Concours e3a-Polytech, France

Session 2021, filière PC

Modélisation de systèmes physiques ou chimiques

Durée : 4 heures

Ce que traite le sujet

- Partie I : fuite d'un réservoir, mise en équation
- Partie II : fuite d'un réservoir, traitement numérique
- Partie III : corrélations pour la capacité thermique

Sujet reproduit sans retouche à partir de la page suivante.

Résumé des parties, chapitres de cours liés et autres années :
prepahelp.com/annales/e3a/pc/2021/modelisation

Prepa Help ne revendique aucun droit sur le sujet, propriété de l'organisateur du concours.

ÉPREUVE MUTUALISÉE AVEC E3A-POLYTECH

ÉPREUVE SPÉCIFIQUE - FILIÈRE PC

MODÉLISATION DE SYSTÈMES PHYSIQUES OU CHIMIQUES

Durée : 4 heures

N.B. : le candidat attachera la plus grande importance à la clarté, à la précision et à la concision de la rédaction. Si un candidat est amené à repérer ce qui peut lui sembler être une erreur d'énoncé, il le signalera sur sa copie et devra poursuivre sa composition en expliquant les raisons des initiatives qu'il a été amené à prendre.

RAPPEL DES CONSIGNES

- *Utiliser uniquement un stylo noir ou bleu foncé non effaçable pour la rédaction de votre composition ; d'autres couleurs, excepté le vert, peuvent être utilisées, mais exclusivement pour les schémas et la mise en évidence des résultats.*
- *Ne pas utiliser de correcteur.*
- *Écrire le mot FIN à la fin de votre composition.*

Les calculatrices sont autorisées.

Le sujet est composé de trois parties indépendantes.

Sujet : page 1 à page 12.
Annexe : page 13 à page 16.

Modélisation de la fuite de matière d'un réservoir rempli de dioxyde de carbone gazeux

Présentation générale

Ce sujet repose sur l'étude théorique et numérique d'une fuite de matière au sein d'une cuve contenant du dioxyde de carbone (CO_2) gazeux. Il est constitué de **trois parties indépendantes**.

- La première partie est dédiée à l'établissement du modèle thermodynamique du phénomène de fuite de matière contenue dans une cuve à travers un orifice.
- La seconde partie est consacrée à l'étude numérique du problème ; la relation entre la capacité thermique molaire $C_{P,m}$ du CO_2 gaz parfait et la température T est admise.
- La troisième partie est consacrée à l'étude de deux modèles pour décrire la relation entre $C_{P,m}$ et T dont les coefficients sont déduits de mesures expérimentales.

Traitement numérique et calcul scientifique réalisés à partir d'un programme écrit en langage Python

- Les programmes demandés au candidat seront réalisés dans le langage Python.
- On veillera à apporter les commentaires facilitant la compréhension du programme et à utiliser des noms de variables explicites.
- **Il est demandé de répondre précisément aux questions posées (par exemple, on écrira une fonction uniquement lorsque cela est explicitement demandé).**
- ***Une annexe décrivant quelques éléments de langage Python utiles pour ce sujet est fournie en page 13.***

Précisions concernant les notations utilisées

Symbole	Nom	Unité
$C_{P,m}$	Capacité thermique **molaire** à pression constante	$J{\cdot}K^{-1}{\cdot}mol^{-1}$
c_P	Capacité thermique **massique** à pression constante	$J{\cdot}K^{-1}{\cdot}kg^{-1}$
D_m	Débit **massique**	$kg{\cdot}s^{-1}$
h	Enthalpie **massique**	$J{\cdot}kg^{-1}$
M	Masse **molaire**	$kg{\cdot}mol^{-1}$
m	Masse	kg
P	Pression	Pa
$\dot{Q} = \delta Q/\mathrm{d}t$	Puissance thermique	W
R	Constante des gaz parfaits	$J{\cdot}K^{-1}{\cdot}mol^{-1}$
T	Température	K
T_c	Température critique	K
U	Énergie interne	J
u	Énergie interne **massique**	$J{\cdot}kg^{-1}$
V	Volume	m^3
$\dot{W} = \delta W/\mathrm{d}t$	Puissance utile	W
ω	Vitesse des courants de matière	$m{\cdot}s^{-1}$
Ω	Section de fuite	m^2

Partie I - Modélisation de la fuite d'un réservoir : mise en équation

Généralités sur les bilans de matière et d'énergie en système ouvert

On s'intéresse au système ouvert décrit par la **figure 1**. Ce système possède une entrée de matière (notée A), une sortie (notée B). Il reçoit du milieu extérieur une puissance thermique $\dot{Q}_e$ et une puissance de force $\dot{W}_e$. Il fournit au milieu extérieur une puissance thermique $\dot{Q}_s$ et une puissance de force $\dot{W}_s$. Les grandeurs $\dot{Q}_e$, $\dot{Q}_s$, $\dot{W}_e$, $\dot{W}_s$ sont définies comme des quantités algébriques. En revanche, les débits massiques $D_{m,\mathrm{A}}$ et $D_{m,\mathrm{B}}$ sont définis comme des quantités positives.

