}{l} & i\omega + \frac{\kappa}{C_2}\frac{A}{l} \end{pmatrix} \begin{pmatrix} \Delta T_1 \\ \Delta T_2 \end{pmatrix} = \begin{pmatrix} P_0 \\ 0 \end{pmatrix}, \tag{16}$$

d'où on tire en inversant la matrice, que

$$\Delta T_1 = \frac{i\omega + \frac{\kappa}{C_2}\frac{A}{l}}{-\omega^2 + i\omega\kappa\frac{A}{l}\left(\frac{1}{C_1} + \frac{1}{C_2}\right)} P_0, \tag{17}$$

$$\Delta T_2 = \frac{\frac{\kappa}{C_2}\frac{A}{l}}{-\omega^2 + i\omega\kappa\frac{A}{l}\left(\frac{1}{C_1} + \frac{1}{C_2}\right)} P_0. \tag{18}$$

La rapport des deux amplitudes vaut

$$\frac{\Delta T_1}{\Delta T_2} = \frac{i\omega + \frac{\kappa}{C_2}\frac{A}{l}}{\frac{\kappa}{C_2}\frac{A}{l}}. \tag{19}$$

Pour le module du rapport, on obtient

$$\left|\frac{\Delta T_1}{\Delta T_2}\right| = \frac{\sqrt{\omega^2 + \frac{\kappa^2}{C_2^2}\frac{A^2}{l^2}}}{\frac{\kappa}{C_2}\frac{A}{l}} = \sqrt{1 + \frac{\omega^2 l^2}{\kappa^2 A^2}} > 1. \tag{20}$$

Ainsi, l'amplitude des oscillations de température est plus forte dans le sous système 1 que dans le sous système 2. Cela signifie que l'isolation amortit les fluctuations de température à l'intérieur du bâtiment, comme on le comprend intuitivement. Le rapport des amplitudes complexes s'écrit en terme des angles de déphasage de ces oscillations dans le plan complexe, comme,

$$\frac{\Delta T_1}{\Delta T_2} = \frac{|\Delta T_1|}{|\Delta T_2|}\frac{e^{i\phi_1}}{e^{i\phi_2}} = \left|\frac{\Delta T_1}{\Delta T_2}\right| e^{i(\phi_1-\phi_2)} = \sqrt{1 + \frac{\omega^2 l^2}{\kappa^2 A^2}} e^{i\Delta\phi} = \sqrt{1 + \frac{\omega^2 l^2}{\kappa^2 A^2}}\Big(cos(\Delta\phi) + i\sin(\Delta\phi)\Big). \tag{21}$$

En prenant le rapport des parties réelles et imaginaires de l'équation précédente, on obtient pour la tangente de l'angle de déphasage:

$$\tan(\Delta\phi) = \frac{\sin(\Delta\phi)}{\cos(\Delta\phi)} = \frac{\omega l}{\kappa A} C_2. \tag{22}$$

De là, on déduit une expression pour l'angle

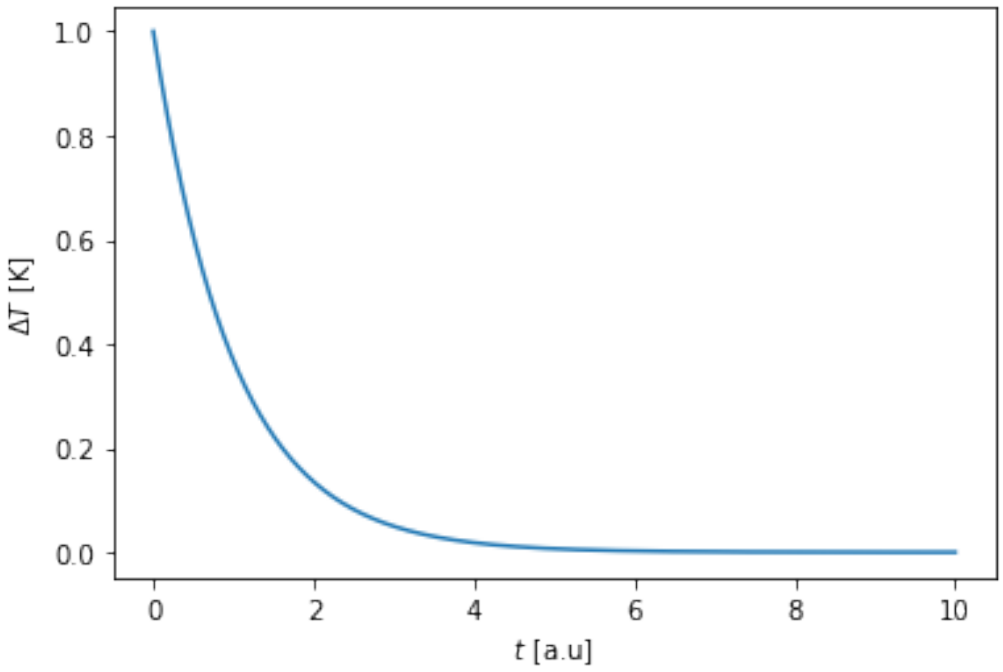
$$\Delta\phi = \arctan\left(\frac{\omega l}{\kappa A} C_2\right). \tag{23}$$

