

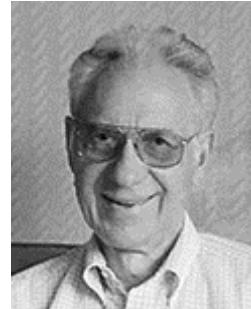
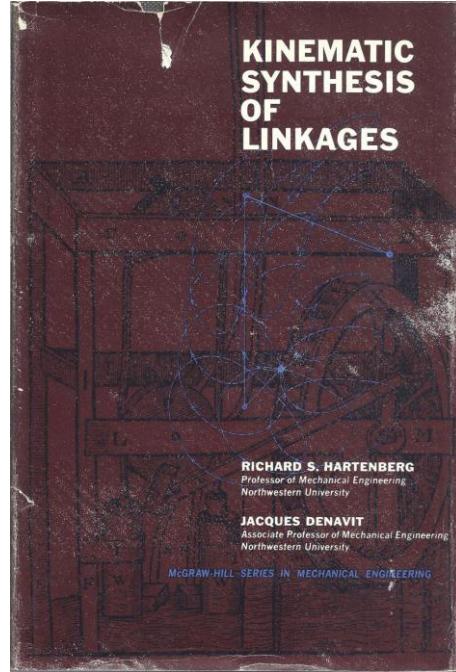
**RELATIVE location of joint {i-1} to joint {i}**

Joint Name	Parent Link	Child Link	x(m)	y(m)	z(m)	roll	pitch	yaw
joint_1	base_link	link_1	0	0	0.33	0	0	0
joint_2	link_1	link_2	0.35	0	0.42	0	0	0
joint_3	link_2	link_3	0	0	1.25	0	0	0
joint_4	link_3	link_4	0.96	0	-0.054	0	0	0
joint_5	link_4	link_5	0.54	0	0	0	0	0
joint_6	link_5	link_6	0.193	0	0	0	0	0
gripper-joint	link_6	gripper_link	0.11	0	0	0	0	0

# MECA 953 - Robotique

MMT5

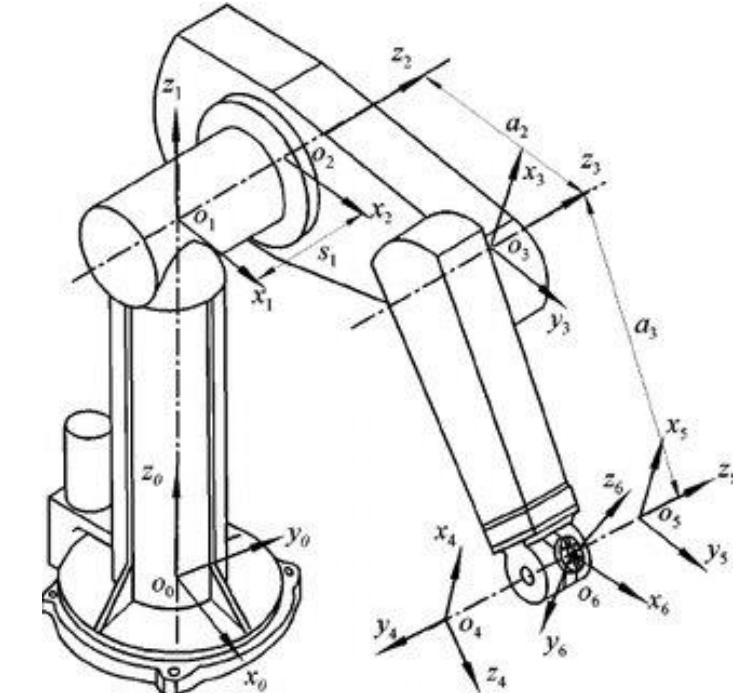
# Convention Denavit Hartenberg (1955)



Jacques **DENAVIT**  
(1930-2012)