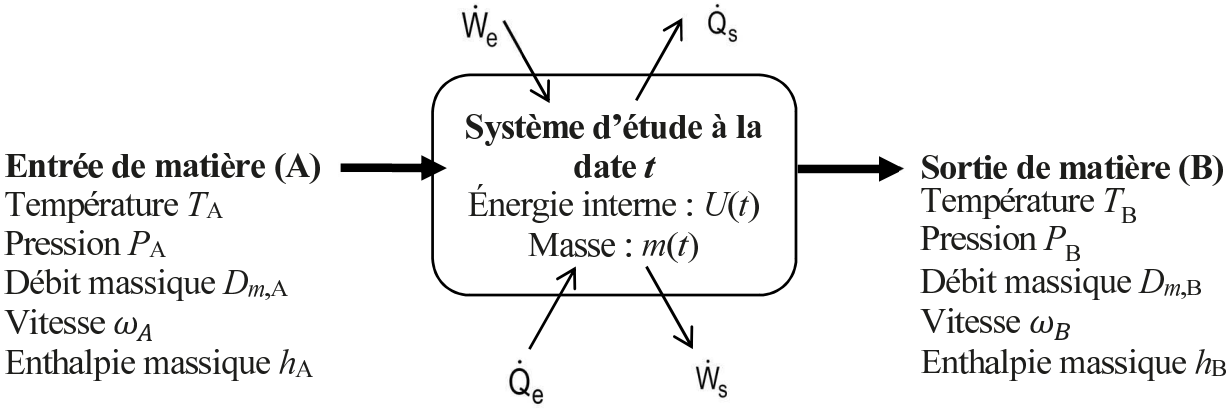


Figure 1 - Représentation schématique d'un système ouvert

Q1. Donner l'unité des grandeurs $\dot{W}$, $\dot{Q}$ et D_m mentionnées sur la **figure 1**.

Q2. Le système considéré est supposé évoluer en régime stationnaire. Quelle est la relation entre les débits des courants de matière entrant $D_{m,\mathrm{A}}$ et sortant $D_{m,\mathrm{B}}$? Justifier.

Q3. Appliquer le premier principe de la thermodynamique au système ouvert stationnaire de la **figure 1**. Montrer qu'il peut se mettre sous la forme :

$$\left(\begin{array}{c}\text{Débit d'énergie}\\ \text{entrant}\end{array}\right) = \left(\begin{array}{c}\text{Débit d'énergie}\\ \text{sortant}\end{array}\right). \qquad (1)$$

On explicitera les termes *débits d'énergie* (homogènes à une puissance) en fonction des grandeurs introduites par la **figure 1**.

Q4. Le système étudié de la **figure 1** est supposé évoluer en régime transitoire. On note $m(t)$, la fonction représentant l'évolution de sa masse m en fonction du temps t. À partir d'un bilan de matière, déduire la relation existant entre $\dfrac{\mathrm{d}m(t)}{\mathrm{d}t}$, $D_{m,\mathrm{A}}$ et $D_{m,\mathrm{B}}$. Proposer une interprétation qualitative du bilan.

On admet dans la suite l'écriture du premier principe en système ouvert, étendue aux systèmes immobiles en régime transitoire :

$$\frac{\mathrm{d}U}{\mathrm{d}t} = \left(\begin{array}{c}\text{Débit d'énergie}\\ \text{entrant}\end{array}\right) - \left(\begin{array}{c}\text{Débit d'énergie}\\ \text{sortant}\end{array}\right) \qquad (2)$$

où $\mathrm{d}U/\mathrm{d}t$ désigne la dérivée de l'énergie interne du système étudié par rapport au temps t.

Q5. Proposer une interprétation qualitative du bilan d'énergie traduit par l'équation (2).

Écriture d'un modèle décrivant la fuite d'un réservoir adiabatique contenant du CO_2

On s'intéresse à présent à un réservoir contenant un gaz supposé parfait. Ce réservoir indéformable (donc de volume V fixe) est le siège d'une fuite vers le milieu environnant à la température T_{ext} = 293 K et à la pression P_{ext} = 1,01 bar supposées fixes dans tout le sujet. Il n'est pas agité mécaniquement. Toutefois, les propriétés du gaz dans le réservoir sont supposées uniformes à chaque instant. Ce réservoir est décrit par la **figure 2**.

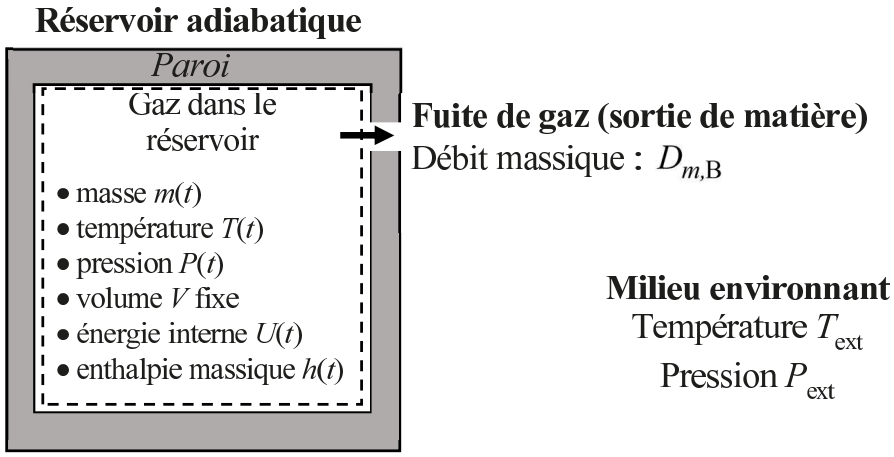


Figure 2 - Réservoir adiabatique renfermant du gaz et soumis à une fuite de gaz vers le milieu environnant

Q6. *Le modèle gaz parfait*

a) Dans quelle situation limite un gaz réel s'identifie-t-il exactement à un gaz parfait ?

b) Donner la valeur du rapport $C_{P,m}/R$ pour un gaz parfait monoatomique. Faire de même pour un gaz parfait diatomique.

c) Pour le CO_2 gaz parfait, le rapport $C_{P,m}/R$ dépend de la température selon une loi notée par la suite $f(T)$. L'étude de la loi $f(T)$ sera l'objet de la partie suivante.

Donner un argument physique expliquant pourquoi, pour le CO_2, $C_{P,m}/R$ est une fonction de la température contrairement au cas des gaz parfaits mono- et di-atomiques.

d) Donner la relation unissant les variations d'énergie interne molaire $\mathrm{d}U_m$ et de température $\mathrm{d}T$ à la fonction $f(T)$. En déduire l'expression de $\mathrm{d}u/\mathrm{d}T$ où u désigne l'énergie interne massique d'un gaz parfait pur.

