

TP 3

Animation d'un robot *Fanuc LR Mate 200iD/4s*

3.1 Objectifs du TP

Vous allez réaliser et implémenter les modèles qui permettent d'animer un robot en position articulaire, en position opérationnelle et en vitesse opérationnelle. Vous allez également utiliser différents outils proposés par *ROS*. Ce TP est découpé en 2 parties.

Première partie (préparation + séance de 4 heures)

- Paramétrage géométrique du robot avec la convention de Denavit-Hartenberg modifiée ;
- Implémentation du modèle géométrique direct pour le contrôle en position articulaire ;
- Implémentation du modèle géométrique inverse pour le contrôle en configuration opérationnelle (position et orientation de son effecteur par rapport au repère de base).

Deuxième partie (préparation + séance de 4 heures)

- Implémentation du modèle cinématique pour le contrôle en vitesse opérationnelle ;
- Contrôle du robot avec les outils de *ROS* ;
- Génération de trajectoire articulaire ;
- Génération de trajectoire opérationnelle :
 - Par modèle géométrique inverse le long d'une trajectoire rectiligne ;
 - Par modèle géométrique inverse le long d'une géodésique.

3.2 Paramétrage géométrique du robot

Représenter les repères des différents corps du robot sur la Figure 3.1 en respectant la convention de Denavit-Hartenberg modifiée.

Convention de Denavit-Hartenberg modifiée

$${}^{i-1}T_i = R(\vec{x}_{i-1}, \alpha_{i-1}) \cdot T(\vec{x}_{i-1}, a_{i-1}) \cdot T(\vec{z}_i, d_i) \cdot R(\vec{z}_i, \theta_i)$$

Les repères sont choisis de la manière suivante :

1. L'axe \vec{z}_i est porté par l'axe de l'articulation reliant le corps $i - 1$ au corps i ;

2. L'axe \vec{x}_i est porté par la perpendiculaire commune aux axes \vec{z}_i et \vec{z}_{i+1} ;
3. L'axe \vec{y}_i est choisi de manière à former un trièdre direct avec les axes \vec{x}_i et \vec{z}_i .

Règles supplémentaires dans le cadre du TP :

1. Les axes \vec{x}_i et/ou \vec{z}_i (en fonction du cas) sont orientés vers le corps suivant (c'est-à-dire qu'il n'y a pas de paramètres a et d négatifs) ;
2. Les axes \vec{z}_i perpendiculaires au plan $(O_0, \vec{x}_0, \vec{z}_0)$ sont orientés comme \vec{y}_0 ;
3. La configuration zéro du robot (c'est-à-dire lorsque toutes les articulations sont à 0°) est celle de la Figure 3.1.