B. Implémentation numérique

1. Implémentez numériquement le résultat de la question 2 et plotter $\Delta T(t)$. Pour cela, complétez le code du Jupyter Notebook associé. Les lignes de code à compléter sont indiquées par les TODO zones. Changez les paramètres physiques, temps caractéristique... Que remarquez vous ?

Solution:

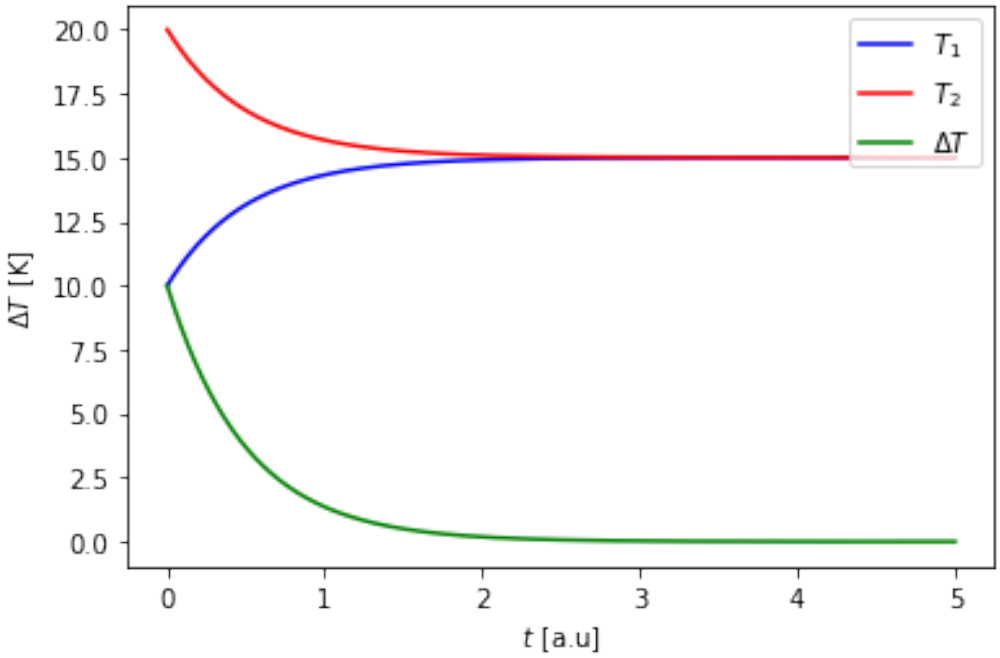
[2]:
```
t = np.linspace(0, 10, 10000);

# TODO: Set arbitrary initial conditions
# Temperatures
T10 = 0;
T20 = 1;
# problem constants
tau=1;

# TODO: Implement and plot Delta T
DeltaT=(T20-T10)*np.exp(-t/tau);
plt.plot(t, DeltaT );
plt.xlabel('$t$ [a.u]');
plt.ylabel('$\Delta T$ [K]');
```