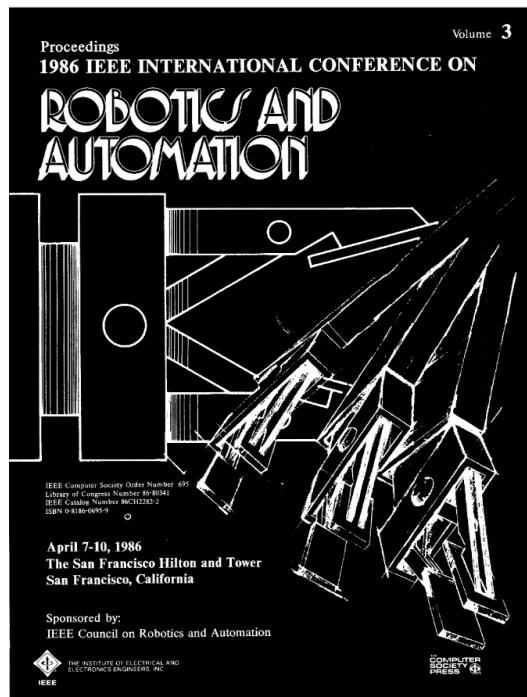


Richard Scheunemann **HARTENBERG**  
(1907-1997)



Initialement introduite en 1955 par Jacques Denavit et Richard S. Hartenberg, la méthode DH permet de normaliser, simplifier et rationaliser la modélisation géométrique d'un robot.

# Convention Denavit-Hartenberg modifiée: DH-KK (1986) (appelée aussi Khalil-Kleinfinger)



Wisama KHALIL

Jean-François KLEINFINGER

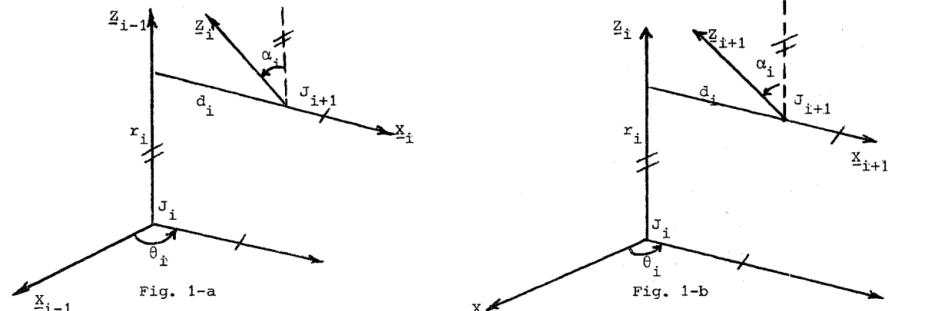


Figure 1. Denavit and Hartenberg Notation

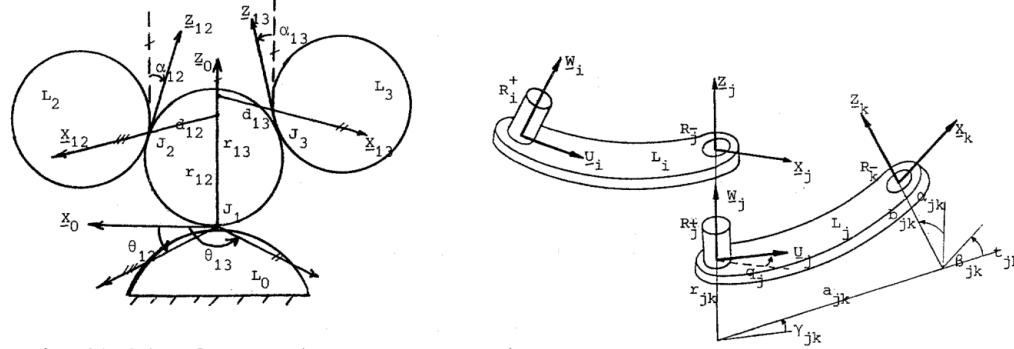


Figure 2. Ambiguities of D-H notation

La convention de Denavit-Hartenberg modifiée, appelée aussi convention de Khalil-Kleinfinger, est préconisée depuis 1986.  
(parce qu'elle permet un allègement du formalisme dans les exposés, portant sur les méthodes numériques par récurrence de la dynamique des robots).

