

Sujet CCINP 2023 PSI – Modélisation et ingénierie numérique

Calcul numérique d'intégrale

Partie I - Modélisation du cœur, un organe indispensable et parfois défaillant

Objectif

L'objectif de cette partie est de comparer les performances d'un cœur sain à celles du cœur d'un patient atteint d'*hypertrophie myocardique*; l'intérêt de la pompe d'assistance cardiaque sera ainsi mis en avant pour compenser les effets de ce type de pathologie.

I.1 - Modélisation d'un cœur sain

En ne considérant que la moitié gauche du cœur, le cycle thermodynamique (noté Γ_{th}) subi par le sang contenu dans le ventricule et l'oreillette peut être tracé dans un diagramme pression - volume (**figure 3**). Les étapes du cycle Γ_{th} sont détaillées ci-après :

- **A → B remplissage ventriculaire** : la valve oriculo-ventriculaire gauche (valve mitrale numérotée ① sur la **figure 1**) s'ouvre, le volume du ventricule gauche ③ passe rapidement d'environ $V_0 = 70$ mL à $V_1 = 140$ mL suivant une courbe appelée "*courbe de compliance*";
- **B → C phase de diastole** : remplissage lent du ventricule, déjà quasiment plein. L'afflux de peu de sang est alors responsable d'une forte augmentation de la pression dans le ventricule. La pression monte jusqu'à la *tension systolique* $p_1 = 1,70 \cdot 10^4$ Pa;
- **C → D éjection systolique** : l'oreillette gauche ⑤ se contracte, éjectant le sang dans l'artère aorte par la valve aortique ② ouverte. Le volume du ventricule redescend à $V_0 = 70$ mL;
- **D → A relaxation isovolumétrique** : toutes les valves se ferment, les cellules musculaires ventriculaires se relâchent et la pression retombe à la *tension diastolique* $p_0 = 1,00 \cdot 10^4$ Pa.

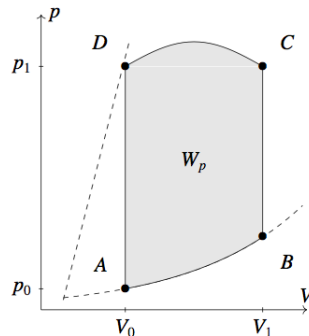


Figure 3 - Cycle thermodynamique Γ_{th} subi par le sang

Q1. Justifier le sens de parcours de ce cycle dans le diagramme (p, V) au vu du rôle du cœur dans le corps humain. Identifier alors la signification de W_p , l'aire algébrique grisée définie positive :

$$W_p = -\oint_{\Gamma_{th}} p dV. \quad (1)$$

Considérons l'équation différentielle suivante :

$$-\frac{dW}{dV}(V) = p(V) \quad \text{soit} \quad W(V_{n-1}) - W(V_n) = \int_{V_{n-1}}^{V_n} p dV. \quad (2)$$

Il est possible de discrétiser l'équation (2) en approximant l'intégrale par les trois surfaces représentées **figure 4** :

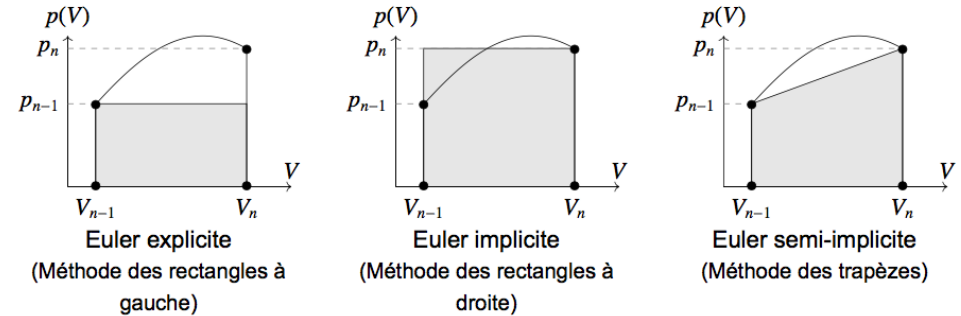


Figure 4 - Différents schémas de discrétisation d'une intégrale

V_{n-1} et V_n seront les volumes du ventricule à deux instants consécutifs, indicés $n-1$ et n en référence au pas de temps du modèle numérique. De même, les notations suivantes seront utilisées : $W_n = W(V_n)$, $W_{n-1} = W(V_{n-1})$, $p_n = p(V_n)$ et $p_{n-1} = p(V_{n-1})$, avec $0 < n \leq N$, $N \in \mathbb{N}$ étant le nombre de points de la simulation numérique.

Q2. Pour les trois schémas de la **figure 4**, préciser le lien qui existe entre W_n , W_{n-1} , p_n , p_{n-1} , V_n et V_{n-1} en fonction du schéma de discrétisation choisi pour le calcul de l'intégrale en utilisant l'équation (2).

Le calcul numérique de $W_p = W_N - W_0$ permet d'obtenir le résultat suivant : $|W_p| = 0,5$ USI.

Q3. Par un raisonnement graphique sur le diagramme (p, V) , montrer que cette valeur obtenue est cohérente. Préciser son signe et son unité.

Q4. En supposant que la partie droite du cœur a un fonctionnement similaire à la partie gauche, exprimer et calculer numériquement la puissance mécanique $\mathcal{P}_{\text{cœur}}$ développée par le cœur pour pomper le sang à une fréquence de $f = 60$ batt $\cdot$ min $^{-1}$.

Q5. En prenant un rendement $\eta = 20$ % du muscle cardiaque, estimer la fraction x de l'énergie consommée par le cœur pour un individu ayant un apport énergétique journalier de $\mathcal{E} = 2,4$ kWh (ce qui représente environ 2 000 kcal, soit l'apport énergétique journalier moyen d'un adulte).

Sujet E3A 2015 PSI Physique et Modélisation – Calcul d'intégrale par la méthode des rectangles

Commentaire : sujet + ancien « + costaud »

F Calcul d'une intégrale

L'objectif de cette partie est d'étudier une méthode générale d'intégration. On l'utilisera pour

obtenir une valeur numérique approchée de : $I = \int_0^\pi \frac{\cos^2\left(\frac{\pi}{2} \cos \theta\right)}{\sin \theta} d\theta$

L'application $f : x \mapsto \frac{\cos^2\left(\frac{\pi}{2} \cos(x)\right)}{\sin(x)}$ est continue sur $]0, \pi[$ et est prolongeable par continuité en

0 et π en posant $f(0) = f(\pi) = 0$. L'intégrale I est donc bien définie.

La méthode des rectangles consiste à approcher l'intégrale par la suite $(R_n)_{n \in \mathbb{N}^*}$ définie pour notre fonction f par :

$$R_1 = 0 \text{ et pour tout } n \geq 2, R_n = \lambda_n \sum_{k=1}^{n-1} f\left(\frac{k\pi}{n}\right)$$

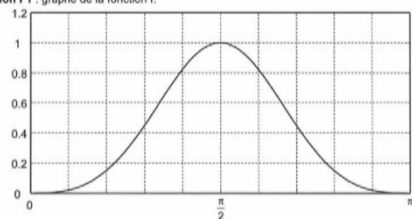
où λ_n désigne un nombre réel qu'il faudra préciser.

Annexe : Document réponse

Question E1 : fonction Resolution en langages Python et Scilab.

En langage Python :	En langage Scilab :
<pre>def Resolution(N,eps) : valeurs=[1*300/N for i in range(-N,N+1)] for X1 in valeurs : for X2 in valeurs : if abs(X1(X1,X2))<eps : sol1=sol1.append(X1) sol2=sol2.append(X2) return [sol1,sol2]</pre>	<pre>function [sol1,sol2]=Resolution(N,eps) valeurs=-300 : 300/N : 300 ; for X1=valeurs for X2=valeurs if abs(X1(X1,X2))<eps then sol1=[sol1, X1] ; sol2=[sol2, X2] ; end ; end ; end ; endfunction</pre>

Question F1 : graphe de la fonction f .



Tous les algorithmes demandés dans cette partie devront être réalisés dans un même langage. Ce langage pourra être le langage Python ou le langage Scilab. On n'utilisera aucune librairie particulière mais on considérera que les fonctions $\cos$ et $\sin$ ainsi que le nombre π ont été importés dans Python. On rappelle que le nombre π se note `%pi` dans Scilab.

Il n'est pas demandé de vérifier dans les algorithmes que les variables d'entrée sont bien du type voulu. Il est ainsi par exemple inutile dans la question F2, de tester si la variable d'entrée x est bien dans l'intervalle $]0, \pi[$.

F1. Illustrer le principe de la méthode des rectangles en représentant R_{10} sur le graphe de la fonction f fourni en annexe. Donner l'expression de λ_n en fonction de n .

F2. Écrire une fonction $f(x)$ qui retourne la valeur de f pour un nombre réel x de $]0, \pi[$.

F3. Écrire une fonction **Rectangle1(n)** qui, pour un nombre entier $n \geq 1$, retourne la valeur de R_n .

F4. Combien d'évaluations de l'application f nécessite cet algorithme pour une valeur de n donnée ?

Il faut déterminer une valeur de n pour laquelle R_n fournit une bonne estimation de I . La méthode 1 consiste à calculer les termes successifs de la suite R_n jusqu'à ce que celle-ci semble se stabiliser.

F5. Écrire une fonction **Integrale1(eps)** qui calcule, pour $\text{eps} > 0$, les termes successifs de la suite $(R_n)_{n \geq 1}$ jusqu'à ce qu'on obtienne $|R_n - R_{n-1}| < \text{eps}$, puis qui retourne la dernière valeur R_n calculée.

F6. Pour $\text{eps} = 10^{-7}$, **Integrale1(eps)** s'arrête pour $n = 34$. Vérifier que cela a nécessité plus de 500 évaluations de la fonction f .

F7. Quelle contrainte informatique empêche d'obtenir un résultat pertinent pour des valeurs arbitrairement petites de eps ?

Il est possible d'améliorer la méthode 1. La méthode 2 repose sur la relation de récurrence suivante :

$$\forall n \in \mathbb{N}^*, R_{2n} = \frac{R_n}{2} + \frac{\pi}{2n} \sum_{k=1}^n f\left(\frac{\left(k - \frac{1}{2}\right)\pi}{n}\right)$$

F8. Écrire la relation de récurrence pour un entier n de la forme $n = 2^{m-1}$. Écrire une fonction récursive **Rectangle2(m)** qui pour un nombre entier m retourne la valeur de R_{2^m} .

F9. Combien d'évaluations de la fonction f nécessite cet algorithme pour une valeur de m donnée ?

F10. Écrire une fonction **Integrale2(eps)** qui calcule, pour $\text{eps} > 0$, les termes successifs de la suite $(R_{2^m})_{m \geq 0}$ jusqu'à ce qu'on obtienne $|R_{2^m} - R_{2^{m-1}}| < \text{eps}$, puis qui retourne la dernière valeur R_{2^m} calculée. Cet algorithme doit impérativement être élaboré dans le but de minimiser le nombre total d'évaluations de f .

F11. Pour $\text{eps} = 10^{-7}$, **Integrale2(eps)** s'arrête pour $m = 7$. Combien d'évaluations de la fonction f ont été nécessaires ?

G Puissance rayonnée de l'antenne réelle du TAG

Dans les questions **C5.** et **C6.** de la première partie, on cherche à déterminer la résistance de rayonnement R_a de l'antenne. On souhaite proposer une façon de déterminer R_a dans un cadre plus général.

Une antenne est en réalité rarement filaire et l'expression analytique du champ électromagnétique produit est très difficile (voire impossible) à obtenir en tout point. Une antenne de TAG RFID est représentée figure 9.