a. Résolution numérique de l'évolution des températures T_1 et T_2

[3]:
```
from array import *
# Paramètres physiques
# TODO: Play with parameters
T1=10;
T2=20;
kappa = -1;
A     = 1;
C1    = 1;
C2    = 1;
l     = 1;

# Paramètres numériques
tfin   = 5;
nsteps = 10000;
dt=tfin/nsteps;

# Initialisation variables temps et température
output = np.zeros((2,nsteps));
arrayT = [T1,T2];
time   = np.zeros((1,nsteps));
time[0][0] = 0.0;
for i in range(1,nsteps):
    time[0][i] = time[0][i-1]+dt;


# Method running Euler method till final time
def run():
    t=0.0;
    for i in range(0, nsteps):
        output[0][i] = arrayT[0];
        output[1][i] = arrayT[1];

        arrayT[0]= arrayT[0] - kappa*A/(C2*l)*(arrayT[1]-arrayT[0])*dt;
```

```
        arrayT[1]= arrayT[1] + kappa*A/(C1*l)*(arrayT[1]-arrayT[0])*dt;
        t+=dt;

run();
# TODO : Plot T1, T2 and T2-T1 = Delta T
plt.plot(time[0], output[0],'b-', label='$T_1$');
plt.plot(time[0], output[1], 'r-', label='$T_2$');
plt.plot(time[0], output[1]-output[0], 'g-', label='$\Delta T$');
plt.xlabel('$t$ [a.u]');
plt.ylabel('$\Delta T$ [K]');
plt.legend(loc="upper right");
```