Q7. On suppose que les propriétés intensives du gaz dans le réservoir (entre autres, sa température T, sa pression P et son enthalpie massique h) sont uniformes. On considèrera que les propriétés intensives du gaz sortant sont les mêmes que celles du gaz dans le réservoir. On s'intéresse au système {volume contenant le gaz à T et P dans le réservoir, paroi non comprise} représenté par des pointillés sur la **figure 2**. Montrer que l'application du bilan de matière et du premier principe au système en pointillés décrit par la **figure 2** amène :

$$\begin{cases} \dfrac{\mathrm{d}m(t)}{\mathrm{d}t} &= -D_{m,B} \\ \dfrac{\mathrm{d}U(t)}{\mathrm{d}t} &= -D_{m,B}\cdot h(t) \end{cases} . \qquad (3)$$

On supposera négligeable l'énergie cinétique massique de la matière quittant le système.

Q8. En introduisant la relation entre l'énergie interne massique u (en J·kg^{-1}) et l'énergie interne U (en J) du système étudié, montrer que l'équation (3) amène :

$$\frac{\mathrm{d}u}{\mathrm{d}t} = \frac{R \cdot T}{M \cdot m} \cdot \frac{\mathrm{d}m}{\mathrm{d}t} \tag{4}$$

où R et M désignent respectivement la constante des gaz parfaits et la masse molaire du gaz.
En déduire que :

$$[f(T) - 1]\frac{\mathrm{d}T}{\mathrm{d}t} = \frac{T}{m} \cdot \frac{\mathrm{d}m}{\mathrm{d}t}. \tag{5}$$

On cherche à présent à estimer le débit $D_{m,\mathrm{B}}$ de gaz quittant le réservoir. Pour ce faire, on s'intéresse à la zone de l'espace dans laquelle se produit la fuite.

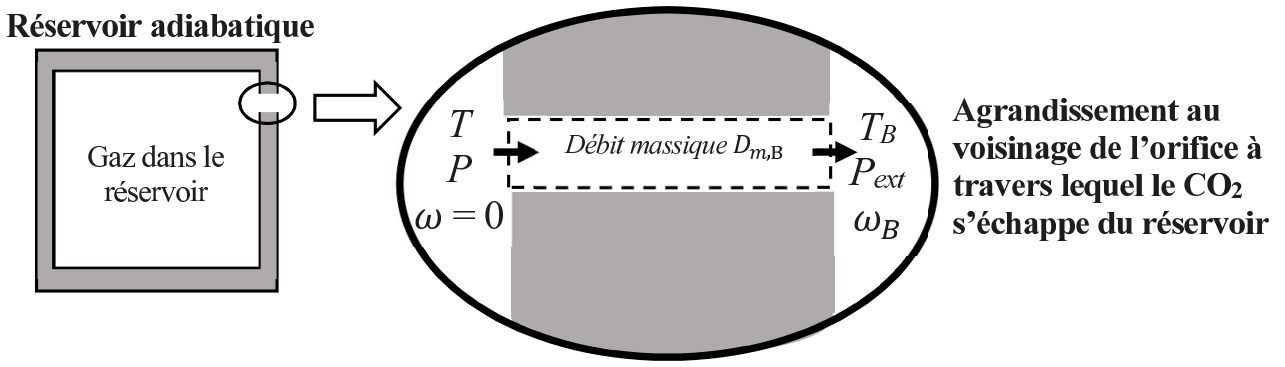


Figure 3 - Fuite de CO_2 gazeux à travers l'orifice dans la paroi du réservoir

On considère le système {orifice dans la paroi du réservoir} ; on supposera que :
- l'écoulement de gaz à travers l'orifice est **adiabatique** et **stationnaire** ;
- la vitesse du gaz en entrée du système est nulle ; elle est notée ω_B en sortie ;
- la température et la pression du gaz en entrée sont notées T et P (ce sont celles du gaz à l'intérieur du réservoir) ; en sortie, elles sont notées T_B et $P_B = P_{\mathrm{ext}} = 1{,}01$ bar.

Q9. Montrer que le débit massique $D_{m,\mathrm{B}}$ de fluide à travers la section de l'orifice est donné par :

$$D_{m,\mathrm{B}} = \frac{\Omega \cdot \omega_B \cdot M \cdot P_{\mathrm{ext}}}{R \cdot T_B} \tag{6}$$

où la surface Ω désigne la section de passage du fluide à travers l'orifice.

Q10. *Expression de ω_B en fonction de T_B*
En appliquant l'expression du premier principe en système ouvert stationnaire, donnée par l'équation (1), au système {orifice dans la paroi du réservoir} délimité par des pointillés sur la **figure 3**, montrer que la vitesse de sortie du CO_2 a pour expression :

$$\omega_B = \sqrt{2\frac{R}{M}\int_{T_B}^{T} f(T)\mathrm{d}T}. \tag{7}$$

Q11. Méthode de calcul de T_B

a) En négligeant les frottements au sein du système considéré, on peut supposer l'écoulement **réversible**. Montrer, par application du second principe, que cette hypothèse amène à supposer l'écoulement *isentropique massique* (ou molaire) (i.e., à entropie massique - ou molaire - constante).

b) Montrer que la variation d'entropie massique d'un gaz parfait pur s'écrit :

$$\mathrm{d}s = \frac{R}{M}\left[\frac{C_{P,m}}{R}\frac{\mathrm{d}T}{T} - \frac{\mathrm{d}P}{P}\right]. \tag{8}$$

En déduire que T_B est solution de l'équation :

$$\int_T^{T_B} \frac{f(T)}{T}\mathrm{d}T + \ln\left(\frac{P}{P_{\text{ext}}}\right) = 0 \quad \text{avec } P = \frac{m\cdot R\cdot T}{M\cdot V}. \tag{9}$$

Finalement, en réunissant les équations (3), (5), (6), (7) et (9), le modèle ainsi constitué est représenté par le système suivant :

$$\begin{cases} \dfrac{\mathrm{d}T}{\mathrm{d}t} = \dfrac{T}{m[f(T)-1]}\cdot\dfrac{\mathrm{d}m}{\mathrm{d}t} \\ \dfrac{\mathrm{d}m}{\mathrm{d}t} = -\dfrac{\Omega P_{\text{ext}}\sqrt{2\int_{T_B}^{T} f(T)\mathrm{d}T}}{\sqrt{\frac{R}{M}}\,T_B} \end{cases} \tag{10}$$

avec T_B solution de :

$$\int_T^{T_B} \frac{f(T)}{T}\mathrm{d}T + \ln\left(\frac{m\cdot R\cdot T}{M\cdot V\cdot P_{\text{ext}}}\right) = 0. \tag{11}$$

Partie II - Modélisation de la fuite d'un réservoir : traitement numérique

Dans cette partie, on fera référence au modèle défini par les équations (10) et (11).

