Détermination des lois de mouvement – 60 minutes

|  |  |
| --- | --- |
| **0bjectifs pédagogiques** | * **B3-01** Vérifier la cohérence du modèle choisi en confrontant les résultats analytiques et/ou numériques aux résultats expérimentaux. * **C1-04** Proposer une démarche permettant d'obtenir une loi entrée-sortie géométrique. * **C2-06** Déterminer les relations entre les grandeurs géométriques ou cinématiques. * **C3-01** Mener une simulation numérique. * **D2-04** Choisir la grandeur physique à mesurer ou justifier son choix. * **D2-05** Choisir les entrées à imposer et les sorties pour identifier un modèle de comportement. * **A4-03** Interpréter et vérifier la cohérence des résultats obtenus expérimentalement, analytiquement ou numériquement. * **A4-04** Rechercher et proposer des causes aux écarts constatés. |

|  |  |
| --- | --- |
| **0bjectif** | **En vue de réaliser le dimensionnement du moteur, il est nécessaire de connaître les lois de position, vitesse et accélération du moteur lorsque la barrière (lisse) suit un trapèze de vitesse.** |

|  |  |
| --- | --- |
| **Résoudre analytiquement** | * On propose le schéma cinématique suivant pour modéliser la barrière.     Justifier brièvement le choix des liaisons.   * Quelle est la grandeur géométrique associée au mouvement de la barrière ? Quelle est la grandeur géométrique associée au motoréducteur ? * Etablir la loi entrée-sortie géométrique du système. |

|  |  |
| --- | --- |
| **Résoudre numériquement** | * En utilisant Python**,** implémenter la fonction calcule\_phi(theta : float)->float qui calcule l’angle phi en fonction de l’angle theta du moteur. * En utilisant Python**,** implémenter la fonction calcule\_theta(phi : float)->float qui calcule l’angle theta en fonction de l’angle phi. * Tracer l’angle du moteur pour phi variant de 0 à 90°. * Commenter l’allure de la courbe. Proposer un modèle linéaire. * Donner l’angle parcouru par le moteur pour que la barrière réalise 90°. |

|  |  |
| --- | --- |
| **Résoudre numériquement** | * Ecrire la fonction calcule\_temps(amax :float, vmax :float, angle :float) -> float,float,float, calculant les temps , et dans le cas où le mouvement est régit par un trapèze de vitesse dont l’accélération maximale est amax, la vitesse maximale accessible est vmax, l’angle à parcourir est angle. * Ecrire une fonction calcule\_profil(amax,vmax,angle) -> np.array, np.array, np.array, np.array retournant :   + les\_t : tableau numpy des temps discrétisés toutes les 0,01 s ;   + les\_x : tableau numpy des positions angulaires (en fonction du temps);   + les\_v : tableau numpy des vitesses angulaires (en fonction du temps);   + les\_a : tableau numpy des accélérations angulaires (en fonction du temps). * Tracer les profils de position, vitesse et accélération du moteur pour que la barrière réalise 90°. |

|  |  |
| --- | --- |
| **Expérimenter et Analyser** | * Réaliser un essai en prenant soin de conserver les conditions expérimentales. * Importer les courbes dans Capytale et les afficher. * Superposer les résultats de votre simulation et la courbe expérimentale. |

|  |  |
| --- | --- |
| **Synthèse** | * **Réaliser une synthèse dans le but d’une préparation orale.**   + Présenter les points clés de la modélisation analytique et de la simulation associée.   + Comparer les résultats de la simulation et les résultats expérimentaux.   + Conclure.   🏳 Pour XENS – CCINP – Centrale :   * Donner l’objectif des activités. * Présenter les points clés de la modélisation. * Présenter les points clés de la résolution utilisant Capytale. * Présenter le protocole expérimental. * Présenter la courbe illustrant les résultats expérimentaux et ceux de la résolution. * Analyser les écarts.   🏳 Pour CCMP :   * Synthétiser les points précédents sur un compte rendu. * Imprimer le graphe o ù les courbes sont superposées. |