Détermination d'une capacité thermique et d'une capacité thermique massique - CORRIGE

I- CAPACITE THERMIQUE C_{CAL} D'UN CALORIMETRE

1- Le calorimètre n'étant pas idéal, ses parois internes réalisent des échanges thermiques avec l'eau froide et l'eau chaude. Ainsi, si on étudiait le système {eau froide + eau chaude}, on n'aurait aucun moyen de calculer les transferts thermiques Q réalisés par ce système avec l'extérieur, dont les parois du calorimètre feraient alors partie.

En incluant le calorimètre dans le système, les transferts thermiques Q entre le système et l'extérieur deviennent nuls puisque les parois du calorimètre sont supposées parfaitement calorifugées (les échanges thermiques réalisés entre l'eau et les parois internes du calorimètre ne sont en effet plus à considérer car ce sont des échanges d'énergie entre parties du système et non avec l'extérieur de celui-ci).

2- La transformation subie par le système, fermé, est monobare car la pression extérieure est constante. D'autre part, le système est uniquement constitué de phases condensées (liquide et solide) pour lesquelles on peut considérer que la pression est la même que la pression extérieure, ce qui fait qu'on peut considérer que le système est en équilibre mécanique dans l'état initial et dans l'état final. On peut donc appliquer le premier principe de la thermodynamique sous la forme d'un bilan d'enthalpie :

$$\Delta E_m + \Delta H = W' + Q.$$

Avec : ΔE_m = variation d'énergie mécanique macroscopique du système **= 0** car le système est macroscopiquement au repos ;
ΔH = variation d'enthalpie du système ;
W' = travaux (des forces autres que celles de pression) algébriquement reçus par le système **= 0** car il n'y a pas d'autres forces que celles de pression qui agissent sur le système ;
Q = transferts thermiques algébriquement reçus par le système **= 0** car le calorimètre est parfaitement calorifugé, empêchant tout transfert thermique avec l'extérieur.

Donc **ΔH = 0**.

Or, l'enthalpie est une grandeur extensive, donc la variation d'enthalpie **ΔH** du système est égale à la somme des variations d'enthalpie de chaque partie du système :

$$\Delta H = \Delta H(\text{eau froide}) + \Delta H(\text{eau chaude}) + \Delta H(\text{calorimètre})$$

$$\boxed{0 = m_1 \times c_{eau} \times (T_F - T_1) + m_2 \times c_{eau} \times (T_F - T_2) + C_{cal} \times (T_F - T_1)}$$

3- On en déduit l'expression suivante de C_{cal} :

$$\boxed{C_{cal} = c_{eau} \times \frac{m_1 \times (T_F - T_1) + m_2 \times (T_F - T_2)}{T_1 - T_F}}$$

$m_1 = 100$ g ; $m_2 = 100$ g
$T_1 = 20{,}0$ °C ; $T_2 = 60{,}0$ °C
$T_F = 35{,}0$ °C

AN → $C_{cal} = 4180 \times \dfrac{0,100 \times (35-20) + 0,100 \times (35-60)}{20-35}$ soit $C_{Cal} = 279$ J/K

4- On cherche une masse d'eau **μ**, de capacité thermique massique c_{eau} dont la capacité thermique C_{eau} serait la même que celle du calorimètre C_{cal} ; autrement dit, $C_{cal} = C_{eau}$.

Or, par définition $C_{eau} = \mu \times c_{eau}$. On en déduit donc que $\mu \times c_{eau} = C_{cal}$, soit $\boxed{\mu = C_{cal} / c_{eau}}$.