Données numériques utiles pour cette partie :

$$R = 8{,}3144\ \text{J}\cdot\text{mol}^{-1}\cdot\text{K}^{-1}$$
$$M = 44\cdot 10^{-3}\ \text{kg/mol}$$
$$T_{\text{ext}} = 293\ \text{K}$$
$$P_{\text{ext}} = 1{,}01\ \text{bar}$$
$$V = 1\ \text{m}^3$$
$$\Omega = 1\ \text{cm}^2$$

Modèle pour $C_{P,m}/R$ *:* $f(T) = A_1 + \dfrac{A_2}{T + A_3}$ avec $\begin{cases} T \text{ en K} \\ A_1 = 8{,}303 \\ A_2 = -\,2\,810 \\ A_3 = 485{,}6 \end{cases}$

Conditions initiales : $\begin{cases} T(0) = 473{,}15\ \text{K} \\ m(0) = 11{,}0\ \text{kg} \end{cases}$

Q12. Intégration numérique

a) Donner le code de la fonction `integ1(T1,T2)` réalisant le **calcul analytique** de l'intégrale $\int_{T_1}^{T_2} f(T)\mathrm{d}T$ (`T1` et `T2` sont des arguments d'entrée de la fonction `integ1` qui désignent les bornes de l'intégrale).

b) Donner le code de la fonction `integ2(T1,T2)` réalisant le **calcul numérique** de l'intégrale $\int_{T_1}^{T_2} \frac{f(T)}{T}\mathrm{d}T$ (`T1` et `T2` sont des arguments d'entrée de la fonction `integ2` qui désignent les bornes de l'intégrale).

La méthode des rectangles sera adoptée pour estimer l'intégrale ; l'intervalle des températures sera divisé en 100 sous-intervalles).

Ces deux fonctions pourront être appelées autant de fois que nécessaire par la suite.

Q13. La fonction `chercheTB(T,m)` fournie ci-dessous renvoie, pour des valeurs connues des variables T et m, la valeur correspondante de T_B obtenue par résolution de l'équation (11) selon la méthode de Newton :

```
import numpy as np
A1 = 8.303; A2 = -2810.; A3 = 485.6
T0 = 473.15; m0 = 11.0
Pext = 1.01e5; Mw = 44e-3; Volume = 1.0; Text = 293.
R = 8.3144
def chercheTB(T,m):
      TB = 300.
      [instruction1]
      while residu > 1e-10:
           eq = [instruction2]
           deq = [instruction3]
           TBold = TB
           TB = TB - eq/deq
           residu = [instruction4]
      return TB
```

Pour compléter la portion de code ci-dessus, indiquer par quelles instructions il convient de remplacer les séquences **[instruction1]** à **[instruction4]**. Justifier.

La fonction `chercheTB(T,m)` pourra être appelée autant de fois que nécessaire par la suite.

Pour résoudre une équation différentielle mise sous la forme $y'(t) = f(t,y(t))$, on peut utiliser le schéma d'Euler dont on rappelle l'expression :

$$y(t + \Delta t) = y(t) + f(t,y(t)) \times \Delta t. \tag{12}$$

Dans le cas d'un système d'équations différentielles d'ordre 1, de la forme $\mathbf{y}'(t) = \mathbf{f}(t,\mathbf{y}(t))$, on peut généraliser le schéma d'Euler de la manière suivante :

$$\mathbf{y}(t + \Delta t) = \mathbf{y}(t) + \mathbf{f}(t,\mathbf{y}(t)) \times \Delta t \tag{13}$$

où $\mathbf{y}(t)$ désigne le vecteur des fonctions inconnues évaluées à la date courante t.

Q14. Donner l'algorithme général du calcul numérique des fonctions temporelles T, m et P entre $t = 0$ et $t = 10$ s, par résolution du système différentiel (10) reposant sur l'utilisation du schéma d'Euler généralisé (on choisira comme pas de temps : $\Delta t = 0{,}10$ s).
Cet algorithme sera fourni sous la forme d'un logigramme. On veillera à mentionner les procédures d'initialisation des processus itératifs ainsi que leurs critères d'arrêt. Quand cela sera nécessaire, on fera appel à la fonction `chercheTB(T,m)` sans détailler sa structure.

Q15. Écrire le code mettant en œuvre l'algorithme proposé à la question précédente. On pourra faire appel à toutes les fonctions programmées précédemment.

Q16. Intuitivement, quand la fuite s'arrêtera-t-elle en pratique ?

Partie III - Développement de corrélations pour la capacité thermique à pression constante molaire du dioxyde de carbone

On dispose d'un faible nombre de mesures expérimentales du rapport $C_{P,m}/R$ du CO_2 gazeux sur un large domaine de températures T sous pression atmosphérique (où $C_{P,m}$ désigne la capacité thermique molaire à pression constante et R, la constante des gaz parfaits).
Ces mesures (au nombre de 6) sont consignées dans un fichier `cP.txt` (voir **tableau 1**). La première colonne donne la température en K, la seconde fournit les mesures expérimentales de la quantité $C_{P,m}/R$.

```
100.0       3.513
500.0       5.367
1000.0      6.532
2000.0      7.257
3000.0      7.475
4000.0      7.586
```

