

Projet de fin de semestre

José de Jesus CASTILLO ZAMORA

IUT de Béziers
Université de Montpellier, LIRMM



In this final project, we will develop a code that allows us to apply the practical knowledge acquired in the field of robotics. The program to be designed will include a graphical interface that reflects this knowledge, enabling the description of the robot's state based on the position of the end-effector and vice versa.

Therefore, it is essential to recall the courses and exercises completed throughout the semester, as they will be key to achieving the proposed objectives.

Mots clés: Codage, Paramètres Denavit-Hartenberg, Cinématique directe ,Cinématique inverse.

Description du projet

Votre tâche consistera à développer un code en langage Python ou Matlab, comme vous l'avez déjà fait dans des cours précédents (par exemple, souvenez-vous du programme que vous avez réalisé le dernier semestre, où vous avez représenté graphiquement le robot dans un espace bidimensionnel).

Pour développer ce code, vous pouvez utiliser des outils d'intelligence artificielle comme ChatGPT. Cependant, leur utilisation sera strictement limitée à la résolution de questions liées à la programmation. Par exemple : "Quelle instruction permet de tracer une ligne dans un espace tridimensionnel ?" ou "Quelle fonction dans les bibliothèques Python permet de réaliser des opérations matricielles, comme l'inversion ou la transposition d'une matrice ?".

Il est essentiel de comprendre que l'utilisation excessive de l'intelligence artificielle pour générer du code sera sanctionnée au point d'invalider complètement le travail. Par conséquent, il est recommandé d'utiliser ces outils avec prudence, ainsi que d'autres ressources disponibles sur Internet, telles que les forums d'aide, les vidéos YouTube, etc.

Pour réaliser cette activité, vous aurez librement accès à toute la documentation officielle du développeur ainsi qu'aux informations pertinentes pour accomplir les tâches de programmation en langage Python ou Matlab.

Éléments essentiels du code

Le programme à livrer devra contenir cinq sections, détaillées comme suit :

1. Description du robot :

Dans cette option, une photo du robot réel sera affichée, indiquant ses composants, y compris la longueur des segments, le type d'articulations, ainsi que les limites des valeurs que chaque variable articulaire peut atteindre. Une liste numérotée devra également identifier les liens et les articulations.

2. Représentation graphique :

Cette section doit afficher une représentation graphique simplifiée du robot. Les liens (liaisons) seront représentés par des lignes épaisses en noir, tandis que les articulations seront représentées par des cylindres pour les articulations rotatives ou des prismes

rectangulaires pour les articulations glissantes. À la base du robot, le système de référence zéro (avec ses axes et son origine) devra être inclus. L'effecteur final sera représenté par une sphère, accompagnée du système de référence correspondant.

3. Paramètres Denavit-Hartenberg :

Les paramètres Denavit-Hartenberg du robot doivent être présentés, accompagnés d'une représentation graphique de chaque système de référence établi selon cette méthodologie. Le robot devra être représenté graphiquement de manière similaire à la section précédente, mais avec l'ajout de tous les systèmes de référence considérés dans l'analyse. Une table des paramètres et leurs valeurs correspondantes devra également être incluse.

4. Cinématique directe :

Cette option doit inclure un menu interactif ou un outil permettant d'introduire les valeurs des variables articulaires. Le programme devra ensuite représenter le robot dans la configuration correspondante et résoudre le problème de cinématique directe. Cela implique de déterminer la position et l'orientation de l'effecteur final (c'est-à-dire du système de référence qui lui est associé) par rapport au système de référence zéro, en fonction d'un ensemble de valeurs pour les variables articulaires.

5. Cinématique inverse :

Cette section devra résoudre le problème de cinématique inverse. Un menu permettra d'entrer la position et l'orientation du système de référence de l'effecteur final (définie par les angles d'Euler) par rapport au système de référence de la base. Une fois ces informations saisies, le programme devra calculer les combinaisons possibles des valeurs des variables articulaires permettant au robot d'atteindre cette position et cette orientation. Le programme devra également représenter graphiquement chaque configuration possible du robot. Par exemple, si la variable articulaire numéro trois peut prendre deux valeurs différentes, les deux configurations devront être représentées. Enfin, le programme devra afficher la matrice Jacobienne à l'écran et déterminer les positions singulières du robot, en indiquant les valeurs des variables articulaires correspondant à ces singularités.

Un exemple développé sous MATLAB version 2023b est fourni pour référence. Ce code est strictement à titre indicatif pour vous donner une idée du projet. Il est important de noter que ce code ne comprend pas la matrice Jacobienne ni la définition des singularités dans la section cinq, car ces éléments constituent un défi à relever par vos propres efforts.

Évaluation

Les critères d'évaluation du projet seront les suivants :

1. Respect des aspects techniques :

Le code doit respecter les cinq points mentionnés et être capable de fournir avec précision les solutions aux problèmes de cinématique directe et inverse. De plus, les paramètres de Denavit-Hartenberg doivent être correctement définis, ainsi que les systèmes de référence obtenus ou établis via cette méthodologie.

2. Créativité et présentation des résultats :

La créativité et la présentation des résultats seront évaluées. Ici, des éléments rendant le programme plus attrayant et convivial pour l'utilisateur seront pris en compte. Vous pouvez inclure une brève description des fonctions du code, une liste d'instructions pour son utilisation correcte, des couleurs variées et des images qui décorent l'interface graphique et améliorent l'interaction de l'utilisateur avec le programme.

3. Propreté et logique du code :

La propreté du code, son efficacité computationnelle et sa logique seront examinées. Il sera également pris en compte si une description adéquate a été ajoutée à chaque ligne du programme.

Affectation des robots par équipe

Vous trouverez ci-dessous un tableau dans lequel est attribué, à chaque équipe, le robot à analyser dans le cadre de ce projet.

Table 1: Affectation des robots par équipe

Robot	Team
ABB IRB 6700	Alain et Sihame
Fanuc M-20iA	Thibault et Julien
Staubli TX2-60	Lucas et Félix
Mitsubishi RV-6SL	Ludovic et Baptiste
Denso Cobotta	Benjamin et Erwan
KUKA KR 6-2	Mateo et Alexandre
Yaskawa Motoman GP8	Rayan et Noémie
Universal Robots UR10	Mehdi et Aurélien