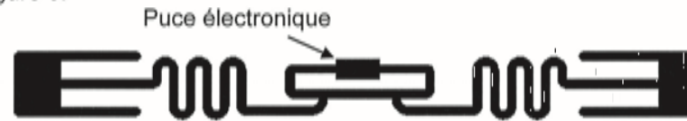


Figure 9

On considère une antenne centrée en l'origine dont les dimensions sont inférieures à 16 cm. Supposons qu'on ait estimé numériquement le champ électrique et le champ magnétique en tout point d'un maillage de pas 0,1 m contenant l'origine et pour des temps espacés d'un centième de période du courant d'alimentation. On a ainsi accès aux coordonnées cartésiennes (E_x, E_y, E_z) et (B_x, B_y, B_z) des champs $\vec{E}(M, t)$ et $\vec{B}(M, t)$ en chaque point $M = (x, y, z)$ de la forme $\left(\frac{a}{10}, \frac{b}{10}, \frac{c}{10}\right)$

où a, b et c désignent des nombres entiers et en tout instant t de la forme $\frac{p}{100f}$ où p est un nombre entier et $f = 860 \text{ MHz}$. Enfin, on supposera $I_0 = 1 \text{ A}$. Ainsi la valeur numérique de R_a est simplement donnée par la valeur de P_a , la puissance rayonnée par l'antenne.

L'objectif de cette partie est de proposer une démarche numérique permettant d'estimer R_a . On ne demande ni programme, ni justification mathématique des formules proposées.

G1. Proposer une façon d'estimer, à partir des données disponibles, la valeur moyenne temporelle $\langle \vec{\Pi}(M, t) \rangle$ du vecteur de Poynting dû à l'antenne en un point M du maillage.

G2. On souhaite maintenant estimer la puissance moyenne rayonnée par l'antenne à travers une surface fermée contenant l'antenne. Compte tenu des données disponibles, quelle est parmi les surfaces suivantes la plus pertinente pour obtenir une estimation précise de P_a

(et donc de R_a) : la surface du cube $\left[-\frac{1}{10}, \frac{1}{10}\right]^3$, celle du cube $[-5, 5]^3$, la sphère de centre

O et de rayon $\frac{1}{10}$ ou la sphère de centre O et de rayon 5 ?

G3. Notons $(\langle \Pi_x(M, t) \rangle, \langle \Pi_y(M, t) \rangle, \langle \Pi_z(M, t) \rangle)$ les coordonnées de la valeur moyenne du vecteur de Poynting en un point M du maillage. Proposer une formule permettant d'estimer numériquement la résistance de rayonnement R_a .

G4. Proposer une démarche, qui en faisant varier la surface choisie, permettrait de s'assurer une bonne précision de la valeur estimée de R_a .

Sujet CCINP 2024 PSI – Modélisation et ingénierie numérique

Calcul numérique série de Fourier

I.2 - Spectre du courant de ligne

La **figure 2** présente les chronogrammes de la tension du réseau $v(t)$ et du courant de ligne $i(t)$ mesurés expérimentalement en présence du redresseur à capacité en tête.

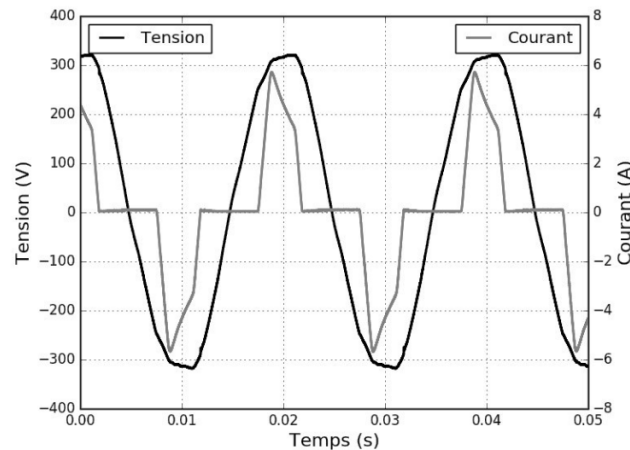


Figure 2 - Chronogrammes de la tension du réseau $v(t)$ et du courant de ligne $i(t)$

Un extrait du programme Python permettant de calculer le spectre du courant de ligne $i(t)$ est donné dans le **tableau 1**. Les bibliothèques numpy et pyplot sont supposées installées de la manière suivante : `import numpy as np` et `import matplotlib.pyplot as plt`. Le courant de ligne $i(t)$, échantillonné à une fréquence de 50 kHz par l'oscilloscope utilisé, est supposé chargé dans un tableau numpy 1D noté "courant".

Q3. Compléter les **lignes 2, 4, 6, 8, 10, 14, 15, 16, 17 et 18** de l'extrait de programme donné dans le **tableau 1**. On rappelle qu'un recueil de fonctions Python utiles est donné en annexe à ce sujet.

N°	Programme Python
1	# Fréquence d'échantillonnage
2	Fe = ...
3	# Nombre d'échantillons
4	N = ...
5	# Calcul de la transformée de Fourier discrète
6	fft_courant = ...
7	# Module de la transformée de Fourier discrète
8	mod_fft_courant = ...
9	# Vecteur fréquence
10	freq = ...
11	
12	# Tracé du spectre
13	plt.figure(1,figsize=(8,6))
14	plt.plot(...,color='gray',linewidth=2)
15	plt.xlim(...,...)
16	plt.xlabel(...,fontsize=16)
17	plt.ylim(...,...)
18	plt.ylabel(...,fontsize=16)
19	plt.tick_params(axis='both',labelsize=16)
20	plt.grid()
21	plt.show()

Tableau 1 - Programme de calcul du spectre du courant de ligne

Le développement en série de Fourier (D.S.F.) de $i(t)$ est de la forme :

$$i(t) = \sum_{n=1}^{+\infty} \hat{I}_n \cos(2\pi n f t + \varphi_n). \quad (2)$$

Q4. Donner les valeurs des coefficients $\hat{I}_n$ qui peuvent être déduites graphiquement du spectre en amplitude de $i(t)$ de la **figure 3**, obtenu à partir du programme. Que peut-on dire des harmoniques pairs ?

Q5. Retrouver par le calcul la constatation faite à la **Q4** en remarquant que le courant d'intensité $i(t)$ vérifie la symétrie de glissement : $i(t) = -i(t - T/2)$.

I.3 - Puissance moyenne fournie par le réseau

On note P la puissance **moyenne** fournie par le réseau à l'entrée du pont de Graetz. Comme ce dernier est constitué de quatre diodes supposées idéales, P est également la puissance moyenne reçue par la charge.

Q6. Établir l'expression de P en fonction de $\hat{V}$ et de certains paramètres du D.S.F. de $i(t)$. Commenter. On rappelle qu'un formulaire est donné en annexe.

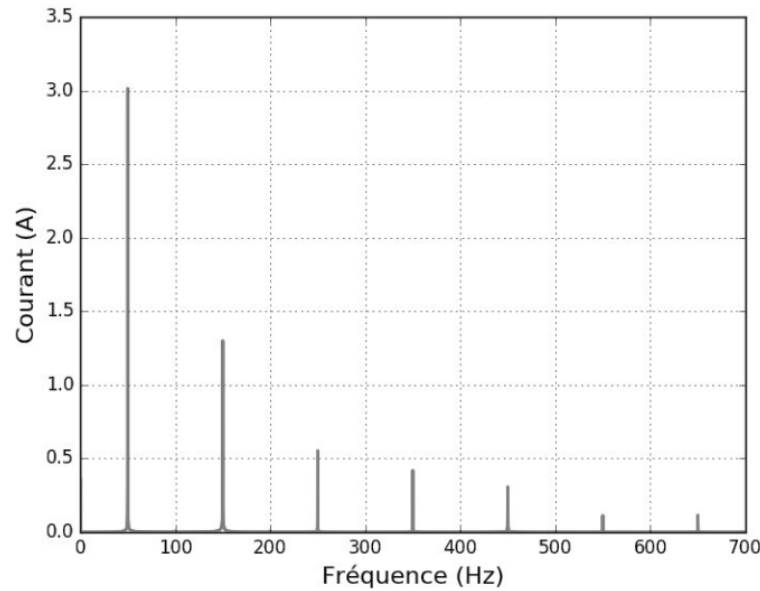


Figure 3 - Spectre en amplitude du courant

Q7. Calculer numériquement P en supposant que $\varphi_1 = 0$.

Pour tous réels a et b :

$$\cos(a + b) = \cos a \cos b - \sin a \sin b$$

$$\cos(a - b) = \cos a \cos b + \sin a \sin b$$

$$\sin(a + b) = \sin a \cos b + \sin b \cos a$$

$$\sin(a - b) = \sin a \cos b - \sin b \cos a$$

$$\cos a \cos b = \frac{1}{2} (\cos(a + b) + \cos(a - b))$$

$$\sin a \sin b = \frac{1}{2} (\cos(a - b) - \cos(a + b))$$

$$\sin a \cos b = \frac{1}{2} (\sin(a + b) + \sin(a - b))$$

Soit un signal $x(t)$, on notera $\langle x(t) \rangle$ la valeur moyenne temporelle de ce signal.

Si $x(t)$ est T-périodique alors :

$$\langle x(t) \rangle = \frac{1}{T} \int_0^T x(t) dt.$$

Pour tous réels f_1, f_2, φ_1 et φ_2 :

$$\langle \cos(2\pi f_1 t + \varphi_1) \rangle = 0 \quad \text{si} \quad f_1 \neq 0$$

$$\langle \sin(2\pi f_1 t + \varphi_1) \rangle = 0 \quad \text{si} \quad f_1 \neq 0$$

$$\langle \cos^2(2\pi f_1 t + \varphi_1) \rangle = \langle \sin^2(2\pi f_2 t + \varphi_2) \rangle = \frac{1}{2}$$

$$\langle \cos(2\pi f_1 t + \varphi_1) \cos(2\pi f_2 t + \varphi_2) \rangle = 0 \quad \text{si} \quad f_1 \neq f_2$$

$$\langle \sin(2\pi f_1 t + \varphi_1) \sin(2\pi f_2 t + \varphi_2) \rangle = 0 \quad \text{si} \quad f_1 \neq f_2$$

$$\langle \cos(2\pi f_1 t + \varphi_1) \sin(2\pi f_2 t + \varphi_2) \rangle = 0 \quad \text{si} \quad (f_1, f_2) \neq (0, 0)$$

Sous des conditions de régularité qui seront supposées vérifiées dans ce sujet, un signal $x(t)$ T-périodique peut s'écrire :

$$x(t) = \langle x(t) \rangle + \sum_{n=1}^{+\infty} \hat{X}_n \cos(2\pi n f t + \varphi_n).$$

Cette expression est appelée *développement en série de Fourier* (D.S.F.) du signal $x(t)$.

Les amplitudes $\hat{X}_n$ du développement en série de Fourier de $x(t)$ vérifient l'identité de Parseval :

$$\langle x(t)^2 \rangle = \langle x(t) \rangle^2 + \frac{1}{2} \sum_{n=1}^{+\infty} \hat{X}_n^2.$$

ANNEXE 2 - FONCTIONS PYTHON UTILES

La bibliothèque numpy est supposée installée de la manière classique - `import numpy as np`.

np.dot (A, B) : retourne le produit de deux matrices $[A]$ et $[B]$.

