Banque PT, France

Session 2024, filière PT

Informatique et modélisation de systèmes physiques

Durée : 4 heures

Ce que traite le sujet

- Partie modélisation, A : étude d'un climatiseur
- Partie modélisation, B : efficacité du refroidissement
- Partie informatique, C : nombre de FFU, répartition et vitesse de rotation
- Partie informatique, D : modèle du couple résistant à partir des essais
- Partie informatique, E : comportement du moteur entraînant le ventilateur

Sujet reproduit sans retouche à partir de la page suivante.

Résumé des parties, chapitres de cours liés et autres années :
prepahelp.com/annales/banque-pt/pt/2024/info

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

Epreuve d'Informatique et Modélisation de Systèmes Physiques

Durée 4 h

Si, au cours de l'épreuve, un candidat repère ce qui lui semble être une erreur d'énoncé, d'une part il le signale au chef de salle, d'autre part il le signale sur sa copie et poursuit sa composition en indiquant les raisons des initiatives qu'il est amené à prendre.

L'usage de calculatrices est interdit.

AVERTISSEMENT

La **présentation**, la lisibilité, l'orthographe, la qualité de la **rédaction, la clarté et la précision** des raisonnements entreront pour une **part importante** dans **l'appréciation des copies**. En particulier, les résultats non justifiés ne seront pas pris en compte. Les candidats sont invités à encadrer les résultats de leurs calculs.

CONSIGNES :

- Composer lisiblement sur les copies avec un stylo à bille à encre foncée : bleue ou noire.

- L'usage de stylo à friction, stylo plume, stylo feutre, liquide de correction et dérouleur de ruban correcteur est interdit.

- Remplir sur chaque copie en MAJUSCULES toutes vos informations d'identification : nom, prénom, numéro inscription, date de naissance, le libellé du concours, le libellé de l'épreuve et la session.

- Une feuille, dont l'entête n'a pas été intégralement renseigné, ne sera pas prise en compte.

- Il est interdit aux candidats de signer leur composition ou d'y mettre un signe quelconque pouvant indiquer sa provenance.

Tournez la page S.V.P

PARTIE MODELISATION

Des applications industrielles, comme la fabrication de batteries, réclament un environnement précisément contrôlé. En particulier, la température doit être finement régulée pour permettre un déroulement fiable des procédés.

Dans cette partie, deux dispositifs qui participent à cette régulation sont modélisés, un climatiseur puis une ailette de refroidissement.

A. Etude d'un climatiseur

1. Transferts thermiques

On envisage la construction d'un climatiseur monotherme, c'est-à-dire une machine thermique cyclique capable de recevoir ou de fournir un travail, et d'absorber de l'énergie thermique d'une source de chaleur, thermostat à la température T.



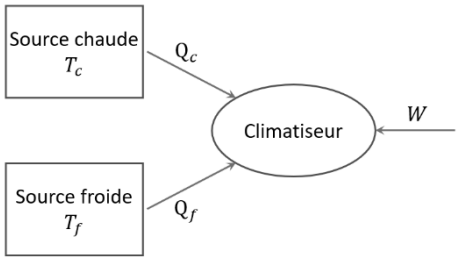
W et Q correspondent aux grandeurs algébriquement reçues sur un cycle

Q1. Exprimer la variation d'énergie interne et d'entropie pour cette machine, en fonction de W, Q, T et de l'entropie $S_{créée}$ créée sur un cycle. Simplifier les expressions en considérant un fonctionnement cyclique.

Q2. Déterminer les signes du travail W et du transfert thermique Q.

Q3. Conclure sur la possibilité de concevoir un tel climatiseur monotherme.

2. Efficacité d'un cycle réversible



On s'intéresse désormais à un climatiseur ditherme, c'est-à-dire à une machine thermique cyclique capable de recevoir ou de fournir un travail, et d'échanger de l'énergie thermique avec deux thermostats aux températures T_f (pour la source « froide ») et T_c (pour la source « chaude »).

Q4. Ecrire deux relations entre le travail W, les transferts thermiques Q_f et Q_c, et les températures T_f et T_c.

Q5. Préciser le signe du transfert thermique Q_f (considérer que la machine est un climatiseur).

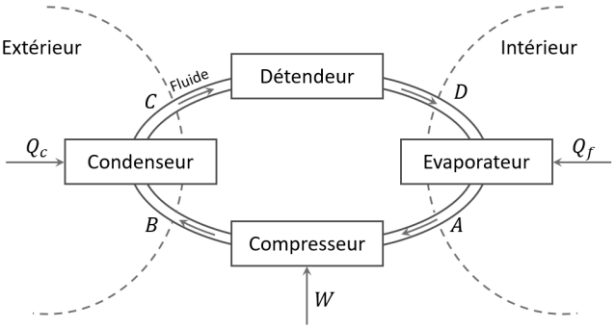
Q6. En déduire le signe du travail W.

Q7. Définir l'efficacité du climatiseur ditherme, et déterminer son expression en fonction des températures T_f et T_c dans l'hypothèse d'un fonctionnement réversible.

Q8. La pièce à climatiser doit conserver une température constante. Commenter l'évolution de l'efficacité dans l'hypothèse d'un fonctionnement réversible, en fonction de la température extérieure.

3. Machine réelle

Le cycle $ABCD$ d'un climatiseur réel peut être représenté par un schéma :



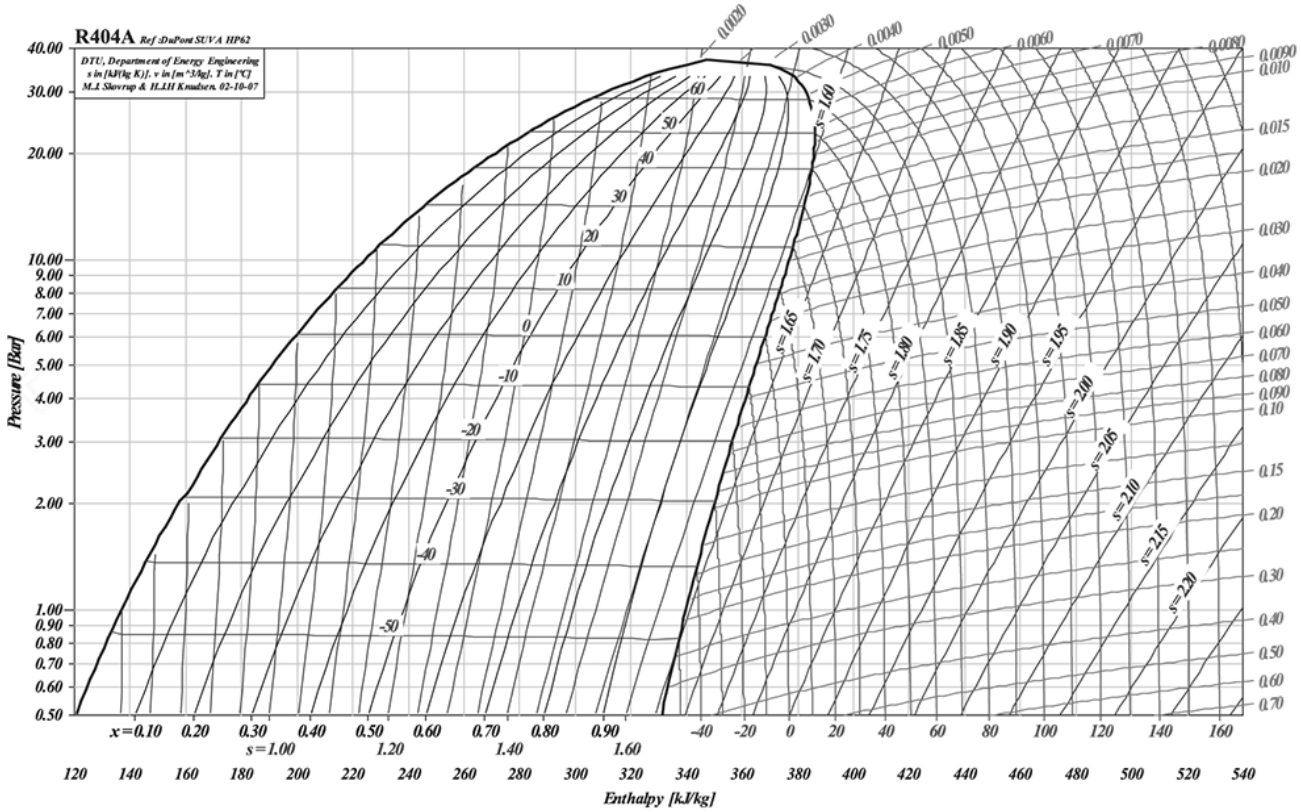
Sur le diagramme enthalpique (P, h) du fluide R404A, sont représentés :