Tableau 1 - Contenu du fichier `cP.txt`

Dans cette partie, on cherche à développer une corrélation de la propriété $C_{P,m}/R$ en fonction de la température. Le modèle est noté $f(T)$ par la suite :

$$(C_{P,m}/R)_{\text{modèle}} = f(T). \tag{14}$$

Les coefficients intervenant dans la corrélation seront déterminés de manière à reproduire efficacement les données expérimentales $(T, (C_{P,m}/R)_{\text{exp}})$ dont on dispose. Pour y parvenir, deux stratégies sont envisagées et testées :

- un premier modèle de la forme :

$$f_{\text{modèle 1}}(T) = A_1 + \frac{A_2}{T + A_3} \tag{15}$$

  où A_1, A_2 et A_3 sont des constantes ;
- un second modèle de forme polynomiale :

$$f_{\text{modèle 2}}(T) = \sum_{i=0}^{n} B_i \cdot T_r^i \qquad \text{avec } T_r = T/T_c \tag{16}$$

  où $T_c = 304{,}21$ K désigne la température critique du CO_2, n, le degré du polynôme. Les B_i sont les coefficients réels du polynôme.

Q17. Les valeurs des capacités thermiques reportées dans le **tableau 1** ont été mesurées par calorimétrie. Il existe de nombreuses techniques de mesures calorimétriques. Décrire succinctement le principe d'une technique expérimentale de mesure de la capacité thermique d'un liquide ou d'un gaz par calorimétrie que vous connaissez ; en particulier, fournir un schéma pour illustrer le principe de la mesure et préciser comment la valeur de la capacité thermique est déduite de la mesure.

Q18. Montrer par une analyse dimensionnelle que le rapport $C_{P,m}/R$ est sans dimension.

Q19. *Programme de lecture des données*
Indiquer la syntaxe à utiliser pour charger le fichier `cP.txt` qui contient les données expérimentales et stocker celles-ci dans deux tableaux de réels `temp` (pour les températures) et `CpR_exp` (pour les valeurs expérimentales du rapport $C_{P,m}/R$).
De plus, déterminer automatiquement le nombre de points expérimentaux N_{exp} présents dans le fichier chargé.

III.1 - Développement du premier modèle

On cherche à déterminer les coefficients A_1, A_2 et A_3 du modèle n°1 donné par l'équation (15).
Ces coefficients sont inconnus dans cette partie. Pour ce faire, on va utiliser une technique de minimisation. L'idée générale est la suivante :

- on forme une fonction dite *objectif*, notée f_{obj} rendant compte des écarts entre les prédictions du modèle et les valeurs expérimentales ;
- on va chercher à minimiser cette fonction en jouant sur les valeurs des coefficients A_1, A_2 et A_3.

La fonction objectif choisie a pour expression :

$$f_{\text{obj}}(A_1,A_2,A_3) = \sum_{i=1}^{N_{\text{exp}}} \delta_i^2 \qquad \text{avec : } \delta_i = f_{\text{modèle 1}}(T_{\text{point exp. n°}i}) - (C_{P,m}/R)_{\text{point exp. n°}i}\,. \tag{17}$$

Q20. Dans l'expression de la fonction objectif, pour quelle raison les écarts ont-ils été élevés au carré ?

On range les coefficients recherchés dans le vecteur $\mathbf{a} = (A_1,A_2,A_3)$.

Q21. Écrire la syntaxe de la fonction `delta(vecA,temp,CpR_exp)` qui prend comme argument d'entrée un jeu quelconque de coefficients **a** (`vecA` désigne le tableau contenant les éléments du vecteur **a**), les tableaux `temp` et `CpR_exp` et renvoie en sortie le vecteur $\boldsymbol{\delta}$ contenant les éléments δ_i définis par l'équation (17).

Q22. Écrire la syntaxe de la fonction `fobj(vecA,temp,CpR_exp)` permettant l'estimation de la fonction-objectif pour un jeu quelconque de coefficients **a** (`vecA` désigne le tableau contenant les éléments du vecteur **a**). On pourra utiliser la fonction `delta` définie à la question précédente.

Un extremum local de la fonction objectif vérifie :

$$\frac{\partial f_{\text{obj}}}{\partial A_j} = 0 \qquad \text{pour } 1 \leq j \leq 3. \tag{18}$$

C'est donc, en particulier, le cas d'un minimum. À présent, on a besoin d'exprimer et d'estimer les dérivées $\dfrac{\partial f_{\text{obj}}}{\partial A_j}$ (pour $1 \leq j \leq 3$).

Q23. Donner les expressions analytiques des dérivées $\frac{\partial f_{\text{modèle 1}}}{\partial A_1}$, $\frac{\partial f_{\text{modèle 1}}}{\partial A_2}$ et $\frac{\partial f_{\text{modèle 1}}}{\partial A_3}$.

Q24. Montrer que les expressions analytiques des dérivées $\frac{\partial f_{\text{obj}}}{\partial A_j}$ peuvent se mettre sous la forme :

$$\frac{\partial f_{\text{obj}}}{\partial A_j} = 2 \sum_{i=1}^{N_{\text{exp}}} \delta_i \times \frac{\partial f_{\text{modèle 1}}}{\partial A_j} \qquad \text{pour } 1 \leq j \leq 3. \tag{19}$$

et fournir explicitement leurs expressions.

On note $\mathbf{f}'$, le gradient de f_{obj}, i.e., le vecteur des 3 dérivées partielles $\frac{\partial f_{\text{obj}}}{\partial A_j}$ (pour $1 \leq j \leq 3$).

Q25. Écrire la syntaxe d'une fonction `deriv_fobj(vecA,temp,CpR_exp)` permettant l'estimation de $\mathbf{f}'$. Cette fonction pourra faire appel aux fonctions programmées précédemment.