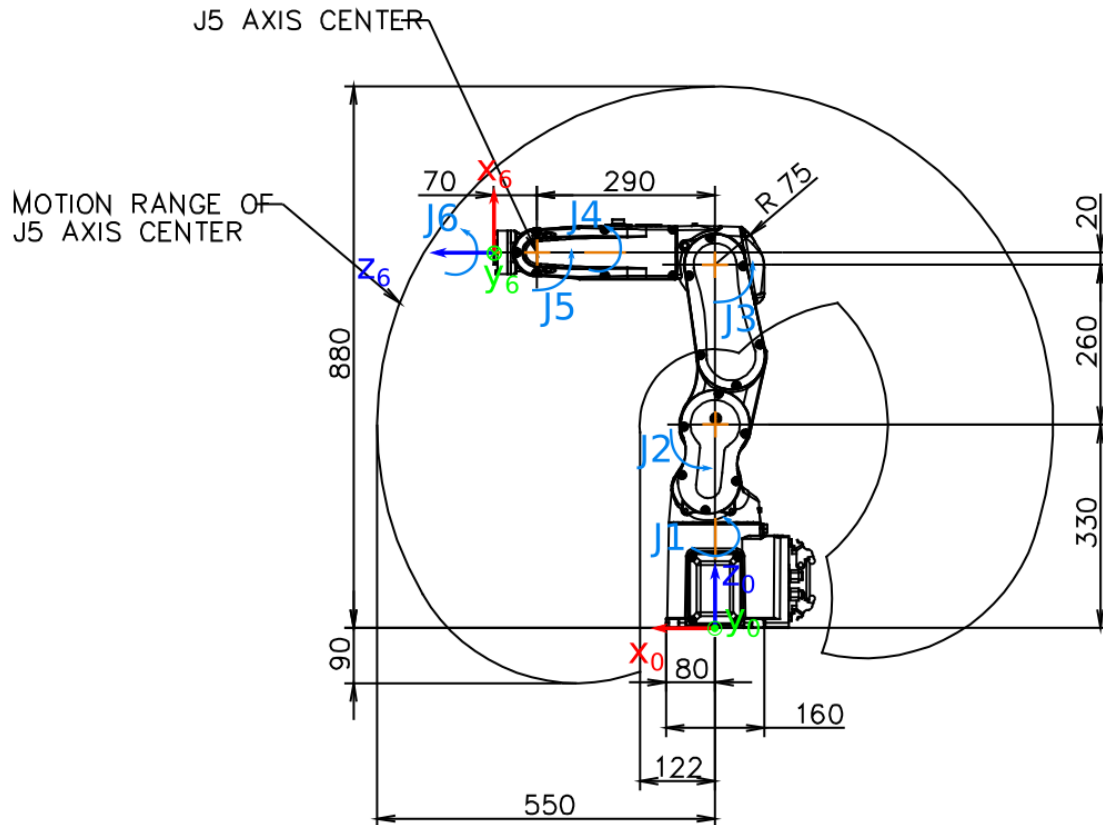


FIGURE 3.1 – Repères des différents corps du robot en suivant la convention de Denavit-Hartenberg modifiée

Compléter le Tableau 3.1 (σ représente le type de liaison : 0 pour rotoïde et 1 pour prismatique).

i	σ_i	a_{i-1} [m]	α_{i-1} [°]	d_i [m]	θ_i [°]
1					
2					
3					
4					
5					
6					

TABLE 3.1 – Tableau des paramètres géométriques en suivant la convention de Denavit-Hartenberg modifiée

3.3 Modèle géométrique direct

Donner la matrice de transformation homogène obtenue à partir des paramètres de Denavit-Hartenberg.

$${}^{i-1}T_i = \left[\begin{array}{ccc|c} & & & \\ \hline & 0 & 0 & 0 \\ & & & 1 \end{array} \right]$$

Quelle est l'expression de la matrice de transformation rigide du corps i au corps j ?

$${}^iT_j =$$

Quelle est l'expression de la matrice de transformation rigide de la base au corps terminal ?

$${}^0T_6 =$$

3.4 Modèle géométrique inverse du *Fanuc LR Mate 200iD/4s*

On cherche à déterminer les configurations articulaires du robot connaissant sa configuration opérationnelle (position et orientation de son effecteur par rapport au repère de base). Il n'y a pas de méthode systématique pour établir un modèle géométrique inverse.

3.4.1 Cas des robots à poignet avec axes concourants

Dans le cas des robots constitués, comme le *Fanuc LR Mate 200iD/4s*, d'un porteur (épaule + coude) puis d'un poignet, les trois premiers axes du robot permettent de positionner le centre du repère lié à l'effecteur tandis que les trois derniers axes permettent de modifier l'orientation de l'effecteur autour de ce point.

On peut donc séparer le problème.

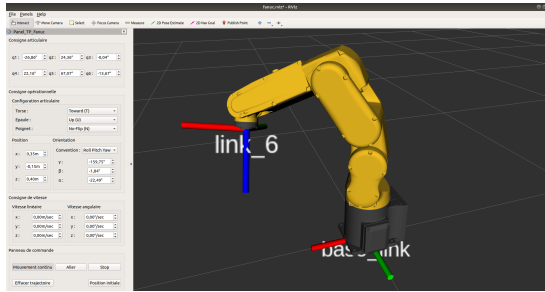
1. On détermine q_1 , q_2 et q_3 qui permettent de positionner l'effecteur ;
2. On détermine q_4 , q_5 et q_6 qui permettent d'orienter l'effecteur.