AN → $\mu = \dfrac{279}{4180}$ soit $\mu = 6,67.10^{-2}$ kg $= 66,7$ g

---

5- • Demi-étendues sur les masses déterminées avec une balance affichant un résultat à 1 g près :

$$a(m_1) = a(m_2) = 0{,}001 \text{ kg}$$

• Demi-étendues sur les températures T_1 et T_F relevées graphiquement à l'aide de LatisPro :

$$a(T_1) = a(T_F) = 0{,}3 \text{ °C}$$

• Demi-étendues sur la température T_2 relevée à l'aide d'un thermomètre affichant un résultat à 0,1 °C près :

$$a(T_2) = 0{,}1 \text{ °C}$$

6-

```
# Importation des bibliothèques utiles :
import numpy as np                # Pour faire des calculs, des tableaux …
import matplotlib.pyplot as plt   # Pour tracer des graphiques
import numpy.random as rd         # Pour générer des nombres aléatoires sous l'alias rd

# Définition des grandeurs physiques :
```

```
m1 = 0.100                                   # Renseigner la valeur de la masse d'eau froide introduite (en kg)
m2 = 0.100                                   # Renseigner la valeur de la masse d'eau chaude introduite (en kg)
T1 = 20                                      # Renseigner  la valeur de la température initiale de l'eau froide et du calorimètre (en °C)
T2 = 60                                      # Renseigner  la valeur de la température initiale de l'eau chaude (en °C)
TF = 35                                      # Renseigner  la valeur de la température finale du système (en °C)
c_eau = 4180                                 # Renseigner la valeur de la capacité thermique massique de l'eau (en J/K/kg)

# Définition des demi-étendues estimées :
a_m1 = 0.001                                 # Renseigner la valeur de la demi-étendue estimée pour la masse m1 (en kg)
a_m2 = 0.001                                 # Renseigner la valeur de la demi-étendue estimée pour la masse m2 (en kg)
a_T1 = 0.3                                   # Renseigner la valeur de la demi-étendue estimée pour la température T1 (en °C)
a_T2 = 0.1                                   # Renseigner la valeur de la demi-étendue estimée pour la température T2 (en °C)
a_TF = 0.3                                   # Renseigner la valeur de la demi-étendue estimée pour la température TF (en °C)

# Saisie du nombre N d'expériences numériques à simuler pour la méthode de Monte-Carlo :
N = 1000                                     # Renseigner le nombre d'expériences à simuler

# Tirage de valeurs des grandeurs physiques mesurées :
m1s=rd.uniform(m1-a_m1,m1+a_m1,N)            # Formule qui permet de simuler N valeurs de m1 dans l'intervalle [m1 – a(m1) ; m1 + a(m1)]
m2s=rd.uniform(m2-a_m2,m2+a_m2,N)            # Formule qui permet de simuler N valeurs de m2 dans l'intervalle [m2 – a(m2) ; m2 + a(m2)]
T1s=rd.uniform(T1-a_T1,T1+a_T1,N)            # Formule qui permet de simuler N valeurs de T1 dans l'intervalle [T1 – a(T1) ; T1 + a(T1)]
T2s=rd.uniform(T2-a_T2,T2+a_T2,N)            # Formule qui permet de simuler N valeurs de T2 dans l'intervalle [T2 – a(T2) ; T2 + a(T2)]
TFs=rd.uniform(TF-a_TF,TF+a_TF,N)            # Formule qui permet de simuler N valeurs de TF dans l'intervalle [TF – a(TF) ; TF + a(TF)]

# Formule pour calculer la capacité thermique C_cal du calorimètre à partir de c_eau et des valeurs simulées m1s, m2s, T1s, T2s et TFs :
C_cal = c_eau*(m1s*(TFs-T1s)+m2s*(TFs-T2s))/(T1s-TFs)

# Formule qui calcule la valeur moyenne de C_cal et son incertitude à partir des N expériences numériques aléatoires :
C_cal_moy=np.mean(C_cal)                     # Valeur moyenne des N valeurs simulées de C_cal
u_C_cal=np.std(C_cal,ddof=1)                 # Ecart-type des N valeurs simulées de C_cal

# Formule qui permet d'afficher la valeur moyenne de C_cal et son incertitude :
print('C_cal_moy=',C_cal_moy,'J/K')          # Affiche la valeur moyenne des N valeurs simulées de C_cal avec son unité
print('u(C_cal)=',u_C_cal,'J/K')             # Affiche l'écart-type des N valeurs simulées de C_cal avec son unité
```