Pour résoudre l'équation (18), une méthode de type Newton est envisagée. Les valeurs de $\mathbf{a}$ minimisant la fonction-objectif sont obtenues à partir du schéma itératif suivant :

$$\mathbf{a}^{\text{itération } (k+1)} = \mathbf{a}^{\text{itération } (k)} - \mathbf{H}^{-1}\,\mathbf{f}' \tag{20}$$

où $\mathbf{H}$ et $\mathbf{f}'$ sont respectivement la matrice hessienne et le gradient de f_{obj} estimés à l'itération (k). La matrice hessienne $\mathbf{H}$ de la fonction f_{obj} désigne la matrice (symétrique) de ses dérivées partielles secondes dont l'expression est définie ci-après :

$$\mathbf{H} = \begin{pmatrix} \frac{\partial^2 f_{\text{obj}}}{\partial A_1^2} & \frac{\partial^2 f_{\text{obj}}}{\partial A_1 \partial A_2} & \frac{\partial^2 f_{\text{obj}}}{\partial A_1 \partial A_3} \\ \frac{\partial^2 f_{\text{obj}}}{\partial A_2 \partial A_1} & \frac{\partial^2 f_{\text{obj}}}{\partial A_2^2} & \frac{\partial^2 f_{\text{obj}}}{\partial A_2 \partial A_3} \\ \frac{\partial^2 f_{\text{obj}}}{\partial A_3 \partial A_1} & \frac{\partial^2 f_{\text{obj}}}{\partial A_3 \partial A_2} & \frac{\partial^2 f_{\text{obj}}}{\partial A_3^2} \end{pmatrix} = \begin{pmatrix} \frac{\partial f_1'}{\partial A_1} & \frac{\partial f_2'}{\partial A_1} & \frac{\partial f_3'}{\partial A_1} \\ \frac{\partial f_1'}{\partial A_2} & \frac{\partial f_2'}{\partial A_2} & \frac{\partial f_3'}{\partial A_2} \\ \frac{\partial f_1'}{\partial A_3} & \frac{\partial f_2'}{\partial A_3} & \frac{\partial f_3'}{\partial A_3} \end{pmatrix}. \tag{21}$$

On choisit d'estimer numériquement les éléments de la matrice hessienne à partir de l'approximation suivante :

$$H_{ij} = \frac{\partial f_j'}{\partial A_i} \approx \frac{f_j'(\mathbf{a}_{\text{après}}) - f_j'(\mathbf{a})}{\epsilon} \quad \text{pour } 1 \leq i,j \leq 3 \tag{22}$$

où $\mathbf{a}_{\text{après}}$ est égal à $\mathbf{a}$ excepté pour la $i^{\text{ème}}$ composante qui est augmentée de la quantité ϵ.
Par exemple, si $i = 2$, on a : $\mathbf{a}_{\text{après}} = (A_1\ ;\ A_2 + \epsilon\ ;\ A_3)$.

Q26. Écrire la syntaxe d'une fonction `hessienne_fobj(vecA,temp,CpR_exp)` permettant l'estimation de la matrice hessienne de la fonction-objectif en un point $\mathbf{a}$ quelconque. Cette fonction pourra faire appel aux fonctions programmées précédemment. On prendra $\epsilon = 10^{-5}$.

Q27. On choisit comme point de départ de la procédure itérative : $\mathbf{a}^{(0)} = (5\ ;\ -1\,000\ ;\ 500)$.
Écrire un code permettant de mettre en œuvre le calcul du $\mathbf{a}$ optimal (noté $\mathbf{a}^*$) à partir d'une méthode de type Newton. On proposera en particulier un critère d'arrêt pertinent des itérations (ce choix sera justifié).

Q28. Pour s'assurer que l'extremum obtenu est un minimum, il convient de vérifier que la matrice $\mathbf{H}$ évaluée en $\mathbf{a}^*$ est semi-définie positive (i.e., que ses valeurs propres sont positives ou nulles). Pour ce faire, écrire une fonction prenant comme argument d'entrée $\mathbf{H}(\mathbf{a}^*)$ et renvoyant en sortie 0 si la plus petite des valeurs propres de $\mathbf{H}(\mathbf{a}^*)$ est positive ou nulle et 1 si ce n'est pas le cas. Le calcul des valeurs propres sera effectué à l'aide d'une fonction-intrinsèque adaptée (voir annexe).

III.2 - Développement du second modèle : le modèle polynomial

Pour décrire les mesures expérimentales, on choisit d'utiliser un **polynôme d'interpolation** dont l'expression générale est donnée par l'équation (16). Cela signifie que le degré n du polynôme et ses coefficients seront déterminés de manière à ce que la courbe représentative de la fonction $f_{\text{modèle 2}}$ passe exactement par les 6 points expérimentaux.

Q29. Déduire le degré n du polynôme du nombre de données expérimentales. Justifier brièvement.

Q30. Écrire le système d'équations que doivent vérifier les coefficients B_i du polynôme de manière à reproduire exactement les données expérimentales.

Q31. Montrer que ce système est linéaire en le mettant sous la forme :

$$\mathbf{y}_{\text{exp}} = \mathbf{M}\,\mathbf{b} \tag{23}$$

où $\mathbf{y}_{\text{exp}}$ est le vecteur contenant les mesures expérimentales de $(C_{P,m}/R)_{\text{exp}}$, $\mathbf{b}$ le vecteur contenant les coefficients B_i du polynôme et $\mathbf{M}$ une matrice carrée dont les coefficients ne dépendent que des valeurs expérimentales de $T_r = T/T_c$.
Donner explicitement l'expression de la matrice $\mathbf{M}$.

Q32. Mathématiquement, quelle opération algébrique faut-il effectuer pour accéder aux valeurs des coefficients B_i ?

Dans la **figure 4**, on a superposé dans le plan $(C_{P,m},T)$ les points expérimentaux ainsi que les courbes associées aux deux modèles proposés. Les coefficients de ces modèles ont été déterminés suivant les procédures présentées précédemment.

Q33. Donner la syntaxe du code permettant de générer cette figure (*la courbe du modèle 1 sera représentée par un trait pointillé tandis que celle du modèle 2 sera représentée par un trait continu*). On rappelle que les coordonnées des points expérimentaux sont contenues dans les tableaux `temp` et `CpR_exp` ; les 3 coefficients du modèle 1 sont stockés dans `vecA` ; les 6 coefficients du modèle 2 sont supposés être stockés dans un vecteur `vecB`.