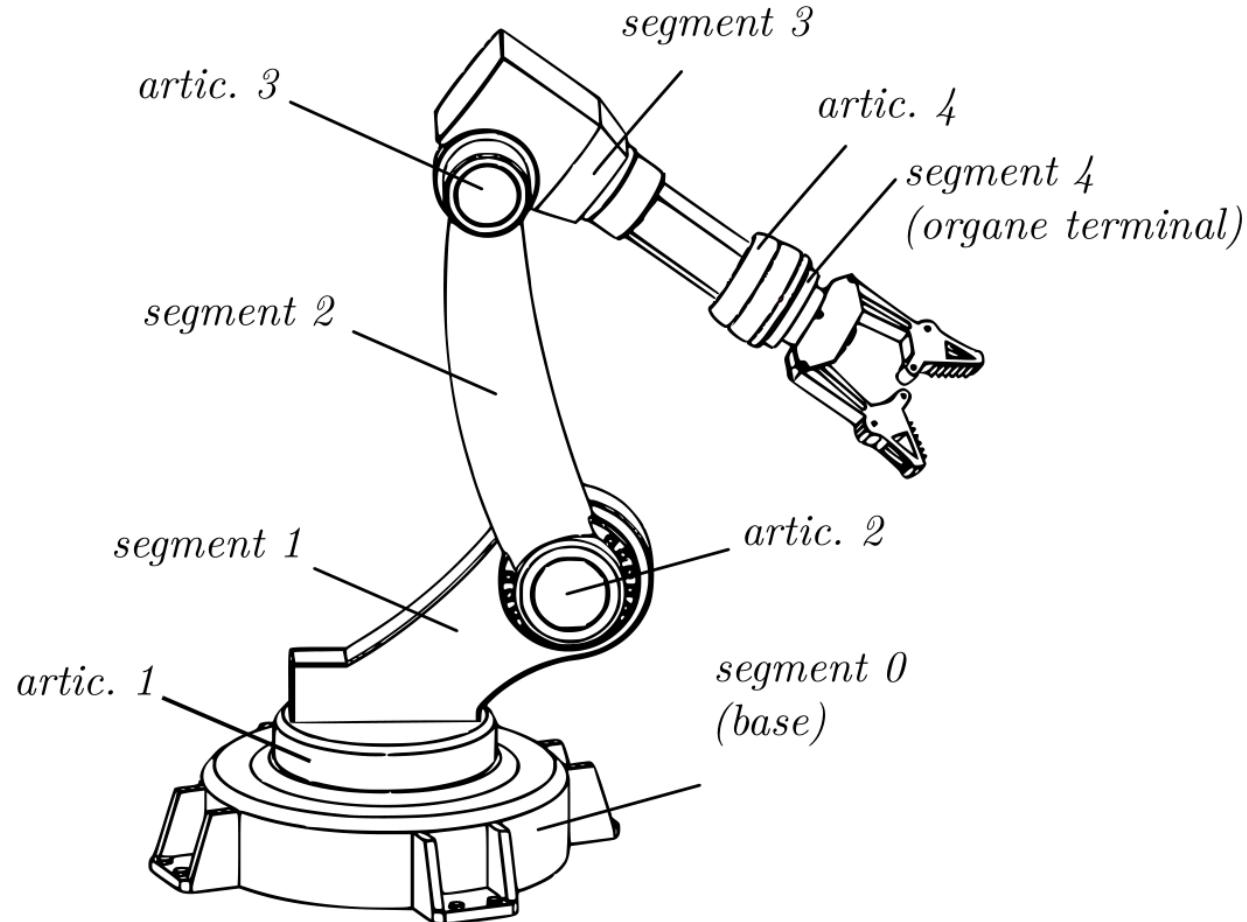
# Règles de paramétrage

## Règles de paramétrage