A.T : retourne la transposée d'une matrice $[A]$.

len(X) : retourne le nombre d'éléments de l'objet X. Ce dernier peut être une liste ou un tableau numpy.

np.abs(X) : retourne un tableau dont les éléments correspondent aux modules des éléments du tableau X.

np.arange(a, b, p) : retourne un tableau 1D dont les éléments sont répartis avec un pas p entre a et b, a est le premier élément du tableau, le dernier élément est strictement inférieur à b. Si l'argument p n'est pas renseigné, il est pris par défaut égal à 1.

np.fft.rfft(X) : retourne, sous forme d'un tableau 1D de nombres complexes, la transformée de Fourier discrète du signal échantillonné correspondant à X, un tableau 1D dont les éléments doivent être réels. Les éléments du tableau retourné correspondent aux fréquences $n \frac{F_e}{N}$ avec $n \in \llbracket 0, \lfloor \frac{N}{2} \rfloor \rrbracket$, N le nombre d'éléments de X et F_e la fréquence d'échantillonnage. Enfin, le facteur multiplicatif $2/N$ doit être appliqué au tableau 1D retourné pour obtenir le spectre du signal continu correspondant au signal échantillonné contenu dans X.

np.linspace(a, b, N) : retourne un tableau 1D de N éléments répartis linéairement entre a et b, a est le premier élément du tableau, b le dernier.

plt.plot(X, Y) : permet de tracer le tableau 1D Y en fonction du tableau 1D X. Les deux tableaux doivent avoir le même nombre d'éléments. Des arguments optionnels permettent de choisir la couleur du tracé, l'épaisseur du trait, etc.

plt.xlim(x_min, x_max) : permet de limiter un tracé aux abscisses comprises entre x_min et x_max.

plt.xlabel('titre abscisses') : permet d'attribuer un titre à l'axe des abscisses. Des arguments optionnels permettent de choisir la taille de la police, la couleur du texte, etc.

plt.ylim(y_min, y_max) : permet de limiter un tracé aux ordonnées comprises entre y_min et y_max.

plt.ylabel('titre ordonnées') : permet d'attribuer un titre à l'axe des ordonnées. Des arguments optionnels permettent de choisir la taille de la police, la couleur du texte, etc.

Sujet CCINP 2021 PSI – Modélisation et ingénierie numérique
Schéma d'Euler et dichotomie

Thermique	Électrocinétique
Température	Potentiel électrique
Différence de température	Tension
Flux ou puissance thermique	Courant électrique
Résistance thermique	Résistance électrique
Conductance thermique	Conductance électrique
Capacité thermique	Capacité
Source de chaleur ou thermostat	Source de tension
Source de puissance thermique	Source de courant

Tableau 2 - Analogie diffusion thermique-électrocinétique

Partie IV - Influence du débit d'air conditionné sur la température

L'objectif de cette partie est d'établir un modèle simple de l'évolution de la température T d'une pièce en fonction du débit d'air conditionné D_m . On suppose que la pièce dispose d'une fenêtre donnant sur l'extérieur et qu'elle est, comme à la **Q6**, occupée par quatre personnes.

Les lois de la thermique, analogues aux lois de l'électrocinétique (loi des mailles, des nœuds, ponts diviseurs de tension et de courant, associations série et parallèle), pourront être utilisées sans démonstration. L'analogie est rappelée dans le **tableau 2**.

IV.1 - Schéma électrique équivalent de la pièce

Le schéma électrique équivalent de la pièce étudiée est donné sur la **figure 4**. Il comporte :

- une source de tension T_e correspondant à l'extérieur du bâtiment ;
- une association R_1 - C_1 - R_1 modélisant le comportement thermique des murs séparant la pièce considérée de l'extérieur du bâtiment ;
- une association R_2 - C_2 modélisant le comportement thermique du mobilier de la pièce ;
- une source de tension T_c associée à une résistance variable R_3 , l'ensemble modélisant le système de climatisation ;
- une capacité C_3 modélisant l'air à l'intérieur de la pièce ;
- une résistance R_4 modélisant la fenêtre et l'échangeur ;
- une source de courant η .

Par souci de simplification, les échanges thermiques avec les autres pièces du bâtiment ne sont pas pris en compte.

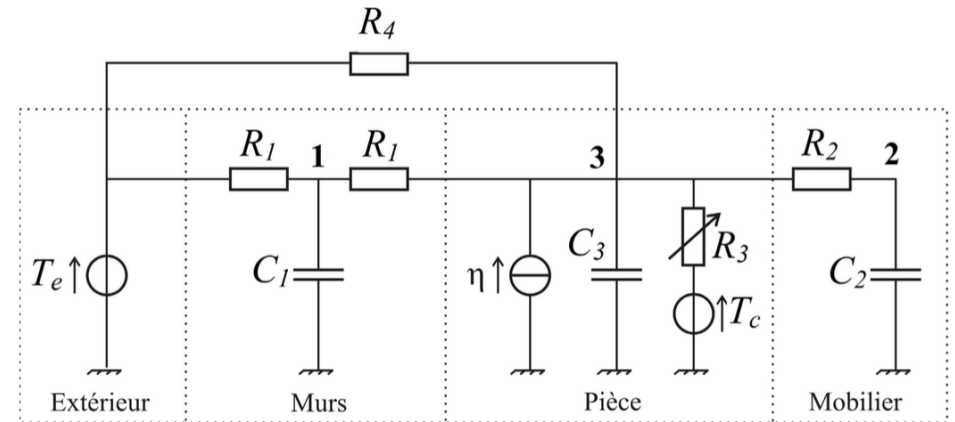


Figure 4 - Schéma électrique équivalent de la pièce

Les points **1** et **2** correspondent à des températures fictives notées respectivement T_1 et T_2 . Le point **3** correspond pour sa part à T , la température de la pièce.

Q17. Exprimer R_4 en fonction des résistances thermiques de l'échangeur et de la fenêtre. Justifier.

Q18. À quoi correspond la source de courant η ? Justifier. On rappelle que la pièce est supposée être occupée par quatre personnes.

Q19. En écrivant la relation (2) pour une seule pièce du bâtiment, donner l'expression de la résistance R_3 . On définira avec soin la (ou les) grandeur(s) introduite(s). Pourquoi s'agit-il d'une résistance variable pour le système de climatisation DAV étudié ici ?

Q20. En appliquant la loi des nœuds aux points **1**, **2** et **3**, établir trois équations faisant intervenir T_1 , T_2 , T et leurs dérivées.

Q21. Montrer que la modélisation adoptée conduit à l'équation différentielle matricielle :

$$\dot{X} = A X + B U \quad (7)$$

$$\text{avec } X = (T_1 \quad T_2 \quad T)^T \quad \text{et} \quad U = (T_e \quad T_c \quad \eta)^T. \quad (8)$$

On explicitera avec soin les matrices A et B . M^T désigne la transposée de la matrice M .

IV.2 - Résolution numérique

L'objectif de cette sous-partie est de résoudre numériquement l'équation matricielle (7) à l'aide de la méthode d'Euler, dans le but de déterminer l'évolution temporelle de la température de la pièce T . On définit pour cela les instants $t_k = k \Delta t$ avec $\forall k \in \llbracket 0, K \rrbracket$ où Δt est le pas de calcul, fixé à 6 secondes par la suite, et K un entier naturel.

Le programme Python correspondant est donné dans le **tableau 3**. Les **lignes 5** et **6** définissent les matrices A et B pour la pièce considérée et dans le cas où le débit d'air conditionné dans la pièce est fixé à sa valeur maximale de $3,5 \cdot 10^3 \text{ kg} \cdot \text{h}^{-1}$. Les **lignes 7**, **10**, **21**, **22** et **23** sont manquantes.

La température à l'extérieur du bâtiment vaut $T_e = 28 \text{ }^\circ\text{C}$, la température de l'air conditionné est toujours $T_c = 20 \text{ }^\circ\text{C}$ et la valeur de la source de courant est $\eta = 0,50 \text{ kW}$.

Q22. Compléter la **ligne 7** définissant le vecteur U .

Q23. On souhaite initialiser les températures T_1 , T_2 et T à $28 \text{ }^\circ\text{C}$. Compléter la **ligne 10** initialisant le vecteur X .

Q24. Approcher $X(t_{k+1})$ à l'aide de $X(t_k)$, $\dot{X}(t_k)$ et Δt en utilisant la méthode d'Euler. À quelle condition cette approximation est-elle justifiée ?

Q25. Compléter les **lignes 21** à **23**. Les éléments des listes " temps " (t) et " température " (T) devront respectivement correspondre à des minutes et des degrés Celsius. On pourra utiliser $np.dot(X, Y)$ pour calculer le produit de la matrice X par la matrice Y .

N°	Programme Python
1	import numpy as np
2	import matplotlib.pyplot as plt
3	
4	# Définition des matrices A, B et du vecteur U
5	A=np.array([[-8.4e-6,0,4.2e-6],[0,-2.95e-5,2.95e-5],[2.6e-3,4.7e-4,-1.08e-2]])
6	B=np.array([[4.2e-6,0,0],[0,0,0],[3.2e-5,7.74e-3,7.9e-6]])
7	...
8	
9	# Initialisation du vecteur X
10	...
11	
12	# Pas de calcul fixé à 6 secondes
13	dt=6
14	
15	# Définition des listes "temps" et "température"
16	t=[]
17	T=[]
18	
19	# Itération de la méthode d'Euler
20	for i in range(100) :
21	...
22	...
23	...
24	
25	# Affichage
26	plt.plot(t,T)
27	plt.ylim(min(T)-1,max(T)+1)
28	plt.ylabel("Température en °C")
29	plt.xlabel("Temps en min")
30	plt.grid(True)
31	plt.show()

Tableau 3 - Résolution numérique via la méthode d'Euler

IV.3 - Résultat de simulation et identification à un premier ordre

Le programme élaboré précédemment est modifié de manière à pouvoir tenir compte des variations dans le temps de la source de courant η et du débit massique d'air conditionné injecté dans la pièce D_m . On considère une situation où :

- à $t = 0$, la pièce est depuis longtemps inoccupée et non-climatisée ;
- à $t = 10 \text{ min}$, quatre personnes entrent dans la pièce ;
- à $t = 40 \text{ min}$, le registre à volets qui contrôle l'arrivée d'air conditionné est ouvert en grand, le débit d'air conditionné est alors à son maximum, soit $3,5 \cdot 10^3 \text{ kg} \cdot \text{h}^{-1}$;
- à $t = 80 \text{ min}$, la climatisation est coupée et les quatre personnes sortent de la pièce.

La température à l'extérieur du bâtiment et la température de l'air conditionné valent toujours respectivement $T_e = 28 \text{ }^\circ\text{C}$ et $T_c = 20 \text{ }^\circ\text{C}$.

L'évolution simulée de la température de la pièce en fonction du temps, pour la situation considérée, est donnée **figure 5**.

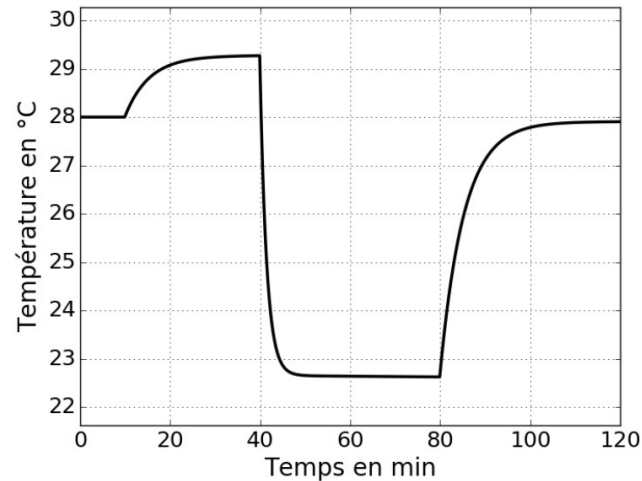


Figure 5 - Simulation de l'évolution temporelle de la température de la pièce

Q26. Représenter graphiquement les évolutions temporelles de la source de courant η et du débit massique d'air conditionné D_m sur les **DR1** et **DR2** du **Document Réponse**.

L'ouverture en grand du registre à volets à $t = 40$ min permet de simuler la réponse indicielle du système S dont l'entrée est le débit massique d'air conditionné et la sortie, la variation de température correspondante. On se propose de déterminer un modèle de comportement de S . Celui-ci, supposé linéaire et invariant, sera par conséquent caractérisé par sa fonction de transfert :

$$H(p) = \frac{\Delta T(p)}{D_m(p)} \quad (9)$$

avec $D_m(p)$ et $\Delta T(p)$ les transformées de Laplace respectivement du débit massique d'air conditionné et de la variation de température.

Le **DR3** présente l'évolution temporelle de la température de la pièce déjà donnée par la **figure 5**, mais sur l'intervalle de temps plus réduit allant de $t = 39$ min à $t = 50$ min.

Q27. Justifier qu'un modèle de comportement de type passe-bas du premier ordre semble approprié pour S .