- la courbe de saturation de l'équilibre liquide-vapeur (en trait fort) ;
- les isothermes pour des températures comprises entre $-50\ °C$ et $160\ °$;
- les isentropiques pour des entropies massiques comprises entre $1{,}00\ kJ \cdot K^{-1} \cdot kg^{-1}$ et $2{,}20\ kJ \cdot K^{-1} \cdot kg^{-1}$;
- les isotitres en vapeur sous la courbe de saturation pour des titres massiques en vapeur x variant de 0 à 1.

Dans le domaine du liquide seul, les isothermes sont verticales.

P est exprimée en bar et h en $kJ \cdot kg^{-1}$.

Il est conseillé de placer les points particuliers sur le diagramme ci-dessous pour déterminer les valeurs demandées, mais le diagramme n'est pas à rendre avec la copie.



Tournez la page S.V.P

On étudie l'évolution du fluide au cours d'un cycle en régime permanent.

Q9. A la sortie de l'évaporateur, le fluide est à l'état de vapeur sèche à la température $T_1 = 10\ °C$ et à la pression $P_1 = 4\ bar$. Repérer le point A sur le diagramme. Relever la valeur de l'enthalpie massique h_A du fluide.

Q10. Le compresseur comprime cette vapeur de façon adiabatique et réversible jusqu'à la pression $P_B = 20\ bar$. Repérer le point B sur le diagramme. Relever la valeur de la température T_B et celle de l'enthalpie massique h_B en sortie du compresseur.

Q11. Déterminer la valeur du travail mécanique massique w reçu par le fluide lors de son passage dans le compresseur.

Q12. Le fluide entre ensuite dans le condenseur dans lequel il est refroidi et liquéfié de manière isobare jusqu'à la température $T_C = 43\ °C$.Repérer le point C sur le diagramme. Relever la valeur de l'enthalpie massique h_C en sortie du condenseur.

Q13. A partir de l'état C, le fluide est détendu de manière adiabatique, jusqu'à la pression de l'évaporateur P_1. Le détendeur ne comprend aucune pièce mobile. Montrer que la transformation dans le détendeur est isenthalpique.

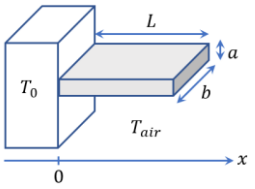
Q14. Repérer le point D sur le diagramme. Relever la valeur de la température T_D et le titre massique en vapeur x_D en sortie du détendeur.

Q15. En déduire le transfert thermique massique q_f échangé par le fluide lors de son passage à travers l'évaporateur (transformation isobare), entre D et A. Commenter son signe.

Q16. Calculer la valeur de l'efficacité e du climatiseur.

Q17. Comparer cette valeur à celle d'un climatiseur réversible fonctionnant entre la température de l'évaporateur et la température de liquéfaction du fluide sous la pression P_2. Commenter le résultat obtenu.

B. Efficacité du refroidissement



On étudie une ailette de longueur L, de section rectangulaire $S = ab$, dont la face en $x = 0$ est en contact avec le dispositif à refroidir, à la température $T_0 = 50\ °C$.

On note $T(x)$ la température dans l'ailette, en la considérant uniforme sur toute section d'abscisse x, et le régime est stationnaire. La température de l'air ambiant, constante et uniforme, est T_{air}.

Le transfert thermique de l'ailette vers l'air ambiant est modélisé, sur la surface latérale de l'ailette, par la relation : $dP = h(T - T_{air})dS$ où dP est la puissance thermique échangée par la surface dS.

h est une constante caractéristique de cet échange thermique, la conductivité thermique de l'ailette est notée λ.

Q18. A l'aide d'un bilan d'énergie réalisé sur une tranche de l'ailette comprise entre les abscisses x et $x + dx$, déterminer l'équation différentielle vérifiée par la température dans l'ailette.

Q19. Introduire une distance caractéristique δ, dont on donnera l'expression en fonction de λ, a, b et h.

Q20. Rappeler la forme générale de la solution de l'équation différentielle, puis préciser à quelle condition portant sur δ il est possible de considérer l'ailette comme infinie.

Q21. En supposant la condition sur δ vérifiée, déterminer complètement l'expression de $T(x)$.

Q22. Exprimer la puissance thermique totale évacuée par l'ailette en fonction de T_0, T_{air}, a, b, λ et h.

Tournez la page S.V.P

PARTIE INFORMATIQUE

Dans cette partie, les fonctions et programmes attendus devront être écrits en langage Python (sauf, bien entendu, lorsqu'une requête SQL est demandée). On prendra garde à la lisibilité du code, et notamment aux indentations. Toute fonction définie dans le sujet pourra être utilisée, même sans avoir été codée dans les questions précédentes.

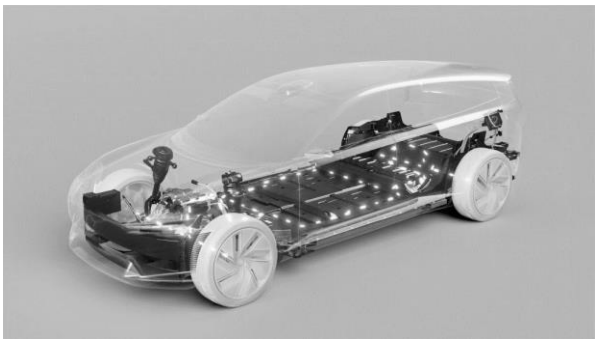
L'objectif de développement de la voiture électrique et des hybrides rechargeables en Europe devait suivre le planning suivant :

- 9 % en 2020 ;
- 38 % en 2025 ;
- 70 % en 2030 ;
- 100 % en 2035.

En effet, l'objectif fixé par le gouvernement consiste à mettre fin à la voiture thermique d'ici à 2035. Toutefois, pour cela, les pays doivent s'équiper de suffisamment de sites de production de batteries pour véhicules électriques afin d'alimenter l'ensemble des constructeurs automobiles européens.

