Détermination des lois de mouvement – 60 minutes

|  |  |
| --- | --- |
| **0bjectifs** | * **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. |

|  |  |
| --- | --- |
| **0bjectifs** | * Analyser les stratégies de ralliement avec trapèze. * Déterminer les lois de commandes de chacun des axes pour ces stratégies de ralliement. * Comparer la commande calculée avec la commande proposée par le logiciel. * Comparer la commande avec le déplacement réel du bras beta. |

|  |  |
| --- | --- |
| **Résoudre analytiquement** | 1. Lois géométriques  * Réaliser un schéma cinématique minimal paramétré du Bras Beta. * Exprimer le vecteur ( origine du repère et extrémité de l’effecteur) en fonction des paramètres. * Soit le point . Exprimer en fonction de . |

|  |  |
| --- | --- |
| **Résoudre analytiquement et numériquement** | 1. Lois de mouvement  * Une image contenant Rectangle    Description générée automatiquementEcrire 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,dt) -> np.array, np.array, np.array, np.array retournant :   + les\_t : tableau numpy des temps discrétisés toutes les dt 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 de l’axe de rotation, pour un angle de 90°. |

Le bras beta propose différentes stratégies de ralliement pour joindre deux pièces :

* échelon de position ;
* rampe de position avec ou sans synchronisation ;
* **trapèze de vitesse avec ou sans synchronisation ;**
* sinus de vitesse avec ou sans synchronisation.

|  |  |
| --- | --- |
| **Analyser et Résoudre analytiquement** | On s’intéresse à la stratégie de ralliement avec et sans synchronisation.   * Réaliser un déplacement du point \*\* au point \*\* avec puis sans synchronisation. * Expliquer la différence entre les deux stratégies de ralliement. * Proposer un algorithme permettant de calculer les temps , et et les grandeurs cinématique vous paraissant utiles pour les deux stratégies. |

|  |  |
| --- | --- |
| **Analyser les écarts** | * Comparer la commande envoyée par le logiciel et la commande déterminée à l’aide de l’activité précédent. * Comparer le signal de commande et le signal réel. * Conclure. |

|  |  |
| --- | --- |
| **Synthèse** | * **Réaliser une synthèse dans le but d’une préparation orale :**   + Présenter les différences entre les stratégies.   + Comparer les lois de commande pour un mouvement programmé avec les déplacements mesurés.   🏳 Pour XENS – CCINP – Centrale – CCMP :   * Donner l’objectif de l’activités. * Donner un schéma cinématique **en couleur et le paramétrage associé.**   🏳 Pour CCMP :   * Ajouter les courbes et les conclusions au compte-rendu. |