3.4.2 Configuration articulaire du robot

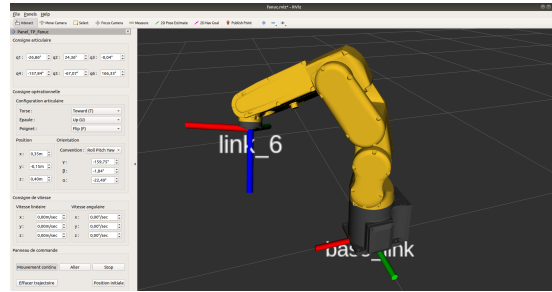
Le robot peut atteindre chaque pose (position + orientation) dans 8 configurations articulaires différentes.

- Le torse peut être vers l'avant (*Backward*) ou vers l'arrière (*Toward*) ;
- Le coude peut être vers le haut (*Up*) ou vers le bas (*Down*) ;
- Le poignet peut être droit (*No-Flip*) ou retourné (*Flip*).

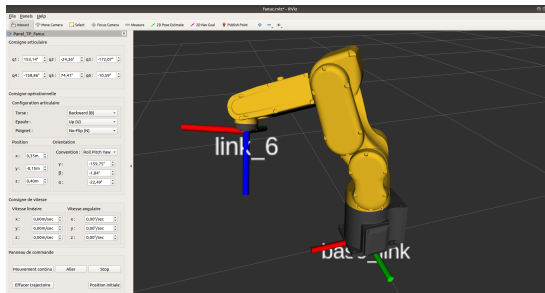
Sur la Figure 3.2, le robot est représenté dans les huit configurations articulaires, pour une pose donnée.



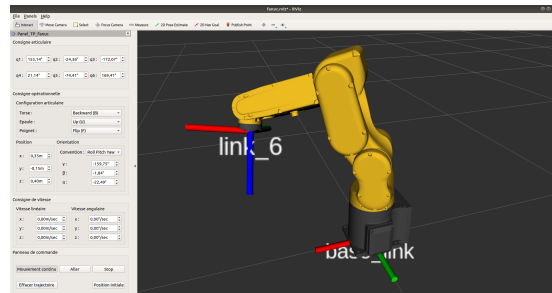
(a) Configuration NUT : No-Flip - Up - Toward



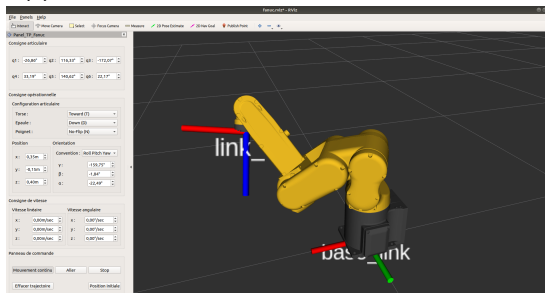
(b) Configuration FUT : Flip - Up - Toward



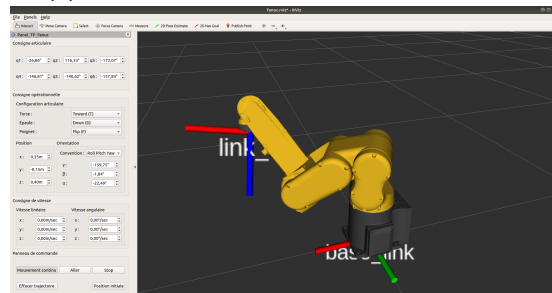
(c) Configuration NUB : No-Flip - Up - Backward