On posera donc dans la suite :

$$H(p) = \frac{H_0}{1 + \tau p} \quad (10)$$

Q28. À partir du **DR3**, identifier, en explicitant clairement la démarche suivie et en adoptant les unités du Système International, le gain statique H_0 et la constante de temps τ du modèle de comportement de S .

Partie V - Contrôle du débit d'air conditionné

L'objectif de cette partie est de modéliser la relation entre l'action du moteur sur les volets du registre et le débit qui va en résulter. Le registre est présenté **figure 6** et sur l'**annexe**.



Figure 6 - Registre à volets

Le registre sert au réglage du débit d'air, par création d'une perte de charge variable, qui n'est pas directement proportionnelle à l'angle de pivotement des volets. Dans le modèle proposé, un seul moteur va piloter l'ensemble des volets grâce à un ensemble de biellettes qui les relie.

Les étapes de la modélisation sont listées ci-dessous :

- modéliser la relation débit-angle à partir de la caractéristique du ventilateur et des conduites ;
- modéliser la relation entre l'angle de rotation du moteur et celui des volets ;
- résoudre numériquement et linéariser le modèle obtenu ;
- modéliser l'action mécanique de l'air sur un volet à partir des résultats d'une expérimentation ;
- modéliser la relation entre le couple moteur et l'angle d'inclinaison des volets.

V.1 - Étude du débit en fonction de l'inclinaison des volets du registre

La **figure 7** montre la caractéristique aérolitique pour des angles d'inclinaison θ de (0° , 15° , 30° , 45° , 60°). L'angle θ correspond à l'inclinaison du volet central. Lorsque celui-ci est à l'horizontale, l'angle est de 0° . Pour le registre fermé, le volet est à la verticale ($\theta = 90^\circ$). Le schéma cinématique du registre à volets est donné dans l'**annexe**.

Le point de fonctionnement est l'intersection entre la caractéristique du ventilateur et celle de la conduite. Par exemple, pour une vitesse de rotation de 3 000 tr/min et une inclinaison de 30° des volets, le débit sera de 1 000 m³/h.

Afin de minimiser les pertes d'énergie et équilibrer les différentes sorties, il faut travailler à pression constante. La pression de fonctionnement choisie est de 70 Pa. Cette pression est relative par rapport à la pression atmosphérique. Il est aussi possible de l'appeler surpression.

Q29. En utilisant les différents points de fonctionnement de la **figure 7**, tracer sur le **DR4** la caractéristique de débit en fonction de l'angle d'inclinaison des volets. Sachant qu'il y a deux ventilateurs qui doublent le débit, proposer une modélisation affine sous la forme :

$$D_m = K_R \theta + D \quad (11)$$

et donner les valeurs numériques de D et K_R .

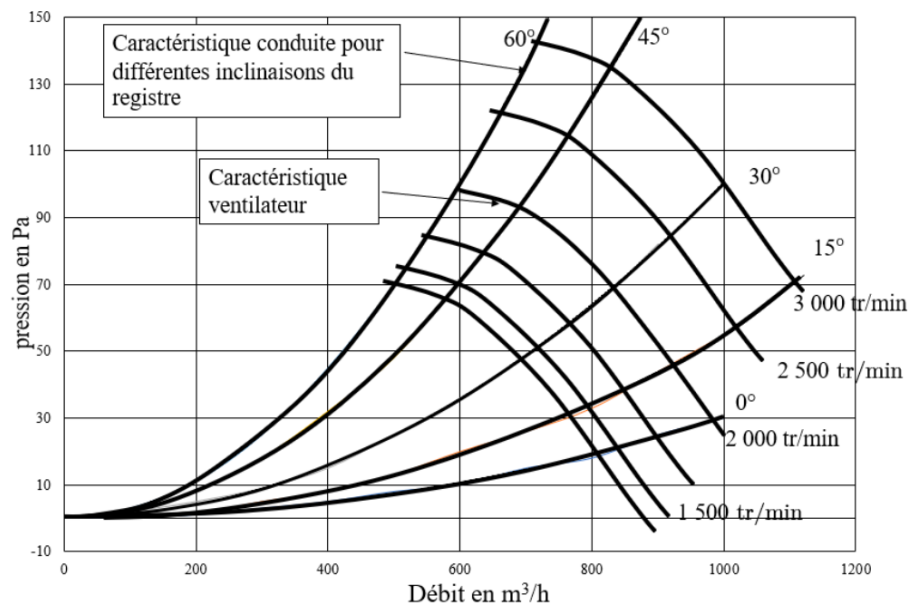


Figure 7 - Caractéristique aérolique

V.2 - Modélisation de la relation entre l'angle des volets et l'angle moteur

Le schéma cinématique du registre à volets est donné sur l'**annexe**. Le moteur entraîne le volet 1.

Q30. Écrire la fermeture géométrique de la boucle 0-1-4-3-0 en projection sur $\vec{x}_0$ et $\vec{y}_0$. Éliminer le paramètre θ_4 . Mettre le résultat sous la forme $f(\theta_1, \theta_3) = 0$.

Q31. La résolution de cette équation non-linéaire est effectuée informatiquement. Ainsi, pour la plage de valeur de $\theta_1 \in [0, 90^\circ]$, il faut résoudre l'équation $f(\theta_3) = 0$. La résolution de cette équation est obtenue en utilisant la technique de la dichotomie. Recopier et compléter les **lignes 15 à 18** de la fonction dichotomie du **tableau 4**.

Le résultat de la simulation numérique est donné dans la **figure 8**.

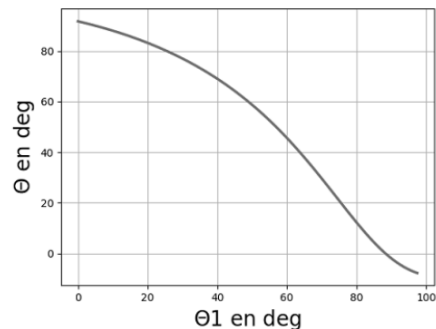
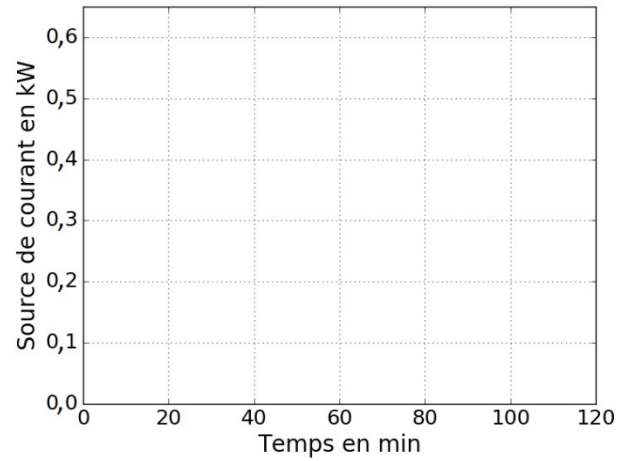


Figure 8 - Résultat simulation

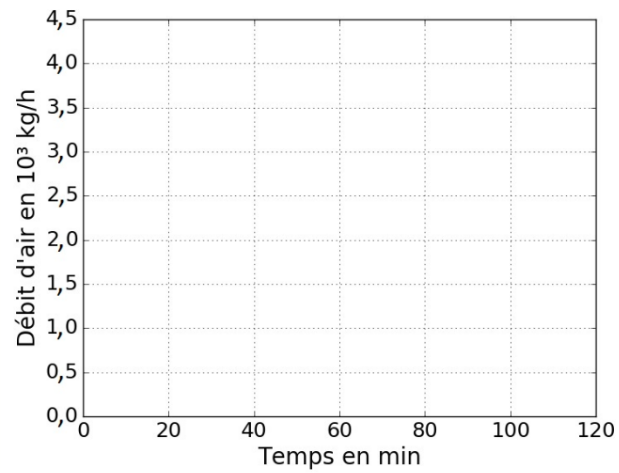
N°	Programme Python
1	import numpy as np
2	import matplotlib.pyplot as plt
3	from math import *
4	
5	# Définition de la fonction f(theta3)
6	def f(theta3) :
7	s=(2.5-1.3*sin(theta1)-1.8*sin(theta3))*2+(1.8*cos(theta3)+1.3*cos(theta1))*2-2.8**2
8	return s
9	#liste de valeurs pour theta1
10	abscisse = np.linspace(0, 1.7, 100)
11	abscissedeg=(180/pi)*abscisse
12	#résolution par dichotomie avec a<b
13	def Dichotomie(f,a,b,eps) :
14	if f(b)*f(a)>=0 :return None
15	while...
16	...
17	...
18	...
19	return a
20	ordonnee_scipy=[]
21	for i in range(len(abscisse)) :
22	theta1=abscisse[i]
23	theta3=dichotomie(f,-5,5,0.1)
24	thetadeg=(180/pi)*theta3+45
25	ordonnee_scipy.append(thetadeg)
26	#Tracé du résultat
27	plt.legend()
28	plt.grid(True)
29	plt.xlabel('Θ1 en deg')
30	plt.ylabel('Θ en deg')
31	plt.title('loi entree sortie')
32	plt.show()

Tableau 4 - Programmation de la technique de la dichotomie

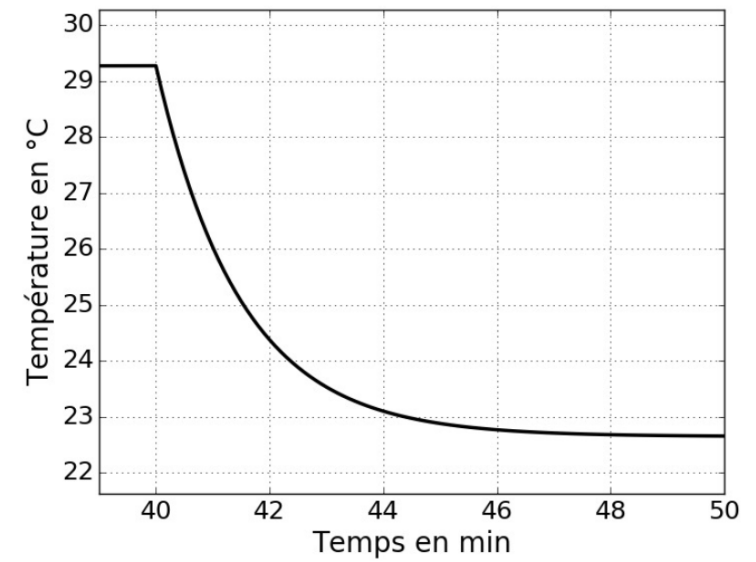
DR1 - Q26 - Évolution temporelle de la source de courant η



DR2 - Q26 - Évolution temporelle du débit massique d'air conditionné D_m



DR3 - Q28 - Évolution de la température de la pièce entre 39 min et 50 min



Sujet CCINP 2020 PSI – Modélisation et ingénierie numérique
Schéma d'Euler - Résolution de l'équation de Laplace $\Delta T(x,y) = 0$

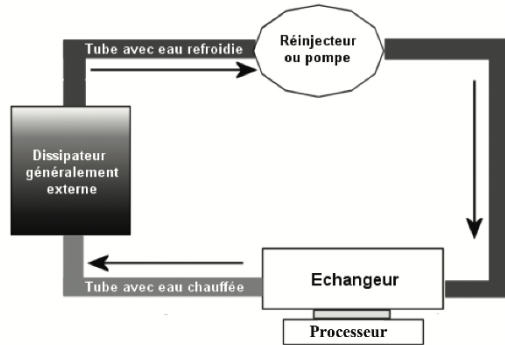


Figure 13 - Schématisation d'un système de watercooling

On s'intéresse ici à un échangeur thermique (**figure 14**) constitué d'un ensemble de micro-canaux percés dans le corps de l'échangeur et d'ailettes de refroidissement. Ces dispositifs sont fabriqués dans des métaux conduisant la chaleur. Les plus rencontrés sont le cuivre de conductivité thermique $401 \text{ W}\cdot\text{K}^{-1}\cdot\text{m}^{-1}$ et l'aluminium de conductivité thermique $237 \text{ W}\cdot\text{K}^{-1}\cdot\text{m}^{-1}$). L'eau circule dans les canaux et emporte l'énergie thermique produite par le processeur.

