Génération de trajectoires circulaires avec un robot manipulateur

I Objectif

Le but de ce projet est de développer une primitive de mouvement afin de faire réaliser à l'organe terminal d'un robot manipulateur un mouvement imposé.
La trajectoire dans l'espace opérationnel se fait à vitesse imposée et les modèles d'un robot manipulateur permettent de calculer les commandes en position, vitesse et accélération à envoyer au robot pour générer le mouvement désiré.
La tâche est donc définie dans l'espace opérationnel et il est demandé de mettre en évidence le lien entre l'espace opérationnel et l'espace généralisé. Travail à réaliser :

- calculer les différents modèles,
- calculer les trajectoires $\boldsymbol{X}(\mathrm{t})$, $\dot{\boldsymbol{X}}(\mathrm{t})$et $\ddot{\boldsymbol{X}}(\mathrm{t})$ à chaque instants d'échantillonnage,
- calculer les trajectoires $\boldsymbol{q}(\mathrm{t})$, $\dot{\boldsymbol{q}}(\mathrm{t})$, $\ddot{\boldsymbol{q}}(\mathrm{t})$ à chaque instants d'échantillonnage,
- simuler vos trajectoires.

Ce BE est découpé en 4 séances de Tps encadrées (3h, 3h, 2h, 4h).

Evaluation : chaque équipe doit rendre un rapport ainsi que le code à la fin du projet. Une évaluation partielle est faite à chaque séance et complétée ensuite par des tests après la livraison finale.

Temps estimé : 12 h en présence des encadrants (4 séances) et 4h-12h hors encadrement.
Il est fortement conseillé de se répartir le travail au sein du groupe afin d'optimiser le temps de travail et de ne pas déborder du volume horaire indiqué.

II Modélisation géométrique du robot RX160 réduit

Le robot RX160 de Staubli est un robot 6R. La figure 1 vous indique ses axes de rotations ainsi que ses dimensions nécessaires pour établir les modèles (voir détails sur *datasheet-staubli-rx160.pdf*).
Pour la suite du BE, afin de simplifier les calculs, on supposera que les 3 dernières liaisons sont fixes afin de se ramener à un robot 3R, ce $RX160$ réduit, noté $RX160R$, est notre robot de travail.
La configuration du robot est définie dans l'espace généralisé et on ne considère pas de butée mécanique sur ce robot dans un premier temps, i.e., $\boldsymbol{q} \in \mathbb{R}^3$.
La situation de l'O.T. est définie par la position du point $O_6 = O_e$ par rapport au repère de base R_0, (O_0, x_0, y_0, z_0).
On a donc la configuration du robot qui est définie par $\boldsymbol{q} = (q_1, q_2, q_3)^T$ et le vecteur de situation est défini par $\boldsymbol{X} = (x_e, y_e, z_e)^T$.

Travail

Sachant que l'origine du repère R_0 (O_0) et du repère R_3 (O_3) et que le point outil (noté $O_6 = O_e$) sont imposés (voir figure 2), calculer le MGD de ce robot.

- Placer les repères et déterminer les paramètres de DHM associés (figure 2).
- Calculer le MGD.
- Programmer une fonction Python qui calcule votre MGD. Tester votre MGD.