✍**7-** L'exécution du programme Python conduit au résultat suivant :

```
C_cal_moy= 279.3325066207917 J/K
u(C_cal)= 16.366965098703332 J/K
```

On conclue donc en écrivant **C_{cal} = (279 +/- 17) J.K^{-1}**

II- CAPACITE THERMIQUE C_S D'UN SOLIDE

✍**8-** Par analogie avec la partie **I-**, il est ici conseillé d'étudier le système {eau + solide + calorimètre}.

✍**9-** De même que dans la partie I-, pour le système {eau + solide + calorimètre}, on a la relation :

$$\Delta H = \Delta H(\text{eau}) + \Delta H(\text{solide}) + \Delta H(\text{calorimètre})$$

$$0 = m_3 \times c_{eau} \times (T_F - T_3) + m_C \times c_S \times (T_F - T_C) + C_{cal} \times (T_F - T_3)$$

✍**10-** On en déduit l'expression suivante de c_S :

$$\boxed{c_S = -\frac{(m_3 \times c_{eau} + C_{cal}) \times (T_F - T_3)}{m_C \times (T_F - T_C)}}$$

m_3 = 72 g ; m_C = 58 g
T_3 = 28,0 °C ; T_C = 66,0 °C
T_F = 30,0 °C **(aluminium)**

$$\text{AN} \rightarrow \quad c_S = -\frac{(0,200 \times 4180 + 279) \times (30 - 28)}{0,072 \times (30 - 66)}$$