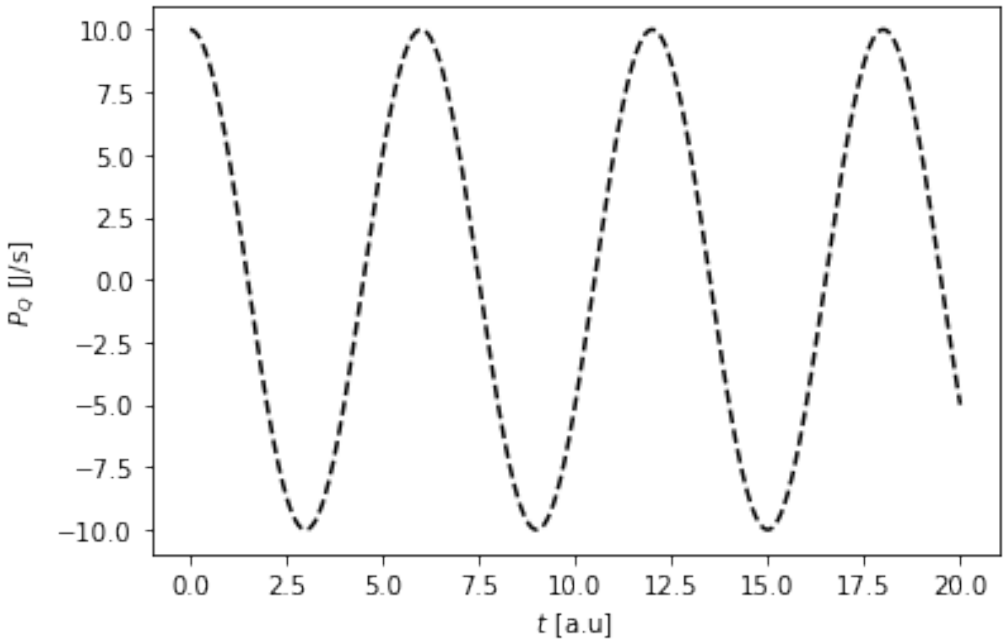


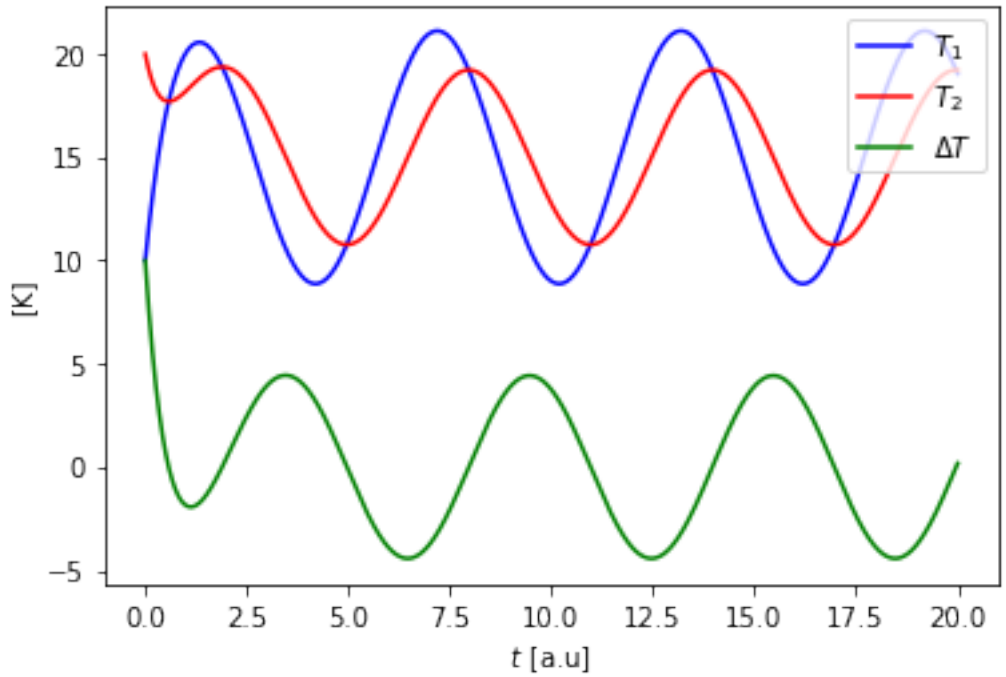
2. Dans le cas d'un transfert de chaleur périodique comme dans la question 4, plottez les températures T_1 et T_2 résolues numériquement par le code de la cellule. Plottez également la différence de températures. Changez les paramètres physiques des sous systèmes 1 et 2, de même que le temps de simulation $t_{fin} \in [10, 50]$. Distinguez le régime transitoire du régime harmonique. Que remarquez vous quant à la période des oscillations ?

Solution:

[4]:
```
from array import *
################################################
# Paramètres physiques
# TODO: Play with parameters
T1=10;
T2=20;
kappa = -1;
A     = 1;
C1    = 1;
C2    = 1;
l     = 1;
K     =10;
```

```
# Paramètres numériques
# TODO: Play with tfin
tfin =20;
nsteps=100000;
dt=tfin/nsteps;

######## DO NOT modify anything below #########
################################################

# Initialisation variables temps et température
output = np.zeros((2,nsteps));
arrayT = [T1,T2];
time = np.zeros((1,nsteps));
time[0][0] = 0.0;

for i in range(1,nsteps):
    time[0][i] = time[0][i-1]+dt;

PQ = np.zeros((1,nsteps));
for i in range(0,nsteps):
    PQ[0][i] = K*np.cos(np.pi/3*time[0][i]);

# Method running Euler method till final time
def run():
    t=0.0;
    for i in range(0, nsteps):
        output[0][i] = arrayT[0];
        output[1][i] = arrayT[1];

        arrayT[0]= arrayT[0] - kappa*A/(C2*l)*(arrayT[1]-arrayT[0])*dt +␣
 ↪PQ[0][i]/C1*dt;
        arrayT[1]= arrayT[1] + kappa*A/(C1*l)*(arrayT[1]-arrayT[0])*dt;
        t+=dt;

run();
# TODO: Plot the temperatures T1 and T2
plt.figure(1)
plt.plot(time[0], PQ[0], 'k--')
plt.xlabel('$t$ [a.u]');
plt.ylabel('$P_Q$ [J/s]');
plt.ticklabel_format(axis='y', style='sci');

plt.figure(2)
plt.plot(time[0], output[0],'b-', label= '$T_1$');
plt.plot(time[0], output[1], 'r-', label='$T_2$');
plt.plot(time[0], output[1]-output[0], 'g-', label='$\Delta T$');
plt.xlabel('$t$ [a.u]');
plt.ylabel('[K]');
plt.legend(loc="upper right");
```