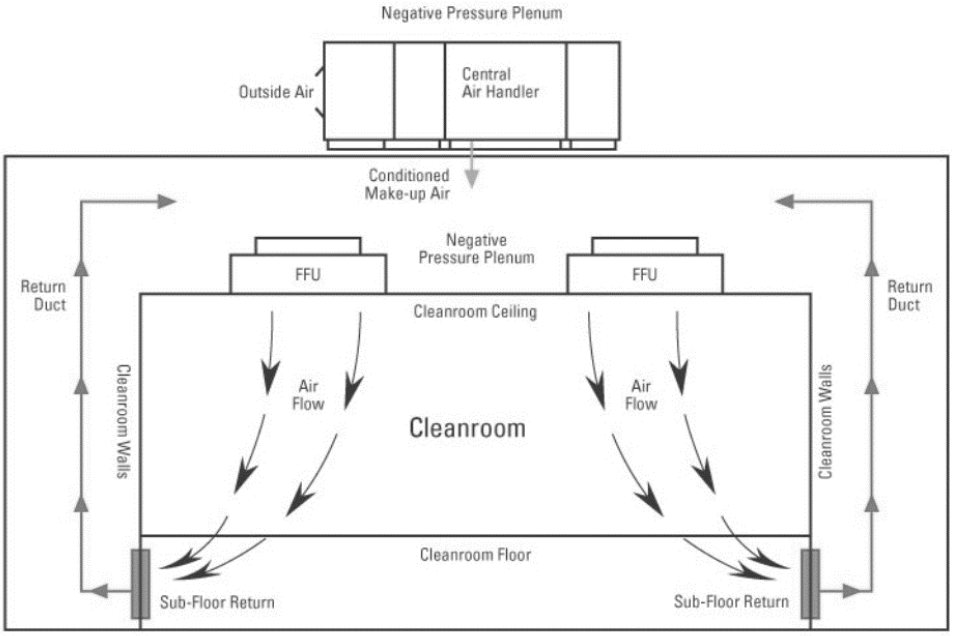






Ainsi la première gigafactory européenne de fabrication de batteries Lithium-ion pour véhicules automobiles voit le jour en Suède et les premières batteries intègreront les véhicules européens dès 2024. Les salles de fabrication d'une telle usine sont des salles dites *salles blanches* avec climatisation et contrôle environnemental précis :

- taux d'empoussièrement classe ISO5
- taux d'humidité réduit : -40° de point de rosée (air très sec comme au pôle Nord)
- température régulée à 20°C +/- 1°C
- pression de +40 Pa (surpression)



Process de contrôle environnemental d'une salle blanche

C. Déterminer le nombre de FFU, leur répartition et ajustement de la vitesse de rotation

Le plafond de la salle blanche est muni de FFU (Fan filter unit) qui sont des ventilateurs de 1 mètre de diamètre. Ils aspirent l'air et le poussent vers le bas. Ils assurent un taux de renouvellement de l'air de 50 volumes / heure.

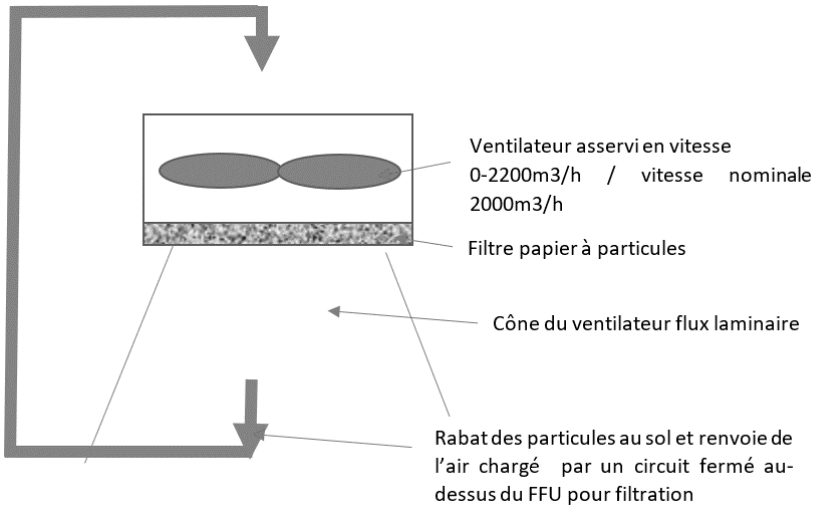


Schéma de principe d'un FFU

L'objectif de cette partie est de déterminer d'une part le nombre de FFU, leur répartition spatiale afin qu'ils soient équirépartis dans la salle et d'autre part la vitesse de rotation proportionnelle au débit à régler sur chacun des FFU afin de respecter une exigence de taux d'empoussièrement imposé par la norme noté tx_renew

Tournez la page S.V.P

On note :
L : longueur de la salle blanche en m ;
ℓ : largeur de la salle blanche en m ;
h : hauteur de la salle blanche en m ;
V : volume de la salle blanche en m^3 ;
Q : débit d'air d'un seul FFU en m^3/h ;
tx_renew : taux de renouvellement d'air indique le nombre de fois à l'heure que l'air total d'une pièce doit être renouvelé ;
nx : nombre de FFU sur une ligne horizontale ;
ny : nombre de FFU sur une ligne verticale ;
n : nombre total de FFU dans la salle.

Q23. Ecrire une fonction **`volume`** qui prend en argument les trois dimensions de la pièce et qui fournit le volume de la pièce en sortie.

Q24. Ecrire une fonction **`nbre_ffu`** qui prend en argument V, Q et tx_renew et qui retourne le nombre n de FFU à prévoir pour répondre à l'exigence du renouvellement d'air du cahier des charges.

On souhaite déterminer le nombre de FFU positionnables le long de l'horizontale noté nx, le long de la verticale noté ny ainsi que l'espacement entre chaque appareil noté u en horizontal et v en vertical afin d'avoir une répartition équirépartie des ventilateurs dans la salle blanche.

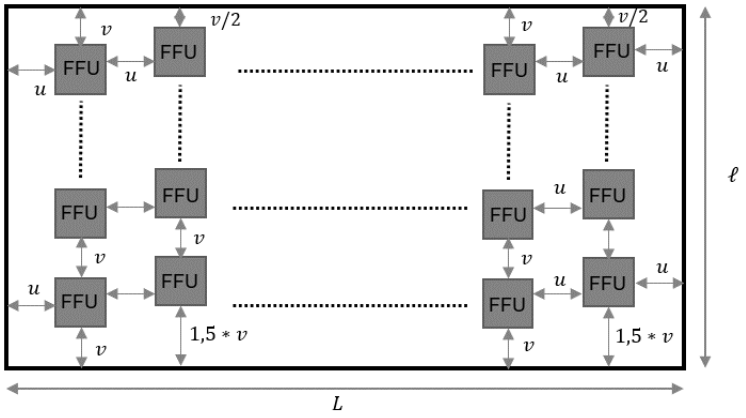


Schéma de positionnement des FFU dans une salle blanche de dimension $L \times \ell$

L'encombrement des FFU est simplifié et modélisé par un carré de 1 m de côté.

L'espacement u ne peut être inférieur à 1 m et l'espacement v ne peut être inférieur à zéro car aucun chevauchement n'est acceptable techniquement.