soit $c_S = \mathbf{860\ J.K^{-1}.kg^{-1}}$

✍**11-**

```
# Importation des bibliothèques utiles
import numpy as np                  # Pour faire des calculs, des tableaux …
import matplotlib.pyplot as plt     # Pour tracer des graphiques
import numpy.random as rd           # Pour générer des nombres aléatoires sous l'alias rd

# Définition des grandeurs physiques :
m3 = 0.200                          # Renseigner la valeur de la masse d'eau présente dans le calorimètre (en kg)
mc = 0.072                          # Renseigner la valeur de la masse du cylindre métallique utilisé (en kg)
```

```python
T3 = 28                    # Renseigner  la valeur de la température initiale de l'eau et du calorimètre (en °C)
Tc = 66                    # Renseigner  la valeur de la température initiale du cylindre métallique (en °C)
TF = 30                    # Renseigner  la valeur de la température finale du système (en °C)
c_eau = 4180               # Renseigner la valeur de la capacité thermique massique de l'eau (en J/K/kg)
C_calo = 279               # Renseigner la valeur de la capacité thermique du calorimètre (en J/K)
uC_calo = 17               # Renseigner la valeur de l'incertitude obtenue sur la capacité thermique du calorimètre dans la partie I- (en J/K)

# Définition des demi-étendues estimées :
a_m3= 0.001                # Renseigner la valeur de la demi-étendue estimée pour la masse m3 d'eau (en kg)
a_mc = 0.001               # Renseigner la valeur de la demi-étendue estimée pour la masse mc du cylindre (en kg)
a_T3 = 0.3                 # Renseigner la valeur de la demi-étendue estimée pour la température T3 (en °C)
a_Tc = 0.1                 # Renseigner la valeur de la demi-étendue estimée pour la température Tc (en °C)
a_TF = 0.3                 # Renseigner la valeur de la demi-étendue estimée pour la température TF (en °C)
a_C_calo = uC_calo*np.sqrt(3)   # Renseigner la valeur de la demi-étendue estimée pour la capacité thermique C(calo) du calorimètre (en J/K)

# Saisie du nombre N d'expériences numériques à simuler pour la méthode de Monte-Carlo :
N = 1000                   # Renseigner le nombre d'expériences à simuler

# Tirage de valeurs des grandeurs physiques mesurées :
m3s=rd.uniform(m3-a_m3,m3+a_m3,N)      # Formule qui permet de simuler N valeurs de m3 dans l'intervalle [m3 – a(m3) ; m3 + a(m3)]
mcs=rd.uniform(mc-a_mc,mc+a_mc,N)      # Formule qui permet de simuler N valeurs de mc dans l'intervalle [mc – a(mc) ; mc + a(mc)]
T3s=rd.uniform(T3-a_T3,T3+a_T3,N)      # Formule qui permet de simuler N valeurs de T3 dans l'intervalle [T3 – a(T3) ; T3 + a(T3)]
Tcs=rd.uniform(Tc-a_Tc,Tc+a_Tc,N)      # Formule qui permet de simuler N valeurs de Tc dans l'intervalle [Tc – a(Tc) ; Tc + a(Tc)]
TFs=rd.uniform(TF-a_TF,TF+a_TF,N)      # Formule qui permet de simuler N valeurs de TF dans l'intervalle [TF – a(TF) ; TF + a(TF)]
C_calos=rd.uniform(C_calo-a_C_calo, C_calo+a_C_calo,N)   # Formule qui permet de simuler N valeurs de C(calo) dans l'intervalle
                                                         [Ccalo – a(Ccalo) ; Ccalo + a(Ccalo)]

# Expression de la capacité thermique massique c_s du solide à partir de c_eau et des valeurs simulées m3s, mcs, T3s, Tcs TFs et C_calos :
c_s = -(m3s*c_eau+C_calos)*(TFs-T3s)/(mcs*(TFs-TcS))

# Formule qui calcule la valeur moyenne de c_s et son incertitude à partir des N expériences numériques aléatoires :
c_s_moy=np.mean(c_s)                   # Valeur moyenne des N valeurs simulées de c_s
u_c_s=np.std(c_s,ddof=1)               # Ecart-type des N valeurs simulées de c_s

# Formule qui permet d'afficher la valeur moyenne de c_s et son incertitude :
print('c_s_moy=',c_s_moy,'J/K/kg')     # Affiche la valeur moyenne des N valeurs simulées de c_s avec son unité
print('u(c_s)=',u_c_s,'J/K/kg')        # Affiche l'écart-type des N valeurs simulées de c_s avec son unité
```

✍**12-** L'exécution du programme Python conduit au résultat suivant :

```
c_s_moy= 862.4383693612691 J/K/kg
u(c_s)= 107.96210203303023 J/K/kg
```

On conclue donc en écrivant **c_S = (8,6 +/- 1,1).10^2 J. K^{-1}.kg^{-1}**

✍**13-** La capacité thermique massique théorique de l'aluminium vaut 897 J.K^{-1}.kg^{-1}. Pour comparer cette valeur théorique à la valeur obtenue expérimentalement, on peut calculer l'écart normalisé entre ces deux valeurs, ce qui conduit à :

$$\mathbf{EN} = \frac{|\mathbf{c_S(exp)} - \mathbf{c_S(théo)}|}{\sqrt{\mathbf{u(c_S(exp))^2} + \mathbf{u(c_S(théo))^2}}} \qquad soit\ \mathbf{EN} = \frac{|\mathbf{c_s(exp)} - \mathbf{c_S(théo)}|}{\mathbf{u(c_S(exp))}}$$

car il n'y a à priori pas d'incertitude sur la capacité massique thermique théorique, ou, du moins, elle est probablement négligeable par rapport à l'incertitude sur la capacité thermique massique expérimentale.

$$\mathbf{EN} = \frac{|8,6.10^2 - 897|}{1.10^2} = \mathbf{0,4}$$

▶ Cet écart normalisé **inférieur à 2** témoigne d'un **bon accord entre la valeur expérimentale et la valeur théorique**.