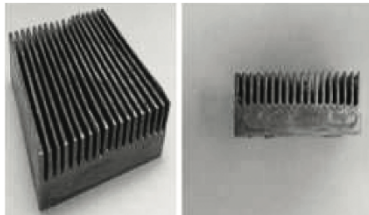


Figure 14 - Photographie d'un échangeur thermique à micro-canaux (sans son "courvercle")

Q22. Lequel des deux métaux (aluminium ou cuivre) est le plus adapté pour réaliser l'échangeur thermique. Quel est son inconvénient ?

Q23. Le débit massique D_m de l'écoulement à travers l'échangeur est de l'ordre de $10^{-2} \text{ kg}\cdot\text{s}^{-1}$. À l'aide d'un bilan d'énergie sur un système bien choisi, estimer la température du fluide en sortie de l'échangeur pour une puissance thermique dissipée de 100 W. On rappelle la valeur de la capacité thermique massique de l'eau liquide $c_e = 4,18 \text{ kJ}\cdot\text{K}^{-1}\cdot\text{kg}^{-1}$.

II.3 - Détermination du champ de température dans l'échangeur

On souhaite maintenant déterminer le nombre nécessaire d'ailettes de refroidissement ainsi que la température dans ces ailettes en régime permanent.

Nous allons pour cela déterminer le champ de température dans l'échangeur et en particulier dans la base métallique en contact avec le processeur (**figure 15**).

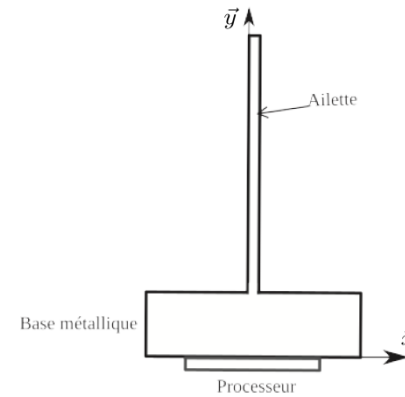


Figure 15 - Schématisation de l'échangeur et d'une ailette de refroidissement

Dans cette partie du système, l'équation vérifiée par le champ de température en régime permanent est :

$$\frac{\partial^2 T(x,y)}{\partial y^2} + \frac{\partial^2 T(x,y)}{\partial x^2} = 0.$$

Le problème est toujours supposé invariant selon l'axe (Oz). Pour illustrer le principe de la méthode, l'ailette est modélisée par un rectangle. Considérons le rectangle **figure 16** dont trois côtés sont à la température T_1 et le dernier à la température T_2 . Ce rectangle est découpé en 8 petits carrés de côté h (appelé pas de discrétisation) dont les sommets forment un réseau de points de coordonnées (ih, jh) auxquels correspondent les températures $T(ih, jh)$ que l'on mémorise dans un tableau $T_{(i,j)}$. L'objectif est de déterminer les températures $T_{(1,1)}$, $T_{(1,2)}$ et $T_{(1,3)}$ aux points de coordonnées (h, h) , $(h, 2h)$ et $(h, 3h)$.

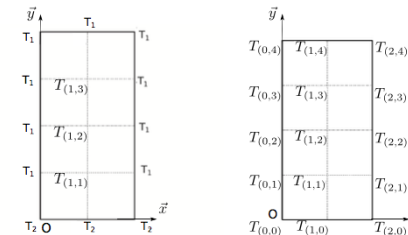


Figure 16 - Discrétisation de l'ailette

La méthode des différences finies centrées avec le pas de h est utilisée pour déterminer le champ de température dans l'échangeur et en particulier dans la base métallique en contact avec le processeur.

$$\frac{\partial T}{\partial x}(x, y) \cong \frac{T_{(x+\frac{h}{2}, y)} - T_{(x-\frac{h}{2}, y)}}{h}$$

$$\frac{\partial T}{\partial y}(x, y) \cong \frac{T_{(x, y+\frac{h}{2})} - T_{(x, y-\frac{h}{2})}}{h}$$

Q24. Selon une logique analogue, expliquer comment approcher $\frac{\partial^2 T}{\partial x^2}(x, y)$ et $\frac{\partial^2 T}{\partial y^2}(x, y)$, d'abord à partir des dérivées premières de $T(x, y)$, puis sans aucune dérivée. Les expressions finales pourront faire intervenir $T(x+h, y)$, $T(x-h, y)$, $T(x, y+h)$, $T(x, y-h)$ et $T(x, y)$.

Q25. En déduire les coefficients a , b , c , d et e de l'équation différentielle sous forme discrétisée suivante :

$$a \cdot T_{(i-1, j)} + b \cdot T_{(i+1, j)} + c \cdot T_{(i, j-1)} + d \cdot T_{(i, j+1)} + e \cdot T_{(i, j)} = 0.$$

Q26. Écrire les équations pour $i=1$, $j=1$, pour $i=1$, $j=2$ et pour $i=1$, $j=3$.

Q27. En déduire le système d'équations à résoudre, le mettre sous forme de deux matrices A et B tel que :

$$A \cdot T = B$$

$$T = \begin{bmatrix} T_{(1,1)} \\ T_{(1,2)} \\ T_{(1,3)} \end{bmatrix}$$

S

ujet

avec A une matrice 3×3 et B un vecteur de 3 composantes.

La discrétisation de la base métallique n'est pas suffisamment fine et ne prend pas en compte les effets à l'interface ailette/embase.



Figure 17 - Calcul sur différentes géométries

Pour permettre d'obtenir une bonne approximation du champ de température T sur l'ensemble de la structure (Ω) et de connaître le pas de discrétisation optimal, il est nécessaire de calculer l'énergie interne pour différentes discrétisations. L'énergie interne s'exprime par :

$$E_i = \frac{1}{2} \iint_{\Omega} \lambda \nabla T \cdot \nabla T d\Omega.$$

Il est possible de montrer qu'en prenant une discrétisation très fine (non demandé), l'énergie interne va tendre vers une valeur exacte (E_i) correspondant à une valeur calculée numériquement pour une discrétisation suffisamment fine.

On note E_i^n l'énergie interne approchée calculée numériquement en fonction du nombre d'inconnues de discrétisation n . On peut ensuite tracer l'erreur sur l'énergie interne $(E_i^n - E_i)/E_i^n$ en fonction de n en échelle log (**figure 18**). Pour ne pas effectuer ensuite des calculs trop importants, on choisit une valeur de discrétisation n correspondant à une erreur de 5 %.

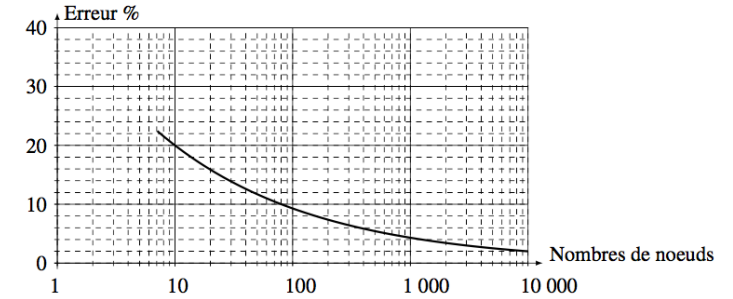


Figure 18 - Erreur sur l'énergie interne $(E_i^n - E_i)/E_i^n$ en fonction du nombre de nœuds n

Q28. Déterminer la valeur de n correspondant à l'énergie interne permettant d'assurer que la discrétisation du domaine Ω soit suffisamment fine pour avoir une bonne approximation du champ de température $T(x, y)$ sur l'ensemble de la structure.

Après discrétisation sur un modèle plus complexe (**figure 17**), le système à résoudre est toujours du type $A \cdot T = B$ mais la dimension de la matrice A devient très élevée. La matrice obtenue est une matrice tridiagonale, pentadiagonale par morceau. Plusieurs résolutions sont effectuées autour de quelques températures critiques tel que la résolution devienne :

$$A \cdot (T + \delta T) = (B + \delta B).$$

L'objectif de cette résolution est de regarder l'influence de cette incertitude δB sur la température finale obtenue.

Il est possible de montrer que la variation du premier membre sur le résultat est majorée par la variation du second membre sur les données multipliées par $\|A\| \cdot \|A^{-1}\|$, c'est-à-dire :

$$\frac{\|\delta T\|}{\|T\|} = \left(\|A\| \cdot \|A^{-1}\| \right) \frac{\|\delta B\|}{\|B\|}.$$

Le nombre $\|A\| \cdot \|A^{-1}\|$, appelé $\text{cond}(A)$, définit un bon conditionnement de la matrice A , c'est-à-dire une faible influence des variations de B sur le résultat de T .

Q29. Parmi les valeurs $\{-10\,000, -1\,000, -1, 1, 1\,000, 10\,000\}$ pour $\text{cond}(A)$, donner la valeur de $\text{cond}(A)$ qui entraînera un bon conditionnement de notre matrice.

La norme de A est définie par :

$$\|A\|_{\infty} = \max_{1 \leq i \leq m} \sum_{j=1}^n |A_{ij}|.$$

Partie III - Étude du moteur du vérin électrique

Une partie représentative de la matrice obtenue est représentée ci-dessous :

$$A = \begin{bmatrix} 1 & 2 & 6 & 8 \\ 2 & 5 & 15 & 23 \\ 6 & 15 & 46 & 73 \\ 8 & 23 & 73 & 130 \end{bmatrix}$$

et la matrice inverse :

$$A^{-1} = \begin{bmatrix} 41 & 28 & -24 & 6 \\ 28 & 35 & -23 & 5 \\ -24 & -23 & 17 & 4 \\ 6 & 5 & -4 & 1 \end{bmatrix}.$$

Q30. Calculer $\|A\|_{\infty}$ et $\|A^{-1}\|_{\infty}$.

Q31. Calculer $\text{cond}(A)$. Est ce bien conditionné ?

CCINP

CCINP 2025 PSI – Modélisation

Tirage variable aléatoire - Monté Carlo - Calcul de moyennes, écart-type, régression linéaire

Le moteur alimenté par le hacheur de la partie précédente est celui du vérin électrique. Les concepteurs ont choisi le modèle de vérin électrique RE1004-Strong3 dont le principe de fonctionnement est le suivant : un moteur à courant continu (moteur cc), associé à un réducteur, met en rotation une vis qui entraîne un écrou lié au siège en translation.

L'objectif de cette partie est d'étudier les pertes du vérin à partir des essais expérimentaux.

On peut modéliser le fonctionnement du vérin par le schéma fonctionnel (**figure 5**) dans lequel I est l'intensité du courant consommé par le moteur cc, Ω_1 la vitesse angulaire du rotor, Ω_2 la vitesse angulaire de la vis sans fin et v la vitesse de translation du vérin.

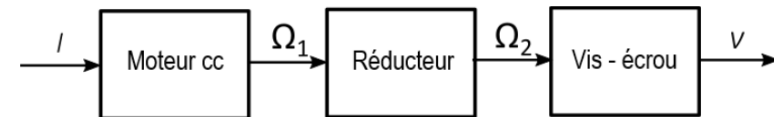


Figure 5 - Schéma fonctionnel du vérin électrique

On dispose des paramètres constructeur suivants :

- le rapport de réduction du réducteur est $r = 1/10$;
- le pas de la vis est $p = 3,0$ mm.

On se propose tout d'abord de déterminer la constante de couplage électromécanique Φ du moteur à courant continu et sa résistance interne R à partir des courbes expérimentales (**figure 6**). Pour les essais, la tension d'entrée U est stabilisée à 24 V et le régime est permanent.

Q11. Définir le rendement global du vérin électrique η_g en fonction de U , I , F_{verin} et de v , où F_{verin} est la force exercée par la tige en translation.

Q12. Donner l'expression de la constante μ , telle que $\Omega_1 = \mu v$, en fonction du pas p et du rapport de réduction r .