On souhaite écrire une fonction **`repartition(L,ℓ,n)`** qui prend en argument la longueur L, la largeur ℓ de la salle blanche, le nombre n de FFU calculé précédemment et qui renvoie les valeurs nx, ny, u et v afin d'avoir une répartition équirépartie des ventilateurs dans la salle.

Q25. Justifier que le système d'équations qui relie nx, ny, u et v est le suivant :

$$\begin{cases} L = (nx+1) * u + nx \\ \ell = (ny+1) * v + ny \\ (nx \times ny) < n \ \ avec\ nx\ et\ ny\ des\ entiers\ naturels \end{cases}$$

Q26. Ecrire la situation d'initialisation de la boucle de la fonction **repartition** qui est la situation où tous les FFU sont alignés sur une seule ligne. Déterminer les expressions de nx, ny, u et v dans cette situation initiale.

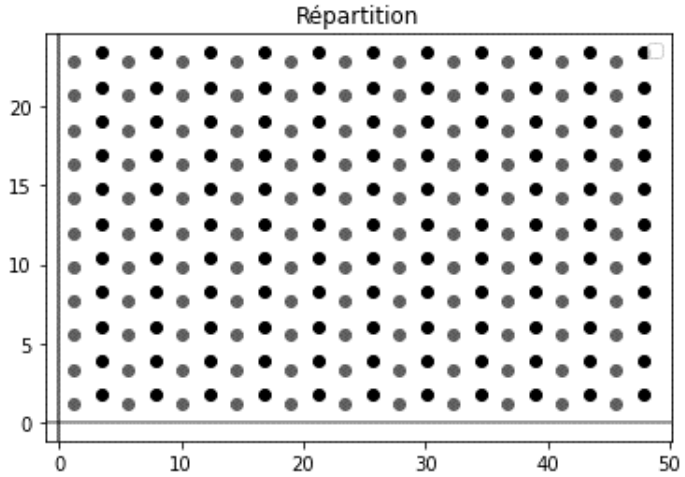
```
def repartition(l,L,n):
    initialisation de nx, ny, u et v

    while COND1 :
        ny=ny+1
        nx=n//ny
        u=EXP2
        v=EXP3
    return nx,ny,a,b
```

script fonction ***repartition(L,ℓ,n)***

Q27. Après l'initialisation des paramètres, la fonction **repartition(L,ℓ,n)** utilise une boucle while pour calculer les nouvelles grandeurs nx, ny, u et v dans la situation d'une répartition des FFU sur une ligne supplémentaire. Une proposition de la boucle while de cette fonction est fournie ci-dessus. Définir la condition d'arrêt de la boucle while COND1 et les expressions EXP2, EXP3 définies dans le script précédent.

La solution obtenue en appliquant la fonction **repartition(L,ℓ,n)** précédente peut être dessinée sous python :



Représentation de la solution de répartition équirépartie des FFU pour une salle blanche de 50x25 m

Comme nx et ny doivent être des entiers naturels, la fonction **repartition(L,ℓ,n)** renvoie une solution telle que $nx * ny < n$.

Pour maintenir le même taux de renouvellement d'air avec moins de ventilateurs, une possibilité est d'augmenter légèrement le débit des ventilateurs par simple augmentation de la vitesse de rotation des ventilateurs sachant que débit et vitesse de rotation sont proportionnels. Le coefficient multiplicateur est calculable par la fonction :

```
def coeff(Q,V,nx,ny,tx_renew):
    Qreel=V*tx_renew/(nx*ny)
    return Qreel/Q
```


Tournez la page S.V.P

D. Etude de résultats expérimentaux afin de modéliser le couple résistant ramené sur l'arbre moteur

L'objectif de cette partie est de dépouiller de manière automatique les fichiers .txt contenant les résultats de 18 essais expérimentaux réalisés sur un FFU afin d'en extraire les données exploitables en vue de construire un modèle de comportement du couple résistant.

1. Extraction des résultats expérimentaux d'un essai

Les fichiers des essais sont tous enregistrés sur le cloud sous le même format .txt. Par exemple les résultats expérimentaux de l'essai 11 nommé "essai11.txt" sont les suivants :

Les 12 premières lignes contiennent les informations relatives à l'essai.

- Ligne 1 : date de l'essai ;
- Ligne 2 : heure de l'essai ;
- Ligne 3 : fréquence d'échantillonnage de la mesure ici 1 000 Hz ;
- Ligne 4 : Système FFU ;
- Ligne 5 : Loi de pilotage : ici la consigne est une loi de vitesse en trapèze avec une phase1 d'accélération – une phase2 à vitesse constante et une phase3 de décélération ;
- Ligne 6 : Nombre de cycles effectués. Ici 1 seul cycle de consigne de vitesse en trapèze ;
- Ligne 7 : durée de la phase d'accélération 30 s ;
- Ligne 8 : consigne de vitesse pour la phase à vitesse constante (différente pour chacun des essais expérimentaux) ;
- Ligne 9 : Durée de la phase à vitesse constante : 50 s ;
- Ligne 10 : : durée de la phase de décélération 30 s ;
- Ligne 11 : indique le nombre de points en lequel est découpé le trapèze de consigne ici 360 points ;
- Ligne 12 : indique l'intervalle de temps entre deux points du trapèze « numérique » précédent ici 0,3 s. Comme la fréquence d'échantillonnage est de 10 Hz = 0,1 s, cela signifie que la consigne est identique pour 3 points de mesure consécutifs.

Les lignes suivantes contiennent les informations relatives aux mesures

- Ligne 13 : titres des grandeurs mesurées séparés par des tabulations : le temps (s) – la consigne (tr/min) – la vitesse du moteur mesurée (tr/min) – le courant moteur mesuré (Ampère) ;
- Ligne 14 et suivantes : les valeurs mesurées à chaque instant. Chaque ligne est une liste de 4 chaines de caractères séparées par des tabulations /t.

Le répertoire courant étant celui qui contient l'ensemble des fichiers résultats, on travaillera par la suite avec les chemins relatifs simples comme "essai11.txt".

Le script suivant permet d'afficher les 26 premières lignes de l'essai 11 :

```
nom_fichier="essai11.txt"
fichier=open(nom_fichier,"r")
for i in range(0,26):
    ligne=fichier.readline()
    print(ligne)
fichier.close()
```

L'instruction **`fichier.readline()`** permet d'obtenir une ligne d'un fichier puis de passer à la ligne suivante.

```
Date Essai  07/02/2023
Heure Essai 14:45
Frequence (Hz)    10
Fonctionnement    ffu
Type Pilotage     Trapeze
Nb cycles   1
Duree acceleration (s)  30
Vitesse (tr/min)  2200
Temps maintien (s)      50
 Duree deceleration (s) 30
Nb points de consigne   365
dt consigne 0,3
Temps (s)   Consigne (tr/min) Vitesse du moteur (tr/min)    Courant (A)
0     22    -0,884      0
0,1   22    -9,96 0,041
0,2   22    -12,924     0,057
0,3   44    -12,924     0,079
4     44    -9,96 0,144
0,5   44    -0,884      0,151
0,6   66    3,744 0,255
0,7   66    3,744 0,349
0,8   66    13,748      0,349
```

Les résultats expérimentaux relevés sont des chaines de caractères type string nombres à virgule alors que Python utilise des nombres à point.