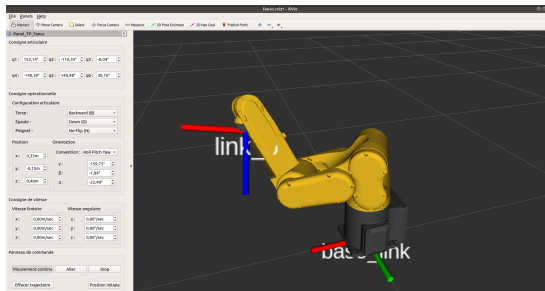
(d) Configuration FUB : Flip - Up - Backward



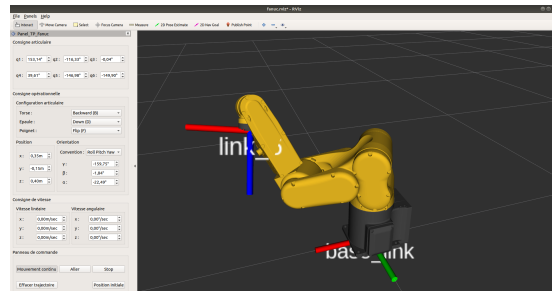
(e) Configuration NDT : No-Flip - Down - Toward



(f) Configuration FDT : Flip - Down - Toward



(g) Configuration NDB : No-Flip - Down - Backward



(h) Configuration FDB : Flip - Down - Backward

FIGURE 3.2 – Représentation des huit configurations articulaires du robot *Fanuc LR Mate 200iD/4s*

3.4.3 Détermination du modèle inverse

Étape 1 : Détermination de la position du centre du poignet

On s'intéresse à la position du centre du poignet O_4 dans R_0 :

On a donc :

$${}^0T_4 = {}^0T_6 \cdot {}^6T_4$$

En développant :

$$\left[\begin{array}{c|c} {}^0R_4 & {}^0O_4 \\ \hline 0 & 0 & 0 & 1 \end{array} \right] = {}^0T_6 \cdot \left[\begin{array}{c|c} {}^6R_4 & {}^6O_4 \\ \hline 0 & 0 & 0 & 1 \end{array} \right]$$

On obtient alors la position de O_4 dans R_0 :

$${}^0O_4 = {}^0T_6 \cdot \left[\begin{array}{c} {}^6O_4 \\ 1 \end{array} \right] = {}^0T_6 \cdot \begin{bmatrix} 0 \\ 0 \\ -d_6 \\ 1 \end{bmatrix}.$$

Étape 2 : Détermination de q_1

On se place dans le plan $(O_0, \vec{x}_0, \vec{y}_0)$, et on projette les points O_1 , O_2 , O_3 et O_4 .

Représenter les segments $\overline{O_1 O_2}$, $\overline{O_2 O_3}$ et $\overline{O_3 O_4}$ et déterminer l'expression littérale de q_1 pour les configurations articulaires base vers l'avant / l'arrière.

Étape 3 : Détermination de q_3

On projette les points O_2 , O_3 et O_4 dans le plan $(O_1, \vec{x}_1, \vec{z}_1)$ (voir Figure 3.4).

1. **Expression de O_4 dans le repère R_1** en fonction des coordonnées de O_4 dans R_0 et de q_1

$$\begin{cases} x_{4/R_1} = \\ y_{4/R_1} = \\ z_{4/R_1} = \end{cases}$$

2. Simplification du modèle

On simplifie le modèle, comme représenté sur la Figure 3.4, pour exprimer x_{4/R_1} et z_{4/R_1} sous la forme suivante :

$$\begin{cases} x_{4/R_1} = a_2 \sin(q_2) + d \sin(q_2 + q_3 + \alpha) \\ z_{4/R_1} = a_2 \cos(q_2) + d \cos(q_2 + q_3 + \alpha) \end{cases} \quad \text{avec :} \quad \begin{cases} \alpha = \\ d = \end{cases}$$

