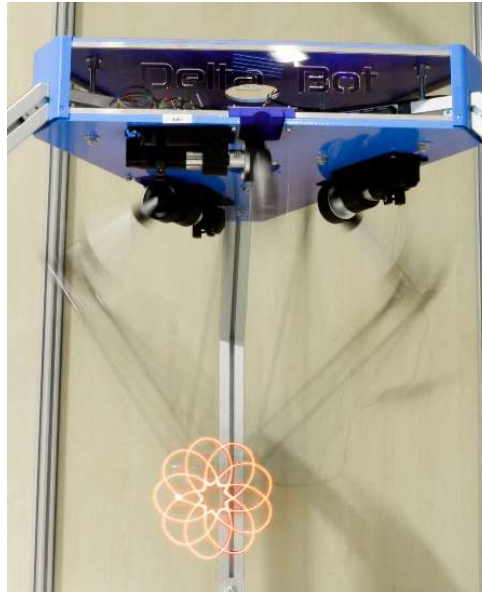


Travaux pratiques

DeltaBot industriel

Thème : Ingénierie numérique – élaboration de trajectoires



Objectif : Le DeltaBot peut être amené, dans certains contextes industriels, à devoir suivre des trajectoires prédéfinies. Cela peut être le cas pour faire de l'impression 3D, pour la réalisation d'une peinture sur une surface de forme gauche, pour une manœuvre automatisée de dégagement d'outil ou d'instrument chirurgical, etc...

Le but de ce TP est de créer ces-dites trajectoires. Celles-ci, imposées dans l'énoncée, prendront la forme de nuages de points créés en langage Python, puis seront simulées en Python, et enfin implantées dans le logiciel de pilotage du DeltaBot afin d'observer la réalité des mouvements que vous aurez programmés.

A/ Qualifications des mouvements d'entrée et sortie

- Lancer le logiciel de pilotage du robot Delta (cf procédure de démarrage).
- Après avoir connecté le robot, et effectué l'initialisation de celui-ci, passer en mode joystick.

Manipulation : Pilotage du robot au joystick

A-1/ Piloter le robot au joystick, afin de vous permettre d'identifier les mouvements d'entrée et le mouvement de sortie.

Combien y a-t-il de mouvements d'entrée ? Qualifier ceux-ci.

Qualifier, en justifiant, le mouvement de sortie.

Le logiciel de pilotage du robot permet d'accéder aux coordonnées cartésiennes du centre de la plate-forme inférieure.

On peut connaître, à chaque instant, les coordonnées (x,y,z) de celle-ci, mais on peut aussi les imposer, grâce au mode de pilotage « CSV ». La façon dont l'asservissement en position est réalisé n'est pas une problématique abordée dans ce TP.

Ce TP s'attache à définir des trajectoires (segments, cercle, hélice, etc...) en écrivant des suites de coordonnées au sein d'un fichier texte. Dans un second temps, il faudra vérifier la validité de ces trajectoires, et enfin les implanter dans le DeltaBot afin qu'il les suive.

Ci-dessous figure un exemple de fichier texte contenant pour chaque ligne :

- Un instant
- Les coordonnées x,y,z de la plate-forme inférieure à cet instant

Une fois téléversé ce fichier texte dans le logiciel, celui-ci le transforme en une suite de positions imposées à la plate-forme inférieure. Le fichier texte permet donc de définir une trajectoire.

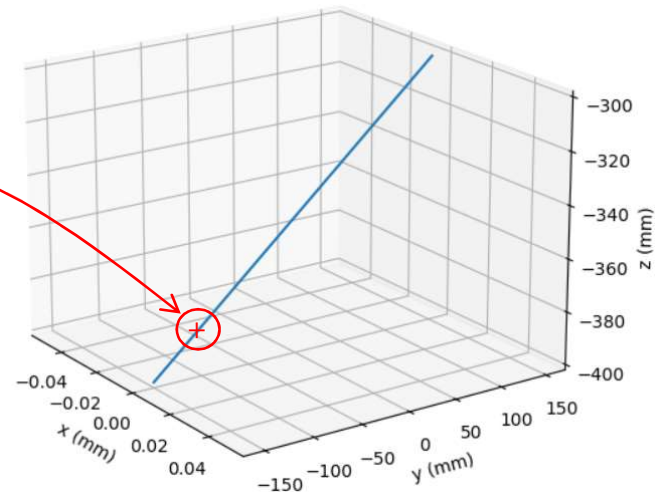
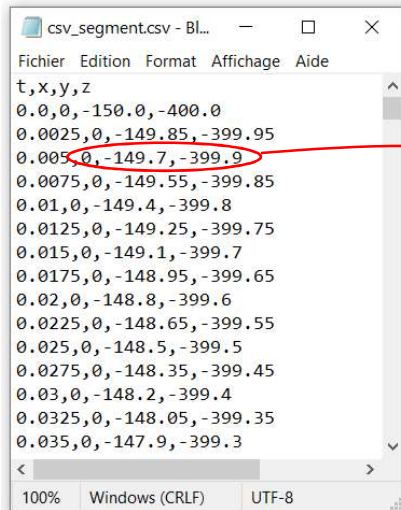


Illustration du mode de définition d'une trajectoire

B/ Trajectoire en coordonnées cartésiennes : SEGMENT

- Création du fichier texte contenant les coordonnées des points successifs :

B-1/ Déterminer les équations paramétriques $x(t)$, $y(t)$, $z(t)$, qui permettent à la plate-forme de relier les points $(0, -150, -400)$ à $(0, 150, -300)$ en 5s, à vitesse constante.

B-2/ Compléter le script `creation_csv_segment_etudiant.py` qui permet de créer le fichier texte à l'image de celui-ci-dessus, et définissant la trajectoire en segment définie précédemment. Exécuter ce script et corriger si nécessaire.

- Affichage de la trajectoire prévisionnelle :

Ouvrir le script `affichage_trajetoire_csv.py`

B-3/ Adapter le script en fonction du nom du fichier texte que vous venez de créer, et compléter les commentaires. Exécuter le script afin de vérifier la validité de la trajectoire définie.

Manipulation : Implantation de la trajectoire sur le DeltaBot

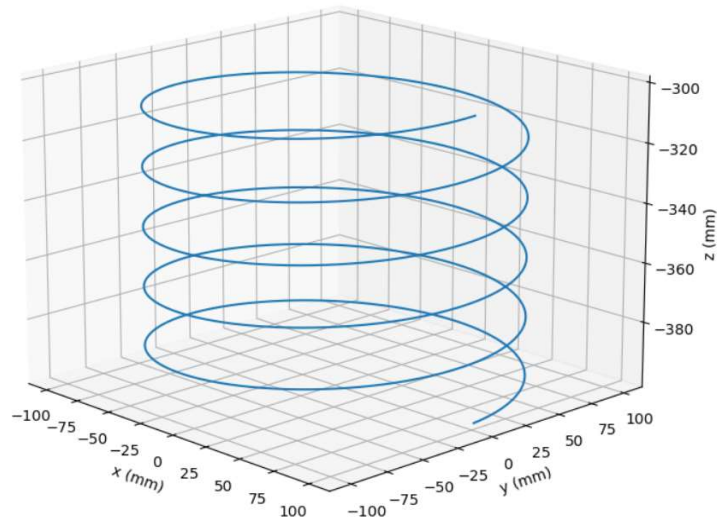
B-4/ Dans le menu « Trajectoire CSV » du logiciel de pilotage du DeltaBot, téléverser le fichier texte que vous avez créé, et lancer son exécution. Valider votre résultat.

C/ Trajectoire en coordonnées cylindriques : HELICE

On souhaite imprimer à la plate-forme inférieure du robot une trajectoire hélicoïdale d'axe vertical, coïncident avec l'axe central du DeltaBot
($x,y)=(0,0)$), à l'image de la figure ci-contre.

Soient :

- R le rayon de l'hélice (exemple : $R=100$ mm)
- pas le pas de l'hélice (exemple : $pas=20$ mm)
- h la hauteur totale de l'hélice (exemple : $h=100$ mm)
- Un point de départ M_0 ($x=R$, $y=0$, $z=-400$)



Trajectoire hélicoïdale

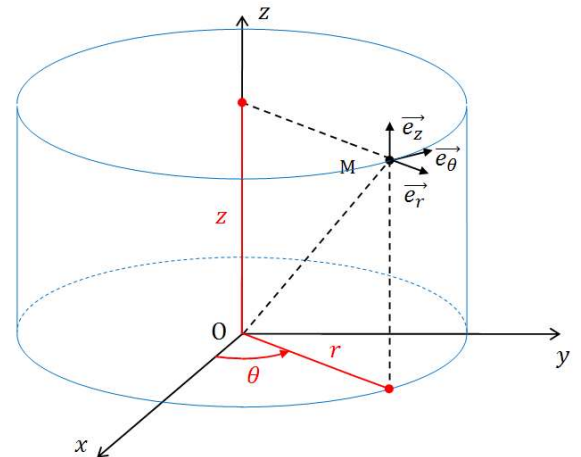
- Création du fichier texte contenant les coordonnées des points successifs :

C-1/ Justifier que le système de coordonnées cylindriques, rappelé ci-contre, soit naturellement adapté à la problématique.

C-2/ Déterminer une relation entre $z(t)$, $\theta(t)$, et p .

C-3/ Si on admet que la plate-forme parcourt l'hélice en 5s à vitesse constante, déterminer les équations paramétriques caractérisant $z(t)$, puis $\theta(t)$, $r(t)$. Compléter le script `creation_csv_helice_etudiant.py` jusqu'à la définition de ces équations paramétriques.

Le logiciel de pilotage du robot n'interprète que les coordonnées cartésiennes d'un point. Il est donc nécessaire d'établir une relation entre les deux systèmes de coordonnées.



Coordonnées cylindriques

C-4/ Exprimer x (puis y) en fonction de r , θ , et z . En déduire les équations paramétriques $x(t)$, $y(t)$, $z(t)$.

C-5/ Compléter le script `creation_csv_helice_etudiant.py` qui permet de créer le fichier texte définissant la trajectoire en hélice définie précédemment. Exécuter ce script et corriger si nécessaire.

- Affichage de la trajectoire prévisionnelle :

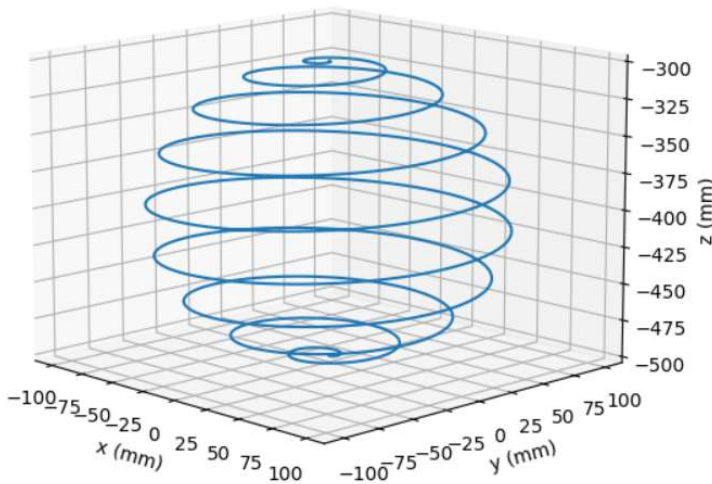
Ouvrir le script `affichage_trajectoire_csv.py`

C-6/ Adapter le script en fonction du nom du fichier texte que vous venez de créer, et compléter les commentaires. Exécuter le script afin de vérifier la validité de la trajectoire définie.

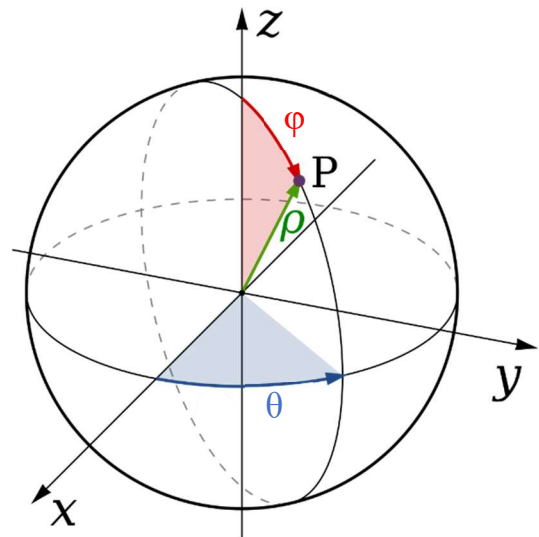
Manipulation : Implantation de la trajectoire sur le DeltaBot

C-7/ Dans le menu « Trajectoire CSV » du logiciel de pilotage du DeltaBot, téléverser le fichier texte que vous avez créé, et lancer son exécution. Valider votre résultat.

D/ Trajectoire en coordonnées cylindriques : HELICE SPHERIQUE



Trajectoire en « hélice sphérique »



Coordonnées sphériques

On souhaite imprimer à la plate-forme inférieure du robot une trajectoire telle que le centre de la plate-forme inférieure se déplace à la surface d'une sphère, en décrivant un mouvement ascensionnel, tel qu'en coordonnées sphériques, on ait :

- La durée totale du mouvement est notée *duree*, et *duree*=9s
- Le zénith $\varphi(t)$ qui évolue à vitesse angulaire constante, en 9 s, de π à 0 rad
- L'azimut $\theta(t)$ qui évolue à vitesse angulaire constante avec une période de 1s
- La sphère a un rayon R de 100 mm
- Le point de départ est M_0 tel que $(x_0=0, y_0=0, z_0=-400)$

- Création du fichier texte contenant les coordonnées des points successifs :

D-1/ Déterminer la période du zénith en fonction de *duree*, puis les expressions de $\varphi(t)$, $\theta(t)$. Compléter le script `creation_csv_sphere.py` en fonction de vos réponses.

Le logiciel de pilotage du robot n'interprète que les coordonnées cartésiennes d'un point. Il est donc nécessaire d'établir une relation entre les deux systèmes de coordonnées.

D-2/ Déterminer les expressions $x(t)$, $y(t)$, $z(t)$. Compléter le script `creation_csv_sphere.py` en fonction de vos réponses.

- Affichage de la trajectoire prévisionnelle :

Ouvrir le script `affichage_trajectoire_csv.py`

D-3/ Adapter le script en fonction du nom du fichier texte que vous venez de créer, et compléter éventuellement les commentaires. Exécuter le script afin de vérifier la validité de la trajectoire définie.

Manipulation : Implantation de la trajectoire sur le DeltaBot

D-4/ Dans le menu « Trajectoire CSV » du logiciel de pilotage du DeltaBot, téléverser le fichier texte que vous avez créé, et lancer son exécution. Valider votre résultat.