Q28. Écrire une fonction python nommée **`remplace_virgule_par_point(chaine)`** qui prend en argument une chaine de caractères appelée `chaine` et qui renvoie une nouvelle chaine de caractères obtenue en remplaçant les virgules "**,**" par des points "**.**" On rappelle que le type string est un conteneur numéroté comme une liste mais non mutable. Ainsi on fabriquera la nouvelle chaine, caractère par caractère.

Pour pouvoir étudier le comportement du système, il est nécessaire pour chacun des essais d'extraire des données numériques de temps, de consigne, de vitesse et de courant d'un fichier .txt issu d'un essai sur le système FFU, puis de stocker les données extraites de chaque ligne de mesures du fichier dans une liste des temps intitulée **`Ltemps`**, une liste des valeurs de consigne intitulée **`Lconsigne`**, une liste des vitesses moteur intitulée **`Lvitesse`** et une liste des courant intitulée **`Lcourant`**.

On rappelle que :

- L'instruction **`ligne.split("\t")`** renvoie une liste de chaînes de caractères obtenue en divisant la chaîne de caractères **`ligne`** à chaque apparition du séparateur tabulation : **`"\t"`**.

Par exemple :

```
fichier=open("essai11.txt","r")
    for i in range(13):
        ligne=fichier.readline()
    for ligne in fichier:
        ligne2=remplace_virgule_par_point(ligne)
        liste=ligne2.split("\t")
        print(liste)
```

renvoie pour les 6 premières lignes :

```
['0', '22', '-0.884', '0\n']
['0.1', '22', '-9.96', '0.041\n']
['0.2', '22', '-12.924', '0.057\n']
['0.3', '44', '-12.924', '0.079\n']
['0.4', '44', '-9.96', '0.144\n']
['0.5', '44', '-0.884', '0.151\n'] etc…
```

- L'instruction **`fichier.readline()`** permet d'obtenir une ligne d'un fichier puis de passer à la ligne suivante, ce qui va permettre de passer les 13 premières lignes du fichier essai11.txt
- L'instruction **`float("3.14")`** permet d'obtenir le flottant 3.14 correspondant à la chaine de caractères "3.14".Par contre, **`float("3,14")`** renvoie un message d'erreur, car Python ne gère pas les flottants avec des virgules.

```
def extraction_resultats_essai(chemin_fichier):
    #initilisation des listes vides
    Ltemps=[]
    Lconsigne=[]
    Lvitesse=[]
    Lcourant=[]
    fichier=open(chemin_fichier,"r")
    #pour passer les 13 premières lignes
    for i in range(13):
        ligne=fichier.readline()
    #pour traiter les lignes suivantes
    for ligne in fichier:
        ligne2=remplace_virgule_par_point(ARGS1)
        liste=ligne2.split("\t")
        #remplissage des données en transformant les str en float
        Ltemps.append(ARGS2)
        Lconsigne.append(ARGS3)
        Lvitesse.append(ARGS4)
        Lcourant.append(ARGS5)
    fichier.close()
      return Ltemps, Lconsigne, Lvitesse, Lcourant
```

*Script incomplet de la fonction **`extraction_resultats_essai(nom_fichier)`***

Q29. La capture précédente fournit le script incomplet de la fonction **`extraction_resultats_essai(nom_fichier)`** qui prend en argument le nom du fichier de l'essai analysé et qui renvoie les quatre listes des grandeurs physiques numériques type flottants de l'essai étudié **`Ltemps`**, **`Lconsigne`**, **`Lvitesse`**, **`Lcourant`**, soit les données présentes dans la première, la deuxième, la troisième et la quatrième colonne du fichier à partir de la 14e ligne. Définir les arguments ARGS1, ARGS2, ARGS3 et ARGS4 définis dans le script précédent.

La fonction précédente **`extraction_resultats_essai(nom_fichier)`** est ensuite complétée par les instructions suivantes :

```
dico={"temps":Ltemps,"consigne":Lconsigne,"vitesse":Lvitesse,"courant
":Lcourant}
return dico
```

Q30. Que signifient ces 2 instructions et décrire la structure de données choisie ? Préciser son intérêt.

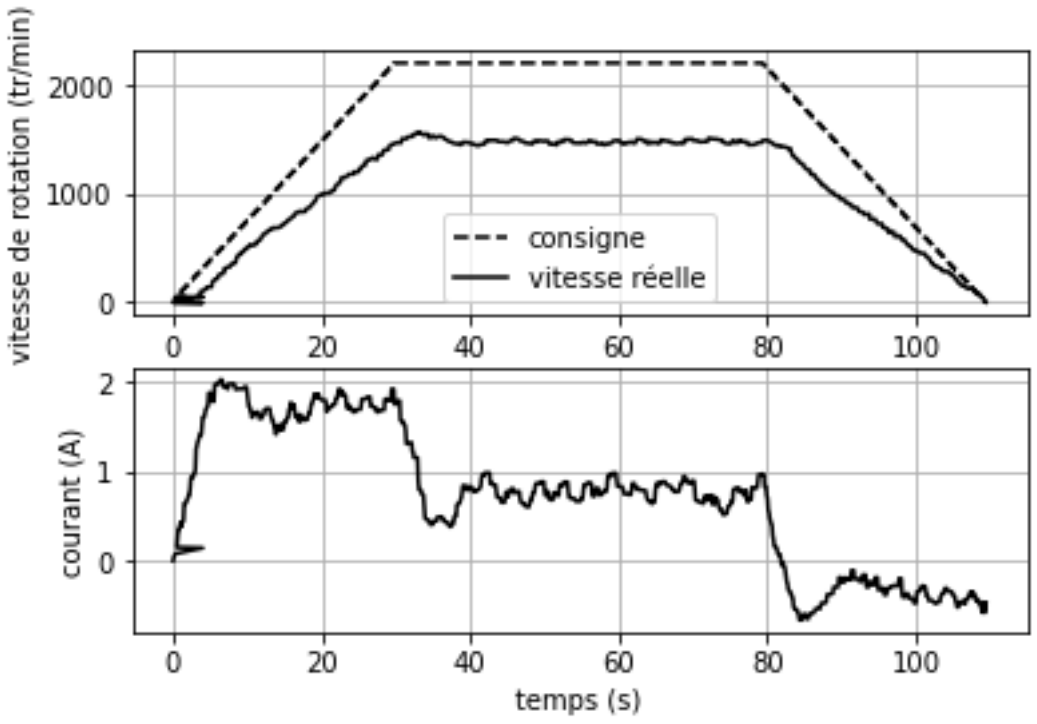
Q31. Ecrire un script python qui permet d'extraire les résultats de l'essai 1 qui seront stockés dans un dictionnaire nommé **`donnees`**, puis qui affiche la liste des temps.

La bibliothèque matplot est intégrée dans le programme :

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

Q32. Ecrire une fonction **`affichage(donnees)`** qui prend en argument le dictionnaire stockant les résultats d'un essai et qui renvoie la courbe des vitesses mesurées d'un essai en fonction du temps.