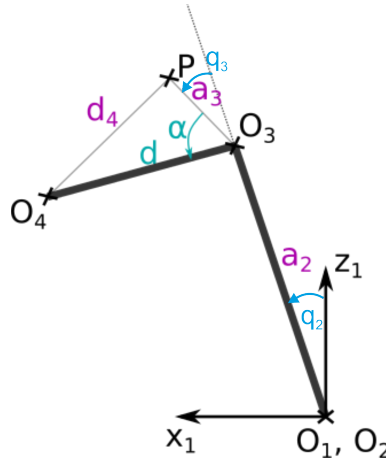


FIGURE 3.3 – Simplification du modèle

3. Détermination de q_3

Faire la somme $x_{4/R_1}^2 + z_{4/R_1}^2$ et simplifier pour arriver à une expression de la forme :

$$x_{4/R_1}^2 + z_{4/R_1}^2 = a + b \cos(q_3 + \alpha)$$

Déterminer l'expression de q_3 pour les différentes configurations articulaires :

Indice : Représenter l'angle $q_3 + \alpha$ comme sur la Figure 3.4 dans les 4 configurations suivantes. Veiller également à représenter les systèmes de coordonnées et plus particulièrement les axes \vec{z}_i .

- Lorsque le torse est vers l'avant (*Toward* - "T") :
 - Coude vers le haut (*Up* - "U") :

 - Coude vers le bas (*Down* - "D") :

- Lorsque le torse est vers l'arrière (*Backward* - "B") :
 - Coude vers le haut (*Up* - "U") :

 - Coude vers le bas (*Down* - "D") :

Étape 4 : Détermination de q_2

On se place dans le triangle formé par les points O_2 , O_3 et O_4 (voir Figure 3.4).

Étape 5 : Détermination de q_4 , q_5 et q_6

Exprimer la matrice de rotation du corps 3 au corps 6 en fonction des paramètres choisis.

$${}^3R_6 = R(\vec{x}_3, \alpha_3) \cdot R(\vec{z}_4, \theta_4) \cdot R(\vec{x}_4, \alpha_4) \cdot R(\vec{z}_5, \theta_5) \cdot R(\vec{x}_5, \alpha_5) \cdot R(\vec{z}_6, \theta_6) \quad \text{avec :} \quad \begin{cases} \alpha_3 = \\ \alpha_4 = \\ \alpha_5 = \end{cases}$$

$${}^3R_6 = \begin{bmatrix} \end{bmatrix}$$

Par ailleurs, on connaît la matrice de rotation de la base au corps terminal 0R_6 (donnée d'entrée du modèle inverse) et on peut calculer la matrice de rotation du corps 0 au corps 3 avec les angles précédemment calculés.

On peut donc calculer la matrice de rotation du corps 3 au corps 6.

$${}^3R_6 = {}^3R_0 \cdot {}^0R_6 = {}^0R_3^T \cdot {}^0R_6$$

Par identification, on peut ainsi déterminer q_5 puis q_4 et q_6 . On notera que $q_4 = \theta_4$, $q_5 = \theta_5$ et $q_6 = \theta_6$.

1. Détermination de q_5

$$\cos \theta_5 =$$

Déterminer l'expression littérale de q_5 pour les configurations articulaires poignet droit / retourné.

2. Détermination de q_4

$$\begin{cases} \cos \theta_4 \sin \theta_5 = \\ \sin \theta_4 \sin \theta_5 = \end{cases} \quad \text{d'où,} \quad \begin{cases} \cos \theta_4 = \\ \sin \theta_4 = \end{cases}$$

Ainsi, lorsque $\sin \theta_5 \neq 0$,

$$q_4 = \theta_4 = \arctan 2(\sin \theta_4, \cos \theta_4) =$$

3. Détermination de q_6

$$\begin{cases} \cos \theta_6 \sin \theta_5 = \\ \sin \theta_6 \sin \theta_5 = \end{cases} \quad \text{d'où,} \quad \begin{cases} \cos \theta_6 = \\ \sin \theta_6 = \end{cases}$$

Ainsi, lorsque $\sin \theta_5 \neq 0$,

$$q_6 = \theta_6 = \arctan 2(\sin \theta_6, \cos \theta_6) =$$

4. Cas de la singularité

Lorsque $q_5 = 0$, une singularité survient. Il n'y a pas de solution unique pour q_4 et q_6 .