Dans le programme Python (**annexe 1**) figurent les valeurs expérimentales du courant I et de la vitesse v pour différentes valeurs de la force F_{verin} , ainsi que les incertitudes de mesure. Les valeurs de Ω_1 et de l'incertitude-type associée sont calculées par l'intermédiaire de la constante μ définie à la question **Q12**.

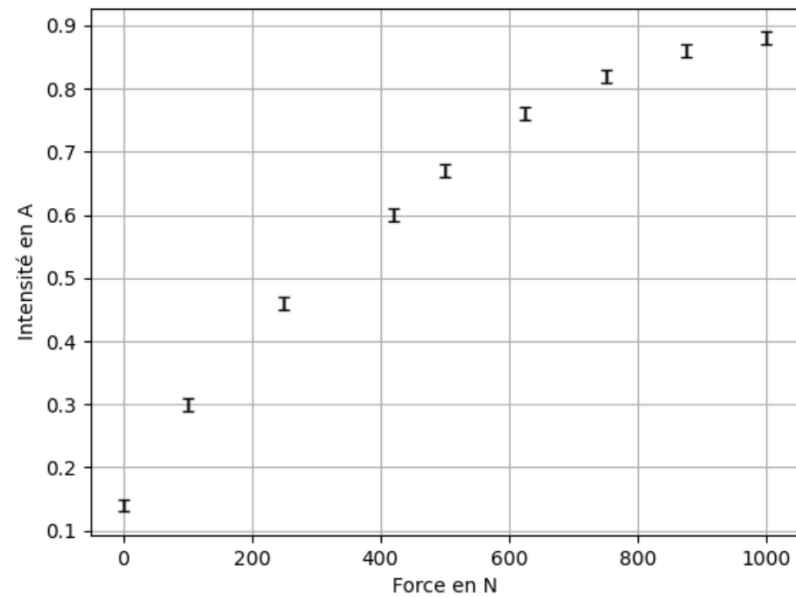
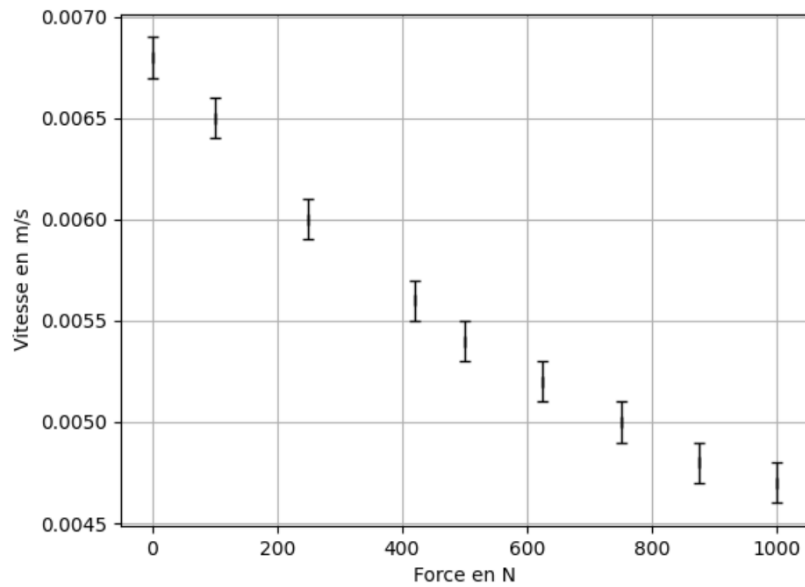


Figure 6 - Courbes d'évolution de la vitesse de translation de la tige du vérin v et du courant absorbé I en fonction de la force F_{verin}

Q13. Donner le schéma électrique équivalent du moteur à courant continu fonctionnant en régime permanent (la force contre-électromotrice sera notée E'). En déduire l'équation électrique du moteur, puis la relation :

$$\Omega_1 = \frac{U}{\Phi} - \frac{R}{\Phi} I$$

La **figure 7** donne le tracé de Ω_1 en fonction de I , réalisé dans le programme Python, ainsi que les barres d'erreurs. La régression linéaire passe par les barres d'erreur, ce qui valide la modélisation. Il reste à en déduire les valeurs de R et de ϕ ainsi que leurs incertitudes-type.

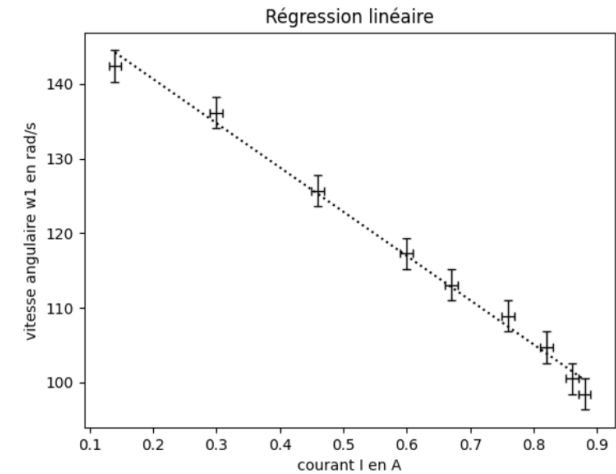


Figure 7 - Vitesse angulaire du moteur en fonction de l'intensité absorbée par le moteur

Q14. Expliquer brièvement le principe de la méthode utilisée pour déterminer l'incertitude-type sur a et b dans la fonction f définie au début du programme. Comment s'appelle cette méthode ?

Q15. Compléter le programme Python (**annexe 1**) pour qu'il calcule R , Φ ainsi que les incertitudes-type associées $u(R)$ et $u(\Phi)$. Utiliser un appel de la fonction f (avec $N = 10\,000$ points) et les formules usuelles de propagation des incertitudes-type (**annexe 2**). On précise que l'incertitude-type sur la tension U est négligeable.

Voici les résultats obtenus lorsque le programme est lancé :

résistance interne $R = 9.332291636352243$ incertitude $u(R) = 0.4770444850736393$
 constante de couplage $\Phi = 0.15733967170583427$ incertitude $u(\Phi) = 0.001985407699473084$

Q16. Donner les résultats des mesures et des incertitudes (avec les unités) en ne gardant qu'un seul chiffre significatif pour les incertitudes.

ANNEXE 1

Programme Python

Les points d'interrogation (?) situés dans le programme sont volontaires. Ils permettent de masquer certaines réponses à des questions du sujet.

```
# -*- coding : utf-8 -*-

# Bibliothèques

import numpy as np
import matplotlib.pyplot as plt
import numpy.random as rd

# fonction régression linéaire de y en fonction de x, et sa documentation

def f(x,y,ux,uy,N):
    a,b = np.polyfit(x,y,1)
    asim = np.zeros(N); bsim = np.zeros(N)
    for i in range(N):
        xsim = x + rd.normal(0,ux)
        ysim = y + rd.normal(0,uy)
        asim[i],bsim[i] = np.polyfit(xsim,ysim,1)
    ua = np.std(asim,ddof=1)
    ub = np.std(bsim,ddof=1)
    return(a,b,ua,ub)

"""
x et y sont les tableaux numpy d'entrée
ux et uy sont les incertitudes-types associées
a et b sont les résultats de la régression linéaire
a est la pente et b l'ordonnée à l'origine
ua et ub sont les incertitudes-types sur a et b
rd.normal(0,1) : nombre aléatoire tiré selon une loi gaussienne
                  centrée sur 0 et d'écart type 1
np.std(tab, ddof = 1) : écart type des éléments du tableau numpy tab
"""

# Constantes

U = 24 # tension moteur
pas = 3e-3 # pas de la vis
r = 1/10 # rapport de réduction du réducteur

# Données issues des mesures expérimentales
```

```
F_verin = np.array([0,100,250,420,500,625,750,875,1000]) # force
v = 1e-3*np.array([6.8,6.5,6.0,5.6,5.4,5.2,?,4.8,4.7]) # vitesse
I = np.array([0.14,0.3,0.46,0.6,0.67,0.76,?,0.86,0.88]) # courant

n = len(v)
uv = 0.1e-3*np.ones(n) # incertitude de 0.1 mm/s sur la vitesse
ul = 0.01*np.ones(n) # incertitude de 10 mA sur le courant

# Calcul de la vitesse angulaire du moteur w1
# et de l'incertitude associée uw1

mu = ?
w1 = mu*v
uw1 = mu*uv

# Calcul de R et Phi par régression linéaire, et des incertitudes uR et
uPhi

-> partie à compléter

print("résistance interne R=",R,"incertitude_u(R)=", uR)
print("constante de couplage Phi=",Phi,"incertitude u(Phi)=", uPhi )

# Tracé de omega en fonction de I et de la régression linéaire

plt.figure(1); plt.clf()
plt.errorbar(I,w1,xerr = ul, yerr = uw1,capsize = 3, ecolor = 'black',\
             elinewidth = 1, capthick = 1, fmt='none', label = 'Données', zorder
             =3)
plt.xlabel("courant en A")
plt.ylabel("vitesse angulaire en rad/s")
plt.title("Régression linéaire")
plt.plot(I,a*I+b,linestyle="dotted")

# Bilan de puissance

Pe = ? ; Ps = ? ; Pj = ? ; Pp = ?

plt.figure(2); plt.clf()
plt.title("Bilan de puissance", fontsize = 12)
plt.plot(F_verin,Pe, label = "Pe")
plt.plot(F_verin,Pp, label = "Pp")
plt.plot(F_verin,Pj, label = "Pj")
plt.plot(F_verin,Ps, label = "Ps")
plt.xlabel("Force en N", fontsize = 10)
plt.ylabel("Puissance en W", fontsize = 10 )
plt.legend()
```

ANNEXE 2

Transmission des incertitudes pour un produit ou un quotient

Soient a et b deux grandeurs physiques et $u(a)$ et $u(b)$ les incertitudes-type associées.

On définit $p = a \cdot b$ et $q = \frac{a}{b}$.

Les incertitudes sur p et q sont alors données par les expressions :

$$u(p) = p \sqrt{\left(\frac{u(a)}{a}\right)^2 + \left(\frac{u(b)}{b}\right)^2}$$

$$u(q) = q \sqrt{\left(\frac{u(a)}{a}\right)^2 + \left(\frac{u(b)}{b}\right)^2}$$

Sujet E3A 2019 PSI - Modélisation et ingénierie numérique

Tirage variable aléatoire et calcul de fonctions, moyennes....

Fonctions et constantes autorisées - Python	
<code>len(L)</code>	renvoie le nombre d'éléments de la liste L.
<code>L = []</code>	création d'une liste vide.
<code>L = [x]*n</code>	création d'une liste de n éléments ayant tous la valeur contenue dans la variable x.
<code>L = [x for i in range(n)]</code>	même opération que précédemment (alternative avec une compréhension de liste).
<code>L.append(elt)</code>	ajoute l'élément elt à la liste L. La méthode <code>append</code> ne renvoie aucune valeur.
<code>cos(x)</code>	renvoie la valeur du cosinus de x radians.
<code>sin(x)</code>	renvoie la valeur du sinus de x radians.
<code>tan(x)</code>	renvoie la valeur de la tangente de x radians.
<code>sqrt(x)</code>	renvoie la racine carrée de x.
<code>pi</code>	constante dont la valeur est $\pi = 3.141592\dots$ à la précision disponible.
<code>random()</code>	renvoie un nombre pseudo-aléatoire compris entre 0 et 1 avec une densité de probabilité uniforme.

Étude numérique

On se propose d'étudier quelques aspects associés au traitement numérique du signal d'un magnétomètre à vanne de flux.

On utilisera uniquement les fonctions données en annexe. Les bibliothèques correspondantes seront supposées avoir été préalablement importées sous Python. L'usage d'autres fonctions sur les listes telles que `max(liste)`, `min(liste)` ou encore `sum(liste)` est interdit. Ces fonctions devront être programmées explicitement si nécessaire.

G / Quelques aspects algorithmiques associés au traitement des données

Dans l'étude du magnétomètre torique effectuée dans la partie **F**, on est amené à évaluer l'intégrale $\alpha = \iiint_V \frac{\cos^2 \theta}{r^2} dV$ (question **F8**). On cherche à déterminer numériquement la valeur de α .