Les courbes obtenues des résultats expérimentaux pour l'essai 11 sont les suivantes :



Représentation des résultats expérimentaux extrait du fichier « essai11.txt »

Tournez la page S.V.P

2. Détermination du couple résistant

En régime permanent (vitesse de rotation constante, pas d'accélération), le couple moteur $C_m(t)$ sert uniquement à vaincre le couple résistant $C_r(t)$ dans un système FFU où l'axe moteur est fixe par rapport au repère galiléen. De plus pour un moteur à courant continu, le couple moteur est proportionnel au courant $i(t)$ selon la constante de couple K, le couple résistant peut donc se déterminer par la relation suivante (en régime permanent) :

$$C_r(t) = C_m(t) = K \times i(t) \text{ (si la vitesse de rotation est constante)}$$
$$\text{Avec } K = 0{,}13\ Nm/A \text{ (Newton mètre par ampère)}$$

Q33. Écrire la fonction **`moyenne_sur_intervalle(Liste,deb,fin)`** qui permet de déterminer la valeur moyenne des nombres compris entre les indices deb (inclus) et fin (exclu) de la liste **`Liste`**.

Le régime permanent est atteint pour $t \in [50\,;70]$ en secondes correspondant aux indices $i \in [500\,;700]$ avec la fréquence d'échantillonnage de 10 Hz.

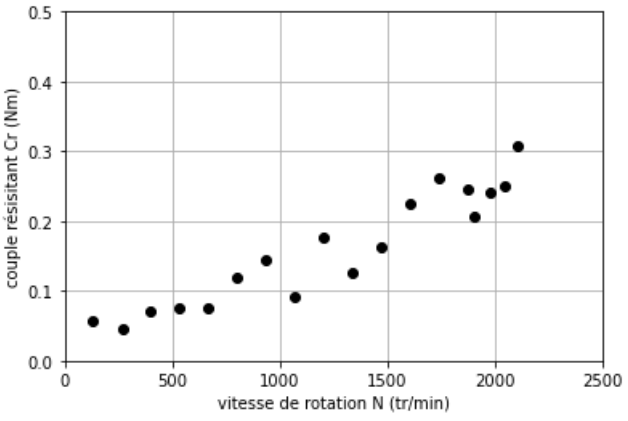
Q34. Écrire une fonction **`point_de_fonctionnement(donnees)`** qui prend en argument le dictionnaire stockant les résultats de l'essai étudié et qui renvoie la valeur de la vitesse de rotation N en régime permanent et du couple résistant Cr de l'essai étudié en tenant compte uniquement du courant entre 50 et 70 s.

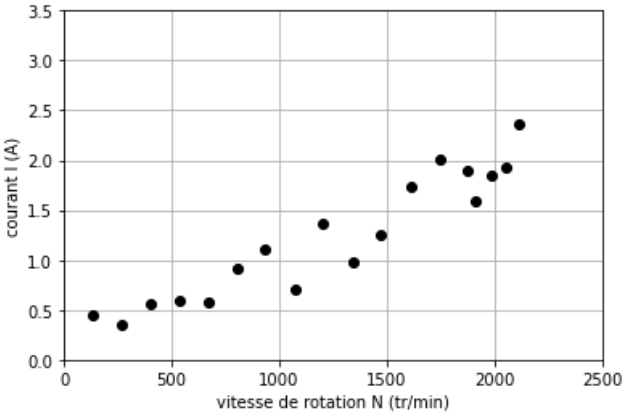
3. Analyse de l'ensemble des essais

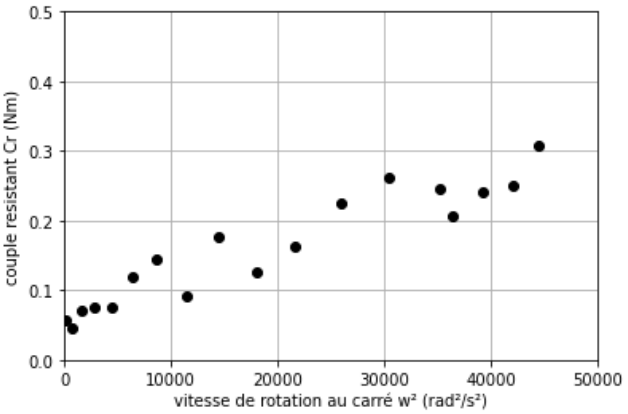
On souhaite maintenant déterminer le point de fonctionnement (N, Cr) de chacun des 18 essais expérimentaux dont les mesures sont stockées dans le même dossier dans 18 fichiers .txt intitulés "essai1.txt" à "essai18.txt". Chacun des points de fonctionnement est mémorisé dans une liste **`L_vit`** et une liste **`L_Cr`**.

Q35. Ecrire un script python qui pour chaque essai présent dans le dossier, détermine le couple résistant Cr et la vitesse de rotation N sur l'intervalle de temps compris entre 50 et 70 s et mémorise Cr et N dans deux listes L_vit et L_Cr.

Les allures des 18 points de fonctionnement expérimentaux du couple résistant (Nm) en fonction de la vitesse de rotation du moteur (tr/min), du courant (A) en fonction de la vitesse de rotation du moteur (tr/min) et du couple résistant (Nm) en fonction du carré de la vitesse de rotation du moteur (rad²/s²) sont représentées sur les trois figures suivantes :







4. Modélisation du comportement couple résistant Cr en fonction de la vitesse du moteur N par une régression linéaire.

L'objectif est de construire un modèle de comportement du couple résistant agissant sur l'arbre moteur dans l'ensemble FFU.

L'analyse des points de fonctionnement expérimentaux nous permet de proposer le modèle de comportement type quadratique :

$$Cr = a_1 * \omega^2 + a_0$$

Il s'agit donc dans la suite de déterminer les constantes a_1 et a_0 par régression linéaire.

La méthode utilisée pour construire la régression est la méthode de descente de gradient utilisée en intelligence artificielle. Celle-ci est utilisée dans notre étude simplifiée pour réaliser une régression linéaire simple mais son champ d'utilisation est bien plus large et permet de réaliser une régression linéaire multiple sur une base de données à plusieurs paramètres. Par exemple on aurait pu ajouter dans le modèle les paramètres : température, humidité et nombre d'heures d'utilisation.

Q36. Justifier que la méthode de descente de gradient est une méthode d'apprentissage supervisé. Expliquer la différence entre un apprentissage supervisé avec classification et avec régression.

Q37. Qu'est-ce qu'une régression linéaire multiple ? Quel serait dans ce cas l'expression du modèle de comportement de Cr à rechercher ?

Tournez la page S.V.P

Description de la base de données résultats de l'étude de la partie précédente:

- la base contient m observations ;

Chaque observation contient :

- une valeur explicative $x^{(k)}$, carré de la vitesse de rotation ω^2 en rad²/s² stockée dans un vecteur V_N2 de dimension $(m \times 1)$;
- une réponse $y^{(k)}$, couple résistant en Nm, stockée dans un vecteur Y de dimension $(m \times 1)$.

$$V_N2 = \begin{bmatrix} x^{(1)} \\ \vdots \\ x^{(m)} \end{bmatrix}, \quad Y = \begin{bmatrix} y^{(1)} \\ \vdots \\ y^{(m)} \end{bmatrix}$$

La bibliothèque numpy est intégrée dans le programme :

```
import numpy as np
```

On rappelle :

- La fonction **np.hstack((M,N))** permet de concaténer deux tableaux pour former une matrice ;
- La fonction **np.ones([i,j])** permet de créer un tableau rempli de 1 de i lignes et j colonnes ;
- La fonction **np.dot(M,N)** renvoie le produit de deux matrices M et N ;
- La fonction **np.transpose(M)** renvoie la matrice transposée de M ;
- La fonction **np.sum(V)** renvoie la somme des composantes du vecteur V.