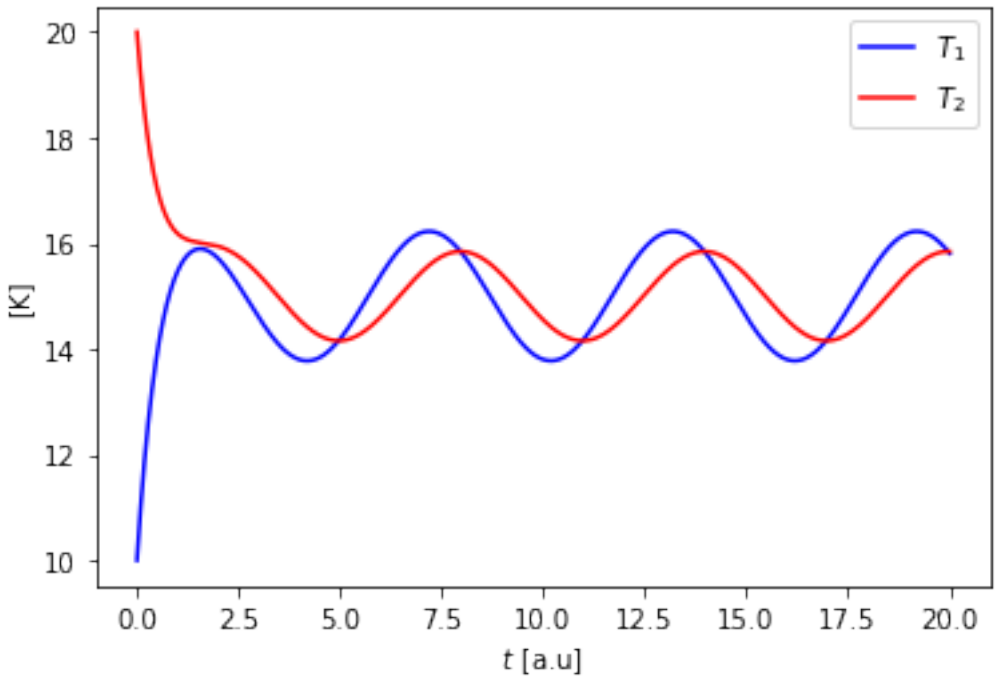
Remarque: Noter que les oscillations de T_2 se font effectivement avec la même période que T_1 (et P_Q), mais à moindre amplitude, ce qui confirme le résultat analytique obtenu précédemment.

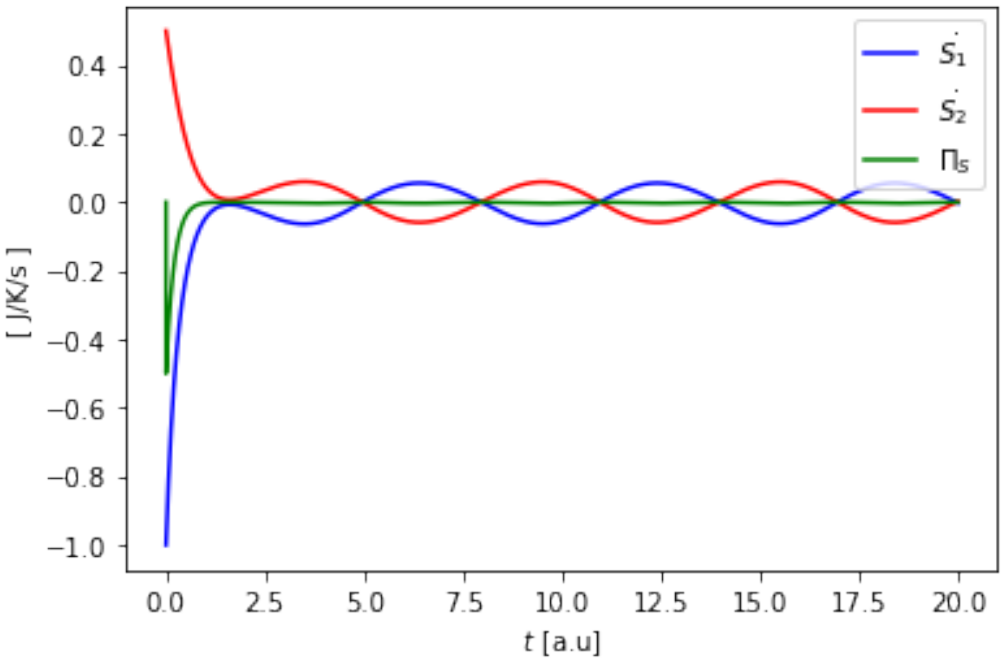
3. Dans le cas du transfert de chaleur périodique, plottez le taux de production d'entropie. Commentez le résultat obtenu.