- G1.** Écrire une fonction `f(x,y,z)` recevant pour paramètres les coordonnées x , y et z d'un point M et qui renvoie la valeur $\frac{\cos^2 \theta}{r^2}$.
On pourra noter que $\frac{\cos^2 \theta}{r^2} = \frac{x^2}{(x^2+y^2)^2}$.
- G2.** Écrire une fonction `Choisir(a,b)` qui renvoie une valeur choisie aléatoirement de manière uniforme entre les bornes a et b passées en argument.

On propose de déterminer la valeur de $\alpha = \iiint_V f(M) dV$ par une méthode numérique

appelée intégration Monte Carlo. L'intégrale s'effectue sur le tore de volume V correspondant au noyau de la bobine ferromagnétique étudiée dans la partie **F**.

L'intégration Monte Carlo est basée sur le fait que l'intégrale de $f(M)$ sur le volume V peut être approximée par $\alpha \simeq V \langle f \rangle$ avec $\langle f \rangle = \frac{1}{N} \sum_{i=1}^N f(M_i)$ où les N points M_i sont choisis aléatoirement et uniformément à l'intérieur du volume V .

L'écart typique entre la valeur exacte de l'intégrale $\iiint_V f(M) dV$ et $V \langle f \rangle$ est donné par $V \sqrt{\frac{\langle f^2 \rangle - \langle f \rangle^2}{N}}$ avec $\langle f^2 \rangle = \frac{1}{N} \sum_{i=1}^N f(M_i)^2$.

Quand le nombre N est suffisamment grand, $V \langle f \rangle$ tend vers $\iiint_V f(M) dV$.

On donne un algorithme permettant de déterminer l'intégrale α effectuée sur le volume V , ainsi que l'erreur relative sur α . On suppose que le nombre de points N est une variable globale préalablement fixée.

- Initialiser les variables I et $I2$ à 0.
- Effectuer les opérations suivantes jusqu'à obtenir N points dans le tore :
 - Choisir un point M au hasard dans le parallélépipède rectangle (de volume minimal) contenant le tore.
 - Si le point M est dans le tore, on détermine $f(M)$. On ajoute cette valeur à I , et la valeur de $f(M)^2$ à la variable $I2$.
- Renvoyer l'estimation de la valeur de α et l'erreur relative associée.

On considérera le parallélépipède rectangle défini par :

$$-(R+a) \leq x \leq R+a, \quad -(R+a) \leq y \leq R+a \quad \text{et} \quad -a \leq z \leq a.$$

On pourra noter qu'un point M de coordonnées cartésiennes (x,y,z) appartient au tore si $(\sqrt{x^2+y^2}-R)^2 + z^2 \leq a^2$.

- G3.** Écrire une fonction `Est_dans_tore(x,y,z)` recevant pour paramètres les coordonnées x , y et z et qui renvoie `True` si le point M est dans le tore et `False` sinon. Les rayons R et a caractérisant le tore seront supposés être des variables globales.
- G4.** Écrire une fonction `Choix_point()` qui renvoie les coordonnées x , y et z d'un point choisi au hasard dans le parallélépipède rectangle de volume minimal contenant le tore.
- G5.** Implémenter le calcul de l'intégrale α et de l'erreur relative associée par la méthode Monte Carlo décrite ci-dessus.

Les différentes opérations décrites dans les parties **D** à **E** du problème peuvent être implémentées de façon numérique. On considère que le signal $u_s(t)$ a été numérisé avec une période d'échantillonnage T_e . Les valeurs numériques associées au signal sont disponibles dans une liste u de taille N . La valeur $u_s(kT_e)$ du signal à l'instant kT_e correspond au k ème élément de la liste u .

L'opération de multiplication apparaissant dans la chaîne de traitement de données (figure 8) peut être implémentée numériquement.

G6. Écrire une fonction `Produit(sx,sy)` qui renvoie le produit des deux listes `sx` et `sy` passées en argument. Les deux listes seront supposées être de même longueur.

On note s la liste obtenue. On a vu dans la partie E que l'information utile correspond à la moyenne de s .

G7. Écrire une fonction `Moyenne(L)` qui renvoie la valeur moyenne des éléments de la liste `L` passée en argument.

La moyenne m de la liste s est liée à l'intensité du champ magnétique à mesurer. Elle dépend aussi du déphasage entre les signaux sx et sy . La valeur de m est maximale quand le déphasage est nul. Il est donc intéressant de chercher à annuler le déphasage entre les signaux sx et sy avant de procéder à l'étape de multiplication.

Pour cela, on commence par rechercher le décalage temporel entre les deux signaux sx et sy de même période. Ce décalage peut être déterminé à l'aide de la fonction d'inter-corrélation $C(n)$ définie numériquement par :

$$C(n) = \sum_{k=n}^{M+n} sx_k \cdot sy_{k-n} \quad (8)$$

où M définit la taille de la fenêtre pour effectuer le calcul. L'idée est de rechercher le retard (ou translation nTe) qui permet de maximiser la ressemblance entre les signaux sx et sy . On cherche alors la plus petite valeur de n pour laquelle $C(n)$ est maximale. La fonction $C(n)$ est elle-même définie pour n compris entre 0 et $N-1-M$ où N est le nombre d'éléments dans la liste sx (ou sy).

On souhaite déterminer la fonction d'inter-corrélation des signaux sx et sy supposés de même longueur $N = 2\ell$ et de moyenne nulle.

G8. Proposer une fonction `Valeur_max(L)` qui renvoie la valeur maximale de la liste `L` passée en argument.

G9. On souhaite que les listes `sx` et `sy` soient de moyenne nulle, ce qui n'est pas a priori le cas. Écrire une fonction `Moyenne_nulle(L)` qui modifie la liste `L` passée en argument de façon à ce que sa moyenne soit nulle.

G10. On choisit de calculer la fonction d'inter-corrélation pour $M = \ell$. Écrire la fonction `Intercorr(sx,sy)` qui renvoie la liste contenant les valeurs de la fonction d'inter-corrélation $C(n)$ pour n compris entre 0 et $\ell - 1$.

G11. Quelle est la longueur de la liste renvoyée par la fonction `Intercorr(sx,sy)` ? Quelle est la complexité temporelle associée à l'exécution de cette fonction dans les conditions précisées ici ?

Sujet MODÈLE travaillé en PCSI – CENTRALE PC 2023

Incertitude de type A et B - Tirage variable aléatoire - Monté Carlo - Calcul de moyennes, écart-type, régression linéaire, schéma Euler

I Une première évolution de la forme du casque : du casque classique vers le casque profilé en goutte d'eau

L'objectif de cette partie est de quantifier le gain de puissance que procure ce type de casque par rapport à un casque dit « classique ».

I.A – Évaluation du gain en puissance par une approche empirique

Afin de quantifier l'apport d'un casque profilé par rapport à un casque traditionnel, on étudie une phase de descente en roue libre (c'est-à-dire sans pédalage) sur route rectiligne, de pente constante. La figure 3 précise le système de coordonnées cartésiennes utilisé.

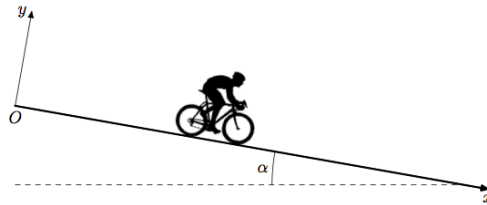


Figure 3 Représentation schématique du système étudié

Pour toute cette partie, on considérera le système {cycliste + vélo}, assimilé à un point matériel M . On supposera ce système soumis aux seules actions extérieures suivantes :

- l'action de la pesanteur, notée $\vec{P}$;
- l'action de la route sur les roues, notée $\vec{R}$ incluant :
 - la réaction normale, notée $\vec{N}$,
 - la résistance au roulement, notée $\vec{F}_r$ et supposée telle que $\vec{F}_r = -\mu_r N \vec{e}_x$ où N est la norme de la réaction normale et μ_r le coefficient de résistance au roulement ;
- l'action de l'air sur le système.

On note m la masse du système, $\vec{v}$ la vitesse du système dans le référentiel terrestre, $\vec{g}$ l'accélération de la pesanteur, α l'angle entre la direction horizontale et la direction de la route.

I.A.1) Étude de la phase de démarrage

Avant d'envisager une étude en régime permanent, il est nécessaire de caractériser la phase de démarrage pour déterminer la distance que parcourt le cycliste avant d'atteindre sa vitesse limite. On supposera que le cycliste est initialement à la position $x = 0$ à l'instant $t = 0$ et se lance dans la pente sans vitesse initiale.

On modélise l'action de l'air sur le système {cycliste + vélo} par une force de trainée de la forme

$$\vec{F}_T = -\frac{1}{2} \rho S C_x v \vec{v}$$

où ρ est la masse volumique de l'air, S la surface frontale projetée sur le plan perpendiculaire à l'écoulement, C_x le coefficient de trainée aérodynamique du système {cycliste + vélo} et v , la norme de $\vec{v}$ sa vitesse par rapport à la route.

Dans cette section I.A.1, on prendra pour les applications numériques $S C_x = 0,30 \text{ m}^2$ et $\alpha = 0,10 \text{ rad}$ (descente sur une pente de 10 %).

Q 1. Rappeler l'expression du nombre de Reynolds, noté R_e , de l'écoulement de l'air autour du cycliste et rappeler ce qu'il caractérise. Donner un ordre de grandeur de R_e pour un cycliste roulant à $40 \text{ km} \cdot \text{h}^{-1}$. En considérant que, si $10^3 < R_e < 10^6$, la trainée est proportionnelle au carré de la vitesse, le modèle de trainée proposé paraît-il adapté ?

Q 2. Déterminer l'équation différentielle vérifiée par v , la norme de la vitesse du système {cycliste + vélo}. On l'écrira sous la forme suivante, en précisant les expressions littérales et les valeurs numériques des paramètres a et b ,

$$\frac{dv(t)}{dt} + av^2(t) = b. \quad (\text{I.1})$$

Q 3. En utilisant l'équation différentielle précédemment établie, déterminer la norme de la vitesse limite atteinte en régime permanent, que l'on écrira sous la forme

$$v_{\text{lim}} = \sqrt{\frac{K}{SC_x}} \quad (\text{I.2})$$

où K est une constante que l'on déterminera en fonction de m , g , ρ , α et μ_r .

Q 4. Calculer la valeur numérique de v_{lim} . Commenter.

La figure 4 propose un programme Python pour résoudre numériquement l'équation (I.1).

```
1 import numpy as np
2 import matplotlib.pyplot as plt
3 def r(f, v, t):
4     V = [v]
5     for i in range(len(t) - 1):
6         V.append(V[i] + (t[i+1] - t[i]) * f(V[i]))
7     return V
8 def g(u):
9     return 0.92 - 2.3e-3 * u * u
10 x = np.linspace(0, 120, 1000)
11 y = r(g, 0, x)
12 plt.plot(x, y)
13 plt.show()
```

Figure 4 Résolution numérique de l'équation (I.1)

Q 5. En explicitant les lignes concernées du programme de la figure 4, expliquer son principe. Quelle méthode de résolution utilise-t-il ?

Le programme de la figure 4 est exécuté avec plusieurs versions de la ligne 10. Les courbes obtenues sont données figure 5.