Q34. Commentez le graphe de la **figure 4**. Finalement, parmi les deux modèles proposés, lequel retiendriez-vous et pourquoi ?
Vous pourrez prendre en considération l'influence des incertitudes expérimentales sur les valeurs des paramètres des deux modèles considérés précédemment.

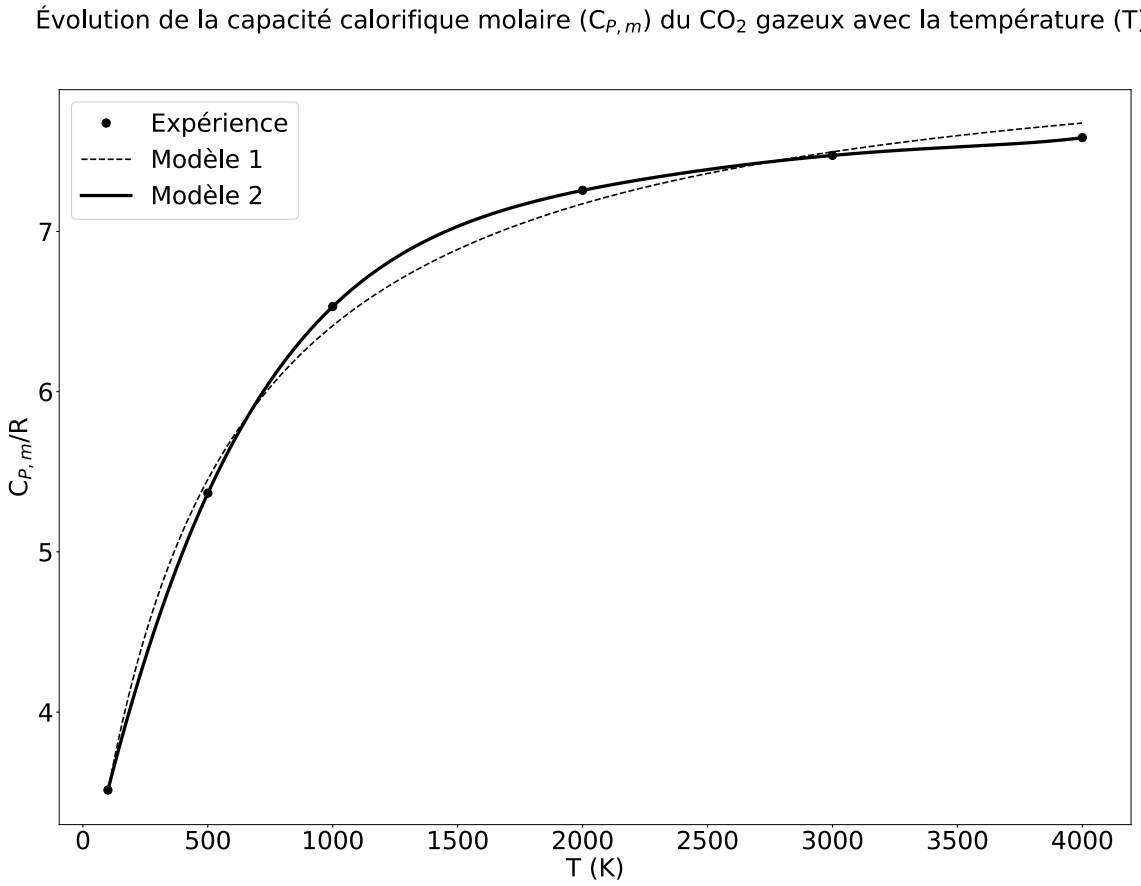


Figure 4 - Confrontation des points expérimentaux et des valeurs prédites par les deux modèles proposés dans le plan $C_{P,m}/R$ versus T

ANNEXE
Quelques commandes utiles en langage Python

A - Bibliothèque NUMPY de Python (gestion des tableaux, matrices, vecteurs et fichiers) 13
B - Résolution d'une équation non linéaire à une inconnue 15
C - Bibliothèque MATPLOTLIB.PYPLOT de Python (gestion des graphes) 16

A - Bibliothèque NUMPY de Python (gestion des tableaux, matrices, vecteurs et fichiers)

Dans les exemples ci-dessous, la bibliothèque `numpy` a préalablement été importée à l'aide de la commande : `import numpy as np`. On peut alors utiliser les fonctions de la bibliothèque, dont voici quelques exemples :

np.array(liste)

Description : fonction permettant de créer une matrice (de type tableau) à partir d'une liste.
Argument d'entrée : une liste définissant un tableau à 1 dimension (vecteur) ou 2 dimensions (matrice).
Argument de sortie : un tableau (matrice).

Exemples :

Commande	Résultat
`np.array([4,3,2])`	`[4 3 2]`

`np.array([[5],[7],[1]])`	`[[5]`
	`[7]`
	`[1]]`

`np.array([[3,4,10],[1,8,7]])`	`[[3 4 10]` `[1 8 7]]`

A[i,j]

Description : fonction qui retourne l'élément $(i+1, j+1)$ de la matrice A. Pour accéder à l'intégralité de la ligne $i+1$ de la matrice A, on écrit $A[i, :]$. De même, pour obtenir toute la colonne $j+1$ de la matrice A, on utilise la syntaxe $A[: , j]$.
Argument d'entrée : une liste contenant les coordonnées de l'élément dans le tableau A.
Arguments de sortie : l'élément $(i+1, j+1)$ de la matrice A.
RAPPEL : en langage Python, les lignes d'un tableau A de taille $n \times m$ sont numérotées de 0 à $n-1$ et les colonnes sont numérotées de 0 à $m-1$.

Exemples :

Commande	Résultat
`A=np.array([[3,4,10],[1,8,7]])` `A[0,2]`	`10`
`A[1,:]`	`[1 8 7]`
`A[:,2]`	`[10 7]`

np.zeros((n,m))

Description : fonction créant une matrice (tableau) de taille $n \times m$ dont tous les éléments sont nuls.
Argument d'entrée : un tuple de deux entiers correspondant aux dimensions de la matrice à créer.
Argument de sortie : un tableau (matrice) d'éléments nuls.