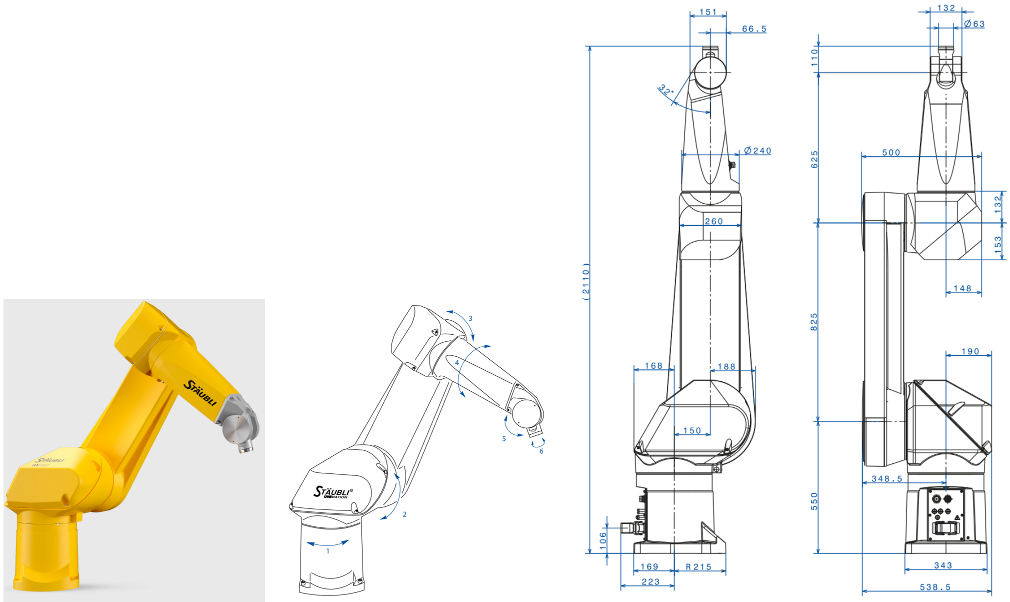

FIGURE 1 – Robot Stauble RX 160

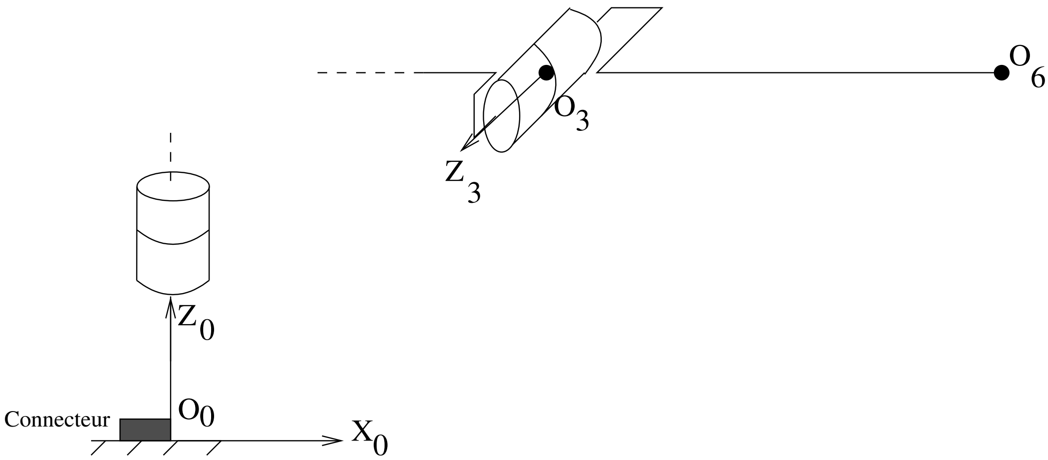

FIGURE 2 – RX 160 réduit

- Calculer le MGI connaissant la position du point $X_e = (x_e, y_e, z_e)$.
- Implémenter votre calcul du MGI.
- Tester vos fonctions et vérifier toutes vos solutions.
- Etablir la conversion entre vos q_i modèle et les qr_i du robot réel que vous simulez.

III Modélisation différentielle

- Calculer la jacobienne (géométrique) de l'UR3R.
- Calculer la jacobienne analytique permettant de connaître $\dot{\boldsymbol{X}}_{\boldsymbol{e}} = (\dot{\boldsymbol{x}}_e, \dot{\boldsymbol{y}}_e, \dot{\boldsymbol{z}}_e)^T$.
- Implémenter en Python votre fonction MDD.
- Tester votre MDD.

IV Génération de mouvement

IV.1 Cahier des charges

On désire graver des cercles sur une plaque verticale avec des vitesses imposées. Le cercle est définit par deux points, A et B qui représente son diamètre. La trajectoire part de A passe en B et termine en A avec le profil de vitesse imposé de la figure 3.
Toutes les accélération/décélération se font à valeur constante K qui est donné par l'utilisateur.
On note $traj(A, B, V_1, V_2)$ cette fonction qui doit retourner les vecteurs $\boldsymbol{q}$(t) , $\dot{\boldsymbol{q}}$(t) (et $\ddot{\boldsymbol{q}}$(t)). .

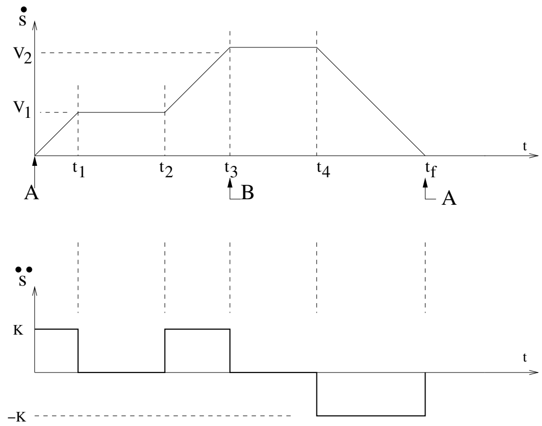

FIGURE 3 – Profils de vitesse et d'accélération de l'abscisse curviligne

Sachant que vous devez envoyer les consignes au robot avec une période Te (de l'ordre de 1 à 5 ms) vous devez calculer vos lois à chaque instants d'échantillonnage.