On choisit donc arbitrairement $q_4 = 0$.

Ainsi, on obtient les deux relations suivantes :

$$\begin{cases} \cos \theta_6 = \\ \sin \theta_6 = \end{cases}$$

D'où,

$$q_6 = \theta_6 = \arctan 2(\sin \theta_6, \cos \theta_6) =$$

3.5 Modèle cinématique

On cherche à contrôler le robot en vitesse dans l'espace opérationnel. Pour ce faire, on doit déterminer les vitesses articulaires par modèle cinématique inverse.

3.5.1 Définition

$$J(q) = \left. \frac{\partial x}{\partial q} \right|_q$$

Le modèle cinématique direct s'écrit

$$\dot{x} = J(q)\dot{q} = \left[\begin{array}{c} \vec{\omega}_{n/0} \\ \vec{V}(P \in R_n/R_0) \end{array} \right]_{/R_0}$$

et le modèle cinématique inverse

$$\dot{q} = J(q)^{-1}\dot{x}.$$

On peut calculer la jacobienne $J(q)$ de deux manières :

1. En calculant les dérivées partielles, pour obtenir une forme littérale de $J(q)$;
2. Avec les torseurs cinématiques. Ici, nous utilisons cette méthode car le nombre de degrés de mobilité est trop important pour obtenir une forme littérale simple de $J(q)$.

$$\left[\begin{array}{c} \vec{\omega}_{n/0} \\ \vec{V}(P \in R_n/R_0) \end{array} \right]_{/R_0} = \sum_{i=1}^n \left[\begin{array}{c} \vec{\omega}_{i/i-1} \\ \vec{V}(P \in R_i/R_{i-1}) \end{array} \right]_{/R_0}$$

Exprimer alors les torseurs cinématiques pour les deux types de liaison :

- Cas d'une liaison rotoïde ($\sigma = 0$) ;

$$\left[\begin{array}{c} \vec{\omega}_{i/i-1} \\ \vec{V}(P \in R_i/R_{i-1}) \end{array} \right]_{/R_0} = \left[\begin{array}{c} \vec{e}_i \\ \vec{u}_i \end{array} \right] \cdot \dot{q}_i$$

- Cas d'une liaison prismatique ($\sigma = 1$).

$$\left[\begin{array}{c} \vec{\omega}_{i/i-1} \\ \vec{V}(P \in R_i/R_{i-1}) \end{array} \right]_{/R_0} = \left[\begin{array}{c} \vec{e}_i \\ \vec{u}_i \end{array} \right] \cdot \dot{q}_i$$

où :

$$\left| \begin{array}{l} \vec{e}_i : \text{axe porté par l'articulation } i \text{ exprimé dans le repère de base } R_0 \\ \vec{u}_i : \text{vecteur du repère de l'articulation } i \text{ au repère terminal exprimé dans le repère de base } R_0 \end{array} \right.$$

3.5.2 Implémentation

Exprimer la colonne i de la matrice jacobienne, correspondant au déplacement de l'effecteur dû à l'articulation i .

$$J_{-,i}(q) = \left. \frac{\partial x}{\partial q_i} \right|_q =$$

Exprimer \vec{e}_i et \vec{u}_i en fonction des matrices de transformation rigide 0T_n et 0T_i .

3.6 Génération de trajectoires

On souhaite générer différentes trajectoires :

- Génération de trajectoire articulaire ;
- Génération de trajectoire opérationnelle :
 - Par modèle géométrique inverse le long d'une trajectoire rectiligne ;
 - Par modèle géométrique inverse le long d'une géodésique.

3.6.1 Trajectoire articulaire

Compléter l'Algorithme 3.