Exemple :

Commande	Résultat

`np.zeros((3,4))`	`[[0 0 0 0]`
	`[0 0 0 0]`
	`[0 0 0 0]]`

np.dot(mat1,mat2)

Description : fonction calculant le produit de deux matrices `mat1` et `mat2`.
Argument d'entrée : matrices `mat1` et `mat2`.
Argument de sortie : la matrice produit de `mat1` et `mat2`.

np.linalg.inv(mat)

Description : fonction calculant la matrice inverse de la matrice `mat`.
Argument d'entrée : matrice `mat`.
Argument de sortie : la matrice inverse de `mat`.

np.linalg.eig(mat)

Description : fonction calculant les valeurs propres de la matrice `mat`.
Argument d'entrée : matrice `mat`.
Argument de sortie : c'est un tuple dont le premier élément correspond aux valeurs propres tandis que le second élément contient les vecteurs propres.
Pour accéder aux valeurs propres de `mat`, on écrira :

```
np.linalg.eig(mat)[0]
```

np.linspace(Min,Max,nbElements)

Description : fonction créant un vecteur (tableau) de `nbElements` nombres espacés régulièrement entre `Min` et `Max`. Le premier élément est égal à `Min`, le dernier est égal à `Max` et les éléments sont espacés de `(Max-Min)/(nbElements-1)`.
Argument d'entrée : un tuple de 3 entiers.
Argument de sortie : un tableau (vecteur).
Exemple :

Commande	Résultat
`np.linspace(3,25,5)`	`[3 8.5 14 19.5 25]`

np.loadtxt('nom_fichier',delimiter='string',usecols=[n])

Description : fonction permettant de lire les données sous forme de matrice dans un fichier texte et de les stocker sous forme de vecteurs.
Argument d'entrée : le nom du fichier qui contient les données à charger, le type de caractère utilisé dans ce fichier pour séparer les données (par exemple, une espace ou une virgule) et le numéro de la colonne à charger (RAPPEL : la première colonne est affectée du numéro 0).
Argument de sortie : un tableau.
Exemple :

```
data=np.loadtxt('fichier.txt',delimiter=' ',usecols=[0])
```

Dans cet exemple, `data` est un vecteur qui correspond à la première colonne de la matrice contenue dans le fichier `fichier.txt`.

B - Résolution d'une équation non linéaire à une inconnue

La bibliothèque adaptée est chargée par l'instruction : `from scipy import optimize`.
L'équation à résoudre est mise sous la forme : $f(x) = 0$, où x désigne l'inconnue. La fonction f doit d'abord être définie à l'aide de l'instruction `def`.
Exemple : si l'équation à résoudre s'écrit $\log x = 3$, la fonction f a pour expression : $f(x) = \log x - 3$.
Codage :

```
def f(x):
      return log10(x) - 3.
```

.

La fonction `root` peut alors être utilisée pour résoudre l'équation. La variable x doit être initialisée à une valeur aussi proche que possible de la solution.

```
# Valeur initiale de x :
Xinit = 1.

# jac = None signifie que la dérivée de f (dont se sert
#     la fonction root pour effectuer la résolution)
#     n'est pas fournie par l'utilisateur mais estimée
#     numériquement par Python.
Xsol = root(f,Xinit,jac=None)
```

La solution de l'équation est rangée dans le premier élément du vecteur `Xsol.x`, c'est-à-dire :

```
Xsol.x[0]
```

C - Bibliothèque MATPLOTLIB.PYPLOT de Python (gestion des graphes)

Cette bibliothèque permet de tracer des graphiques. Dans les exemples ci-dessous, la bibliothèque `matplotlib.pyplot` a préalablement été importée à l'aide de la commande :

```
import matplotlib.pyplot as plt
```

On peut alors utiliser les fonctions de la bibliothèque, dont voici quelques exemples :

plt.plot(x,y,'SC')

Description : fonction permettant de tracer un graphique de n points dont les abscisses sont contenues dans le vecteur x et les ordonnées dans le vecteur y. Cette fonction doit être suivie de la fonction `plt.show()` pour que le graphique soit affiché.

Argument d'entrée : un vecteur d'abscisses x (tableau de n éléments) et un vecteur d'ordonnées y (tableau de n éléments).

La chaîne de caractères `'SC'` précise le style et la couleur de la courbe tracée. Des valeurs possibles pour ces deux critères sont :

Valeurs possibles pour S (style) :

Description	Ligne continue	Ligne traitillée	Marqueur rond	Marqueur plus
Symbole S	-	- -	o	+

Valeurs possibles pour C (couleur) :

Description	bleu	rouge	vert	noir
Symbole C	b	r	g	k

Argument de sortie : un graphique.

Exemple :

```
x= np.linspace(3,25,5)
y=sin(x)
plt.plot(x,y,'-b') # tracé d'une ligne bleue continue
plt.title('titre_graphique') # titre du graphe
plt.xlabel('x') # titre de l'axe des abscisses
plt.ylabel('y') # titre de l'axe des ordonnées
plt.show()
```

FIN

IMPRIMERIE NATIONALE – 21 1163 – D'après documents fournis

Travailler les chapitres de ce sujet

Prepa Help suit le programme de PC chapitre par chapitre : cours détaillé, méthodes, pièges, exercices et problèmes corrigés pas à pas, tirés des concours. L'essai gratuit donne accès aux chapitres offerts de ta filière, sans carte bancaire.

Chapitres liés à ce sujet

- Bases de Python : expressions, types, instructions, fonctions
- Grandeurs, unités, analyse dimensionnelle et ordres de grandeur

Ouvrir l'essai gratuit

Scanne le code ou clique sur le bouton.

prepahelp.com/#essai

Une question sur le concours ou sur ta filière : **écrire sur WhatsApp**

Les autres années et épreuves : prepahelp.com/annales/e3a/pc