IV.2 Trajectoire opérationnelle $\boldsymbol{X}$(s) e

- Calculer les coordonnées ($x(s)$, $y(s)$, $z(s)$) des points en fonction de s.
- Calculer les vitesses et les accélérations opérationnelles ($\dot{x}(s)$, $\dot{y}(s)$, $\dot{z}(s)$), ($\ddot{x}(s)$, $\ddot{y}(s)$, $\ddot{z}(s)$).
- Programmer le calcul de la trajectoire opérationnelle.
- Afficher la trajectoire opérationnelle.

IV.3 Loi de mouvement temporelle

Déterminer les lois d'évolution $s(t)$, $\dot{s}(t)$, $\ddot{s}(t)$ de telle sorte que la trajectoire respecte le profil de vitesse imposé (figure 3) La vitesse de l'organe terminal, O_e, initialement nulle en A, atteint le point A avec une vitesse nulle.

- Programmer le calcul de la loi de mouvement $s(t)$, $\dot{s}(t)$, $\ddot{s}(t)$.
- Afficher les courbes correspondantes.

IV.4 Génération de mouvement dans l'espace de la tâche $\boldsymbol{X}$(t)

- Connaissant $s(t)$, $\dot{s}(t)$, $\ddot{s}(t)$ et $x(s)$, $y(s)$, $z(s)$, $\dot{x}(s)$, $\dot{y}(s)$, $\dot{z}(s)$, $\ddot{x}(s)$, $\ddot{y}(s)$, $\ddot{z}(s)$ afficher $\boldsymbol{X}$(t), $\dot{\boldsymbol{X}}$(t), $\ddot{\boldsymbol{X}}$(t).
- Calculer et afficher la vitesse du point O_e.

IV.5 Génération de mouvement dans l'espace articulaire (généralisé)

- A l'aide des modèles inverses, calculer $\boldsymbol{q}$(t) , $\dot{\boldsymbol{q}}$(t) , $\ddot{\boldsymbol{q}}$(t) (entrées de la commande)
- Programmer votre fonction $traj(A, B, V_1, V_2)$ qui prend en entrée les coordonnées des points, les vitesses désirées de l'outil et qui retourne $\boldsymbol{q}$(t) , $\dot{\boldsymbol{q}}$(t) , $\ddot{\boldsymbol{q}}$(t) .
- Tester votre programme dans différents cas pour s'assurer de la robustesse de votre code.

Rappel : Modélisation
Dans le but de simuler une trajectoire dans l'espace généralisé à partir d'une trajectoire dans l'espace opérationnel, il est nécessaire de calculer les différents modèles inverses.

Modèles géométriques	Modèles cinématiques	Modèles d'accélération
$\boldsymbol{X}$(t)= F ($\boldsymbol{q}$ (t))	$\dot{\boldsymbol{X}}= J\ \dot{\boldsymbol{q}}$	$\ddot{\boldsymbol{X}}= J\ \ddot{\boldsymbol{q}}+ \dot{J}\ \dot{\boldsymbol{q}}$
$\boldsymbol{q}$ (t) = F^{-1} ($\boldsymbol{X}$(t))	$\dot{\boldsymbol{q}}= J^{-1}\ \dot{\boldsymbol{X}}$	$\ddot{\boldsymbol{q}}= J^{-1}\ (\ \ddot{\boldsymbol{X}}\text{-}\ \dot{J}\ \dot{\boldsymbol{q}})$

Rappel : Génération de mouvement
De manière générale, la génération de mouvement dans l'espace opérationnel s'effectue en deux étapes : tout d'abord, on définit la trajectoire géométrique désirée pour l'organe terminal du robot, $\boldsymbol{X}$(s) , avec s l'abscisse curviligne le long de la trajectoire. Puis on établit une loi d'évolution temporelle, $s(t)$, sur cette trajectoire, cette dernière permettant de caractériser le mouvement. On obtient $\boldsymbol{X}$(s(t))= $\boldsymbol{X}$(t).

V Simulation

A suivre lors de la séance 2

VI Séance 4

A suivre lors de la séance 3