Solution:

[5]:
```
from array import *

# Paramètres physiques
# TODO: Play with parameters
T1=10;
T2=20;
kappa = -1;
A     = 1;
C1    = 1;
C2    = 1;
l     = 1;
K     = 2;

# initialise \dot(S1) et \dot(S2) à t=0.0

S1 = kappa*A/l*(T2-T1)/T1;
S2 = -kappa*A/l*(T2-T1)/T2;

# Paramètres numériques
tfin =20;
nsteps=100000;
dt=tfin/nsteps;

# Initialisation variables temps et température
outputT = np.zeros((2,nsteps));
outputS = np.zeros((2,nsteps));
arrayT  = [T1,T2];   # Stocke les températures T
arrayS  = [S1,S2];   # stocke les dérivées de S
PiS     = np.zeros((1,nsteps));
time    = np.zeros((1,nsteps));
time[0][0] = 0.0;
for i in range(1,nsteps):
    time[0][i] = time[0][i-1]+dt;

PQ = np.zeros((1,nsteps));
for i in range(0,nsteps):
    PQ[0][i] = K*np.cos(np.pi/3*time[0][i]);

# Method running Euler method till final time
def run():
    t=0.0;
    outputT[0][0] = arrayT[0];
    outputT[1][0] = arrayT[1];

    outputS[0][0] = arrayS[0];
    outputS[1][0] = arrayS[1];

    for i in range(1, nsteps):

        arrayT[0]= arrayT[0] - kappa*A/(C2*l)*(arrayT[1]-arrayT[0])*dt +␣
↪PQ[0][i]/C1*dt;
        arrayT[1]= arrayT[1] + kappa*A/(C1*l)*(arrayT[1]-arrayT[0])*dt;

        outputT[0][i] = arrayT[0];
        outputT[1][i] = arrayT[1];
```

```
        outputS[0][i] = kappa*A/l*(arrayT[1]-arrayT[0])/arrayT[0];
        outputS[1][i] = -kappa*A/l*(arrayT[1]-arrayT[0])/arrayT[1];

        # To get S: integrate \dot(S), that is arrayS
        #arrayS[0] = arrayS[0] + kappa*A/(l)*(arrayT[1]-arrayT[0])/arrayT[0]*dt
        #arrayS[1] = arrayS[1] - kappa*A/(l)*(arrayT[1]-arrayT[0])/arrayT[1]*dt;

        PiS[0][i] = outputS[0][i] +  outputS[1][i];
        #TODO:Check equality (plot difference eventually)
        #kappa*A/l*(arrayT[1]-arrayT[0])**2/(arrayT[1]*arrayT[0]);

        t+=dt;

run();


# TODO : Plot the entropy production rate
plt.figure(1)
plt.plot(time[0], outputT[0], 'b-', label='$T_1$');
plt.plot(time[0], outputT[1], 'r-', label='$T_2$');
plt.xlabel('$t$ [a.u]');
plt.ylabel('[K]');
plt.legend(loc="upper right");

plt.figure(2)
plt.plot(time[0], outputS[0],'b-', label='$\dot{S_1}$');
plt.plot(time[0], outputS[1], 'r-', label='$\dot{S_2}$');
plt.plot(time[0], PiS[0], 'g-', label='$\Pi_S$');
plt.xlabel('$t$ [a.u]');
plt.ylabel('[ J/K/s ]');
plt.legend(loc="upper right");
```




$\dot{S}_1$
$\dot{S}_2$
Π_S
0.4
0.2
0.0
−0.2
−0.4
−0.6
−0.8
−1.0
[J/K/s]
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0
t [a.u]