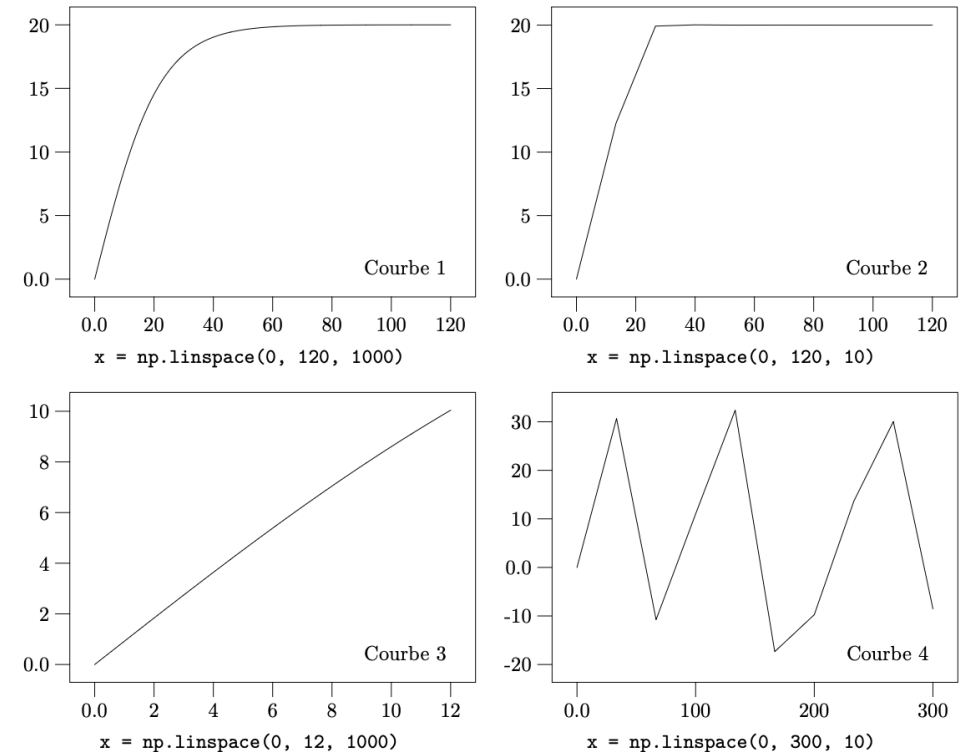


Figure 5

Q 6. Expliciter la signification physique des valeurs figurant dans la ligne 10 et commenter l'allure de ces courbes au regard des valeurs utilisées.

On souhaite déterminer la distance L_{RP} parcourue avant d'atteindre le régime permanent.

Q 7. À partir des courbes proposées sur la figure 5 et en détaillant la démarche suivie, estimer un ordre de grandeur de cette distance.

Q 8. Écrire une fonction Python permettant d'accéder à cette valeur à partir des données numériques issues de ces courbes.

I.A.2) Étude du régime permanent dans la descente de Laffrey

L'objectif est ici de déterminer le gain de puissance que procure un casque profilé par rapport à un casque classique. Pour ce faire, le cycliste est muni d'un capteur GPS permettant de repérer sa position au cours du temps. Différents essais sont réalisés dans la descente de Laffrey en Isère. Cette descente présente l'avantage d'être rectiligne et de pente quasiment constante sur 6,5 km.

On suppose dans cette section que la distance L_{RP} calculée précédemment a été parcourue et que le régime permanent est atteint. Pour cela, on limite l'étude à un sous-segment, de longueur $L' = 5,00$ km et de pente constante 10% en descente (donc d'angle $\alpha = 0,10$ rad), rectiligne. Diverses applications gratuites disponibles sur internet permettent d'accéder au temps mis par le cycliste pour parcourir ce tronçon. On modélise l'action de l'air sur le système par la force de traînée aérodynamique.

Nous avons vu précédemment l'effet du SC_x sur la vitesse du cycliste. Différents paramètres impactent la valeur de ce SC_x . Afin de quantifier l'effet du casque uniquement, nous considérons dans cette partie le même cycliste, muni du même vélo, des mêmes vêtements et dans la même position. Seul le choix du casque diffère. On note alors $SC_{x,c}$ le SC_x du système {cycliste + vélo} lorsque le cycliste est muni d'un casque classique et $SC_{x,p}$ ce SC_x lorsque le cycliste est muni d'un casque profilé. On note respectivement $v_{lim,c}$ et $v_{lim,p}$ les vitesses en régime permanent sur le tronçon considéré.

a) Résultats expérimentaux

Le protocole expérimental mis en œuvre est le suivant : le cycliste parcourt ce tronçon, de longueur L' , 10 fois muni d'un casque de vélo classique, puis 10 fois muni d'un casque de contre-la-montre profilé. Les résultats obtenus sont présentés dans le tableau 1 ci-après.

Numéro de l'essai	Temps de parcours casque classique (en s)	Temps de parcours casque profilé (en s)
1	270	258
2	267	261
3	269	259
4	272	262
5	271	260
6	269	260
7	268	259
8	272	258
9	270	263
10	271	258

Tableau 1 Mesures expérimentales des temps de parcours dans la descente de Laffrey avec casques classique et profilé

Deux sources d'incertitudes sont ici présentes :

- celle de précision de la mesure temporelle, résultant du défaut de précision de la localisation par le GPS. Cet aspect est négligé ici et fera l'objet de la sous-partie I.B ;
- celle de répétabilité du protocole, que nous allons quantifier ici.

Q 9. Déterminer la vitesse moyenne $\langle v_c \rangle$ et l'incertitude-type de type A $u(\langle v_c \rangle)$ sur cette moyenne dans le cas du casque classique.

Le même calcul avec le casque profilé donne $\langle v_p \rangle = 19,25 \text{ m}\cdot\text{s}^{-1}$, $u(\langle v_p \rangle) = 0,041 \text{ m}\cdot\text{s}^{-1}$.

Afin d'estimer l'impact de cette variabilité de mesures sur la faisabilité de l'expérience, on utilise le programme présenté en figure 6 qui simule un grand nombre de valeurs aléatoires répondant aux paramètres calculés à la question précédente. Les résultats obtenus sont présentés sous forme d'histogrammes dans la figure 7.

Q 10. Conclure quant à la pertinence de cette démarche expérimentale.

```
1 import matplotlib.pyplot as plt
2 import numpy.random as rd

3 def MonteCarlo(
4     vitesseMoyenne:float, # vitesse moyenne du cycliste (m/s)
5     incertitudeType:float, # incertitude type sur la vitesse (m/s)
6     N:int, # nombre d'échantillons à produire
7     ) -> [float]: # tableau de N vitesses simulées
8     generateur = rd.default_rng()
9     return generateur.normal(vitesseMoyenne, incertitudeType, N)

10 classique = MonteCarlo(██████, ██████, 10000);
11 profilé = MonteCarlo(19.25, 0.13, 10000);
12 plt.hist([classique, profilé], bins=60)
13 plt.show()
```

Figure 6 Programme de simulation (les valeurs utilisées ligne 10 ont été volontairement masquées)

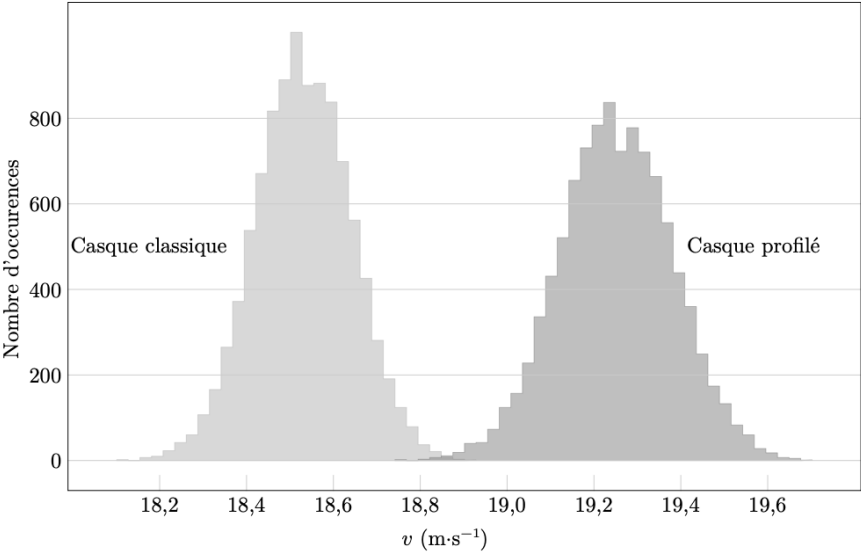


Figure 7 Histogrammes de répartition des vitesses simulées

Q 11. Écrire une fonction Python `MonteCarlo2(vitesseMoyenne, incertitudeType, N, K)`, où K est la constante intervenant dans l'expression (I.2) de la question 3. Cette fonction est une adaptation de la fonction `MonteCarlo` de la figure 6 qui renvoie un couple de valeurs : une évaluation du SC_x et une incertitude-type associée.

Une application de cette fonction conduit aux valeurs du tableau 2.

	Configuration avec casque classique	Configuration avec casque profilé
SC_x (m²)	0,3051	0,2937
incertitude-type (m²)	0,0036	0,0037

Tableau 2 Valeurs de SC_x et incertitude-type associées

b) Exploitation des résultats de l'expérience

Q 12. Exprimer les différentes puissances s'appliquant sur le système {cycliste + vélo} en fonction de m , v , g , ρ , α , μ_r et SC_x .

Q 13. En utilisant le tableau 2, évaluer numériquement le gain en puissance procuré par le casque profilé à $v_{lim} = 54,5 \text{ km}\cdot\text{h}^{-1}$, vitesse moyenne de Greg Lemond lors du contre-la-montre final du Tour 1989. Déterminer également une incertitude-type sur ce résultat. Commenter.

Évaluation des incertitudes

Évaluation de type A :

Si l'on dispose de N mesures y_i d'une grandeur y , alors

Valeur moyenne	Incertitude-type sur une mesure	Incertitude-type sur la moyenne
$\langle y \rangle = \frac{1}{N} \sum_{i=1}^N y_i$	$u(y) = \sigma_y = \sqrt{\frac{1}{N-1} \sum_{i=1}^N (y_i - \langle y \rangle)^2}$	$u(\langle y \rangle) = \frac{u(y)}{\sqrt{N}} = \frac{1}{\sqrt{N}} \sqrt{\frac{1}{N-1} \sum_{i=1}^N (y_i - \langle y \rangle)^2}$

Évaluation de type B :

Si la grandeur mesurée y suit une loi uniforme sur l'intervalle $[a, b]$, l'incertitude-type est alors

$$u(y) = \frac{b-a}{2\sqrt{3}}.$$

Opérations et fonctions Python

On suppose avoir exécuté l'instruction

```
import numpy as np
```

Fonctions

- `range(n:int)` renvoie la séquence des n premiers entiers ($0 \rightarrow n-1$)
`list(range(5))` $\rightarrow$ [0, 1, 2, 3, 4].

Opérations sur les listes

- `len(u)` donne le nombre d'éléments de la liste `u` ;
`len([1, 2, 3])` $\rightarrow$ 3, `len([[1, 2], [3, 4]])` $\rightarrow$ 2.
- `u.append(e)` ajoute l'élément `e` à la fin de la liste `u` (identique à `u[len(u):] = [e]`).

Opérations sur les tableaux

- `a.ndim` nombre de dimensions du tableau `a`.
- `a.shape` tuple donnant la taille du tableau `a` pour chacune de ses dimensions.
- `len(a)` taille du tableau `a` dans sa première (ou seule) dimension (identique à `a.shape[0]`).
- `np.linspace(start, stop, num=50)` crée un vecteur contenant `num` valeurs équiréparties dans l'intervalle `[start, stop]` ;
`np.linspace(1, 10, 4)` $\rightarrow$ array([1., 4., 7., 10.]).
- `np.arange(start, stop, step)` crée un vecteur contenant des valeurs régulièrement espacées de `step` dans l'intervalle `[start, stop[` ;
`np.arange(1, 2, 0.2)` $\rightarrow$ array([1., 1.2, 1.4, 1.6, 1.8]).
- `np.mean(a)`, `np.std(a)` calcule respectivement la moyenne et l'écart-type des éléments du tableau `a`.
- `np.sqrt(a)` crée un tableau de même forme que `a` dont les éléments sont les racines carrées des éléments de `a`.

Générateurs pseudo-aléatoires

- `np.random.default_rng()` crée un générateur de nombres pseudo-aléatoires utilisant l'algorithme *permutation congruential generator* (PGC-64).
- `g.normal(loc=0.0, scale=1.0, size=None)` où `g` est un générateur de nombres pseudo-aléatoires, renvoie un vecteur de `size` échantillons suivant une loi normale d'espérance `loc` et d'écart-type `scale`.