Q38. Ecrire un script python qui permet de construire la matrice $X = \begin{bmatrix} x^{(1)} & 1 \\ \vdots & \vdots \\ x^{(m)} & 1 \end{bmatrix}$ afin de prendre en compte le biais du modèle affine.

Pour déterminer le modèle, un vecteur F de dimension $(m \times 1)$ est créé : $F = \begin{bmatrix} f(x^{(1)}) \\ \vdots \\ f(x^{(m)}) \end{bmatrix}$

Pour calculer ce vecteur, on utilise la relation : $F = X \cdot \theta$ où θ est le vecteur θ qui contient tous les paramètres du modèle : $\theta = \begin{bmatrix} a_1 \\ a_0 \end{bmatrix}$

Q39. Ecrire la fonction **modele** qui prend en argument (X et $theta$) et retourne la matrice F.

La fonction de coût $Q(a_0, a_1)$ caractérise la somme des erreurs au carré entre le modèle $f\left(x^{(k)}\right)$ et la réponse $y^{(k)}$.

Q40. Ecrire la fonction coût $Q(a_0, a_1)$ en fonction de a_0, a_1, les $y^{(k)}$ et $x^{(k)}$ et m.

Q41. Ecrire la fonction coût $Q(\theta)$ sous forme matricielle en fonction de X, θ et Y.

Q42. Ecrire en python la fonction **cout** qui prend en argument trois vecteurs X,Y et theta et qui renvoie la valeur du coût $Q(\theta)$.

Q43. Ecrire les dérivées partielles de la fonction $Q(a_0, a_1)$: $\frac{\partial Q}{\partial a_1}$ et $\frac{\partial Q}{\partial a_0}$.

Q44. Ecrire le vecteur gradient de la fonction coût $\overrightarrow{grad}(Q)$ sous forme matricielle en fonction de X, θ et Y et montrer que : $\overrightarrow{grad}(Q) = 2 \cdot X^T(X \cdot \theta - Y)$ avec X^T la transposée de la matrice X.

Q45. Ecrire en python une fonction **`gradient`** qui prend en argument trois vecteurs X,Y et theta et qui renvoie le vecteur gradient du cout Q, $\overrightarrow{grad}(Q)$. On pourra utiliser la fonction `np.transpose(M)` qui transpose la matrice M.

La descente de gradient est une méthode itérative qui repose sur le fait que, dans le voisinage d'un point, la fonction de coût décroît moins que son gradient. Et au voisinage du minimum l'écart entre la pente de la fonction coût et le gradient tend vers zéro.

Cet algorithme permet ainsi de trouver le minimum de la fonction de coût en partant de paramètres a_0, a_1 initiaux et aléatoires. A chaque itération les paramètres du modèle sont mis à jour en leur soustrayant la plus grande pente possible multipliée par la vitesse de convergence (le pas) α.

$$a_{1,k+1} = a_{1,k} - \alpha \cdot \frac{\partial Q(a_{1,k})}{\partial a_1}$$

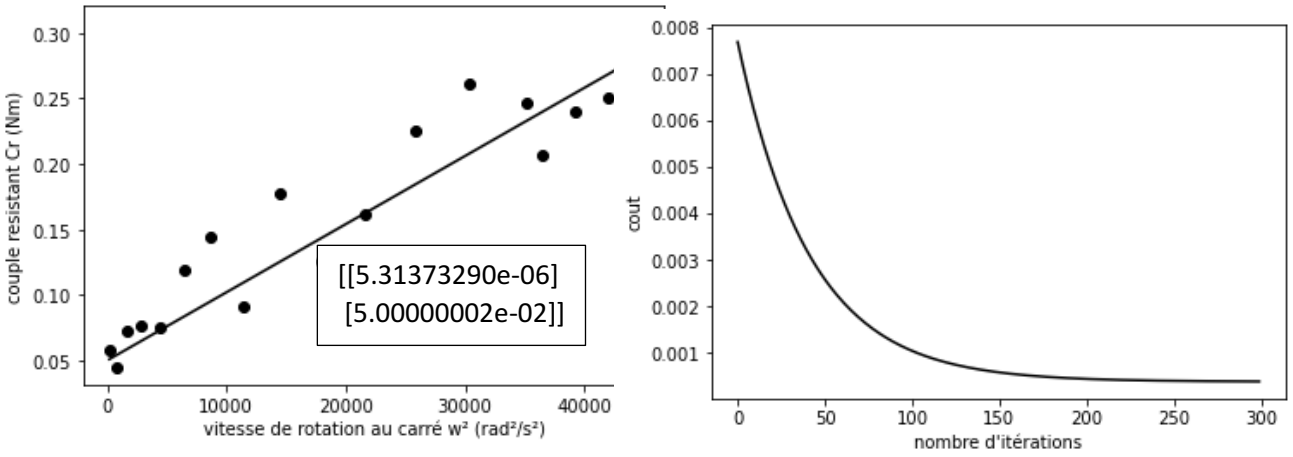
$$a_{0,k+1} = a_{0,k} - \alpha \cdot \frac{\partial Q(a_{0,k})}{\partial a_0}$$

La vitesse de convergence est déterminée par l'hyperparamètre α, appelé vitesse d'apprentissage.

Q46. Ecrire la fonction **`descente_gradient(X,Y,theta_init,alpha,n)`** qui prend en argument les données **X** et **Y**, le vecteur initial noté **`theta_init`**, la vitesse d'apprentissage **`alpha`** choisi, et **n** le nombre d'itérations choisi et qui renvoie le vecteur **`theta`** obtenu par la méthode de descente de gradient après les **n** itérations.

Le programme précédent permet d'obtenir le modèle de couple résistant suivant après 300 itérations:

$$Cr = a_1 * \omega^2 + a_0 \quad avec\ a_0 = 0.05\ et\ a_1 = 5.313e-6$$



Tournez la page S.V.P

E. Etude du comportement du moteur électrique entraînant le ventilateur

Dans la suite de l'étude, l'objectif est d'étudier les caractéristiques du comportement d'un moteur électrique à courant continu à excitation séparée entraînant le ventilateur dont on désire régler la vitesse de rotation.

1. Etude du régime établi : recherche de la tension de commande moteur

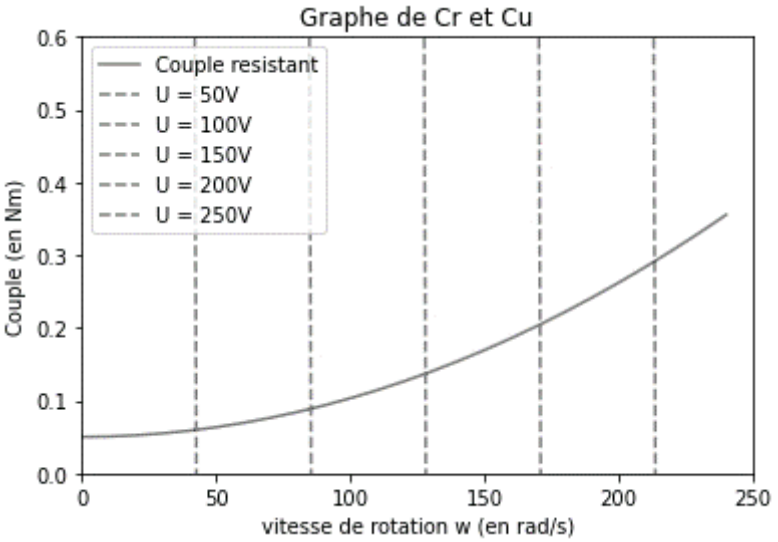
Le ventilateur est entrainé directement par le moteur.

Le couple résistant Cr est lié à la fréquence de rotation ω par la loi : $Cr = a_1 * \omega^2 + a_0$ (avec Cr en Nm, ω en rad/s, a_0=0,05 Nm et a_1 =5,3*10^{-6} Nm/(rad/s)²

Q47. Ecrire une fonction **`affichage`** qui prend en argument les valeurs de a_0 et a_1 et qui permet de tracer l'allure $Cr(\omega)$ pour ω compris entre 0 et 250 rad/s par pas de 10.

Les caractéristiques du moteur du FFU permettent de déterminer l'expression du couple utile Cu (en Nm) sur l'arbre moteur en fonction de ω (en rad/s) et de la tension de commande U (en V) pour I=1,5A=cste dont l'expression est la suivante :

$$C_u = \frac{k}{R}(U - k.\omega) = 0{,}839 \cdot (U - 1{,}17 \cdot \omega)$$



On souhaite déterminer les différentes valeurs de ω pour une tension d'alimentation U donnée. Cela revient à chercher le point d'intersection entre :

$$Cr = a_1 * \omega^2 + a_0$$

$$C_u = 0{,}839 \cdot (U - 1{,}17 \cdot \omega)$$

Et donc à chercher une solution de l'équation :

$$a_1 * \omega^2 + a_0 - 0{,}839 \cdot (U - 1{,}17 \cdot \omega) = 0$$

On définit la fonction f par

```
def f(w,U):
    a1=5.3e-6
    a0=0.05
    return a1*w*w+a0-0.839*(U-1.17*w)
```

Q48. Ecrire une **`fonction dichotomie(U,deb,fin,e)`** qui prend en argument la valeur de la tension d'alimentation imposée U (en V), les bornes de l'intervalle de $\omega\, \epsilon [deb, fin]$, la précision du résultat ε et qui retourne les coordonnées (ω, U) solution de l'équation $f(\omega, U) = 0$ à ε près.

Ainsi grâce à la partie C et la partie E.1 : on connait la tension à imposer à l'ensemble moteur + ventilateur soit 146 V en régime permanent pour assurer la vitesse de rotation nécessaire de 1250 tr/min et donc le taux de renouvellement d'air imposé par le cahier des charges.

2. Etude en régime dynamique : prévision des caractéristiques de la réponse du FFU à un échelon de courant.

Dans cette partie, le courant I dans l'inducteur reste constant.

On néglige désormais la résistance de l'induit R.

On note J le moment d'inertie de l'ensemble aéraulique ramené à l'arbre moteur.

Le moteur est couplé directement au ventilateur, on lui applique un échelon de courant I. On souhaite déterminer l'évolution de la vitesse de rotation $\omega(t)$ au cours du temps sachant que $\omega(0) = \omega_0$.

L'équation de la dynamique s'écrit : $C_m - C_r = J \cdot \frac{d\omega}{dt} \Rightarrow k \cdot I - a_1 * \omega^2 - a_0 = J \cdot \frac{d\omega}{dt}$

Il est donc nécessaire de résoudre l'équation différentielle non linéaire suivante :

$$J \cdot \frac{d\omega}{dt} + a \cdot \omega^2 + b = k \cdot I$$

On utilisera la méthode d'Euler-explicite.

Q49. Ecrire l'équation différentielle régissant la vitesse de rotation du rotor sous la forme $\frac{d\omega}{dt} = g(\omega(t), t)$. On précise que la fonction g ne dépend pas explicitement du temps t.

Q50. A l'aide d'un développement limité de la variable $\omega(t+T)$ à l'ordre 1, montrer que l'approximation de la dérivée peut s'écrire :

$$\frac{d\omega(t)}{dt} \approx \frac{\omega(t+T) - \omega(t)}{T} \quad avec\ T << t$$

Q51. Déterminer l'expression de $\omega(t+T)$ en fonction de T, g(ω(t), t) et $\omega(t)$

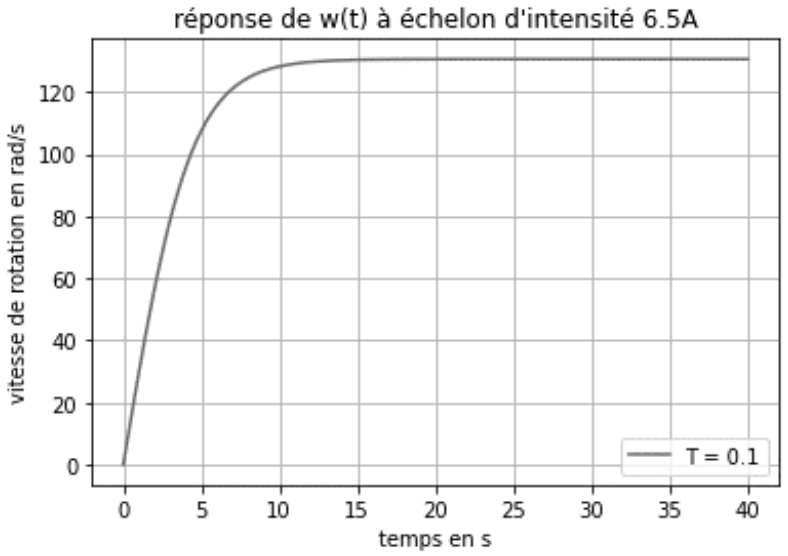
On note $\boldsymbol{LT}$ la liste des temps qui démarre à t0 et s'arrête à t1 et de pas de temps T.

On note $\boldsymbol{L\omega}$ la liste des valeurs de la vitesse de rotation ω .

Tournez la page S.V.P

Q52. Proposer une fonction **`Euler_Explicite`** dont les arguments sont $g, \omega 0, t0, t1, T$ permettant de résoudre l'équation différentielle $\frac{d\omega}{dt} = g(\omega(t), t)$ avec comme condition initiale $\omega(0) = \omega 0$, par itérations successives de pas de temps T sur l'intervalle [$\boldsymbol{t0}$,$\boldsymbol{t1}$] en utilisant la méthode d'Euler explicite, et renvoyant la liste des temps $\boldsymbol{LT}$ et la liste $\boldsymbol{L\omega}$ des solutions approchées associées. On impose d'utiliser une boucle while.

Le tracé de la solution numérique est représenté ci-dessous :



Q53. Les listes $\boldsymbol{LT}$ et $\boldsymbol{L\omega}$ étant calculées, écrire un script python permettant de déterminer puis d'afficher le temps de réponse et la vitesse de rotation en régime permanent du système.

Il sera alors par la suite possible d'analyser les écarts entre les performances simulées et les performances attendues.

IMPRIMERIE NATIONALE – 24 1014 – D'après documents fournis

Travailler les chapitres de ce sujet

Prepa Help suit le programme de PT 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

- Équations différentielles linéaires du second ordre
- Bases de Python : expressions, types, instructions, fonctions
- Spécifier, tester, prouver : terminaison et correction
- Chaînes de caractères et algorithmes sur les textes
- Dictionnaires et principes du hachage
- Listes et algorithmes de recherche (séquentielle, dichotomique)
- Équations différentielles linéaires du premier ordre
- Modèles de comportement et réponse temporelle : premier ordre, second ordre, identification

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/banque-pt/pt