Remarque : Faire uniquement une interpolation entre les positions. Les notions de génération de trajectoires sont approfondies lors du TP 1.

```

/* Variables :                                          */
/* q : coordonnées articulaires à l'instant t          */
/* q0 : coordonnées articulaires de départ             */
/* qf : coordonnées articulaires cible                 */
/* t : instant t                                       */
/* dT : période d'échantillonnage                     */
/* T : durée de la trajectoire                         */
/* nb_points : nombre de points                       */

```

```

pour          faire
|
|
|
fin

```

Algorithme 3 : Pseudo-code pour la génération de trajectoire articulaire

3.6.2 Représentation des angles d'Euler

Exprimer les matrices de rotation d'un angle α , β et γ autour de l'axe \vec{z} , \vec{y} et \vec{x} respectivement.

$$R_{\vec{z},\alpha} = \begin{bmatrix} & & \\ & & \\ & & \end{bmatrix}, \quad R_{\vec{y},\beta} = \begin{bmatrix} & & \\ & & \\ & & \end{bmatrix}, \quad R_{\vec{x},\gamma} = \begin{bmatrix} & & \\ & & \\ & & \end{bmatrix}$$

Déterminer la matrice de rotation pour les angles d'Euler définis pour la convention ZYX.

$$R(\alpha, \beta, \gamma) = \begin{bmatrix} & & \\ & & \\ & & \end{bmatrix}$$

3.6.3 Représentation axe-angle

Questions

Faire un schéma de la représentation axe-angle.

Cette représentation est utile dans le cadre de la trajectoire géodésique (voir Section 3.6.4). Quel est l'intérêt de cette représentation par rapport à la représentation avec les angles d'Euler ?

Détermination de la matrice de rotation depuis la représentation axe-angle

$$\left(\begin{bmatrix} u_x \\ u_y \\ u_z \end{bmatrix}, \theta \right) \Rightarrow R = \begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{bmatrix}$$

Exprimer la matrice de rotation issue d'une rotation suivant la convention axe-angle.

$$R(u, \theta) = \begin{bmatrix} & & \\ & & \\ & & \end{bmatrix}$$

Détermination de la représentation axe-angle depuis la matrice de rotation

$$R = \begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{bmatrix} \Rightarrow \left(\begin{bmatrix} u_x \\ u_y \\ u_z \end{bmatrix}, \theta \right)$$

- Déterminer l'angle θ

- Déterminer l'axe \vec{u}

3.6.4 Trajectoire opérationnelle

Le long d'une trajectoire rectiligne

On cherche à présent à animer le robot le long d'une trajectoire rectiligne permettant de faire varier l'orientation entre deux configurations cartésiennes (position et orientation du repère terminal exprimées dans le repère de base du robot saisies par l'utilisateur).

Les positions initiales et finales sont exprimées en **[m]** et les orientations initiales et finales sont exprimées avec les angles d'Euler en **[deg]**.

Compléter l'Algorithme 4.

Remarque : Faire uniquement une interpolation entre les positions et les orientations. Les notions de génération de trajectoires sont approfondies lors du TP 1.



Algorithme 4 : Pseudo-code pour la génération de trajectoire opérationnelle le long d'une trajectoire rectiligne

Le long d'une géodésique

Si l'interpolation linéaire entre la position initiale et la position finale permet d'obtenir la trajectoire la plus courte entre les positions (trajectoire rectiligne) il n'en est pas de même pour les rotations lorsque l'interpolation se fait sur les angles d'Euler.

Pour obtenir le trajet le plus court en rotation (trajectoire géodésique), on décrit la rotation entre la pose initiale et la pose finale par la représentation axe-angle puis on fait une interpolation sur l'angle.

Compléter l'Algorithme 5.

Remarque : Faire uniquement une interpolation entre les positions et les orientations. Les notions de génération de trajectoires sont approfondies lors du TP 1.

```
pour          faire
|
|
|
fin
```

Algorithme 5 : Pseudo-code pour la génération de trajectoire opérationnelle le long d'une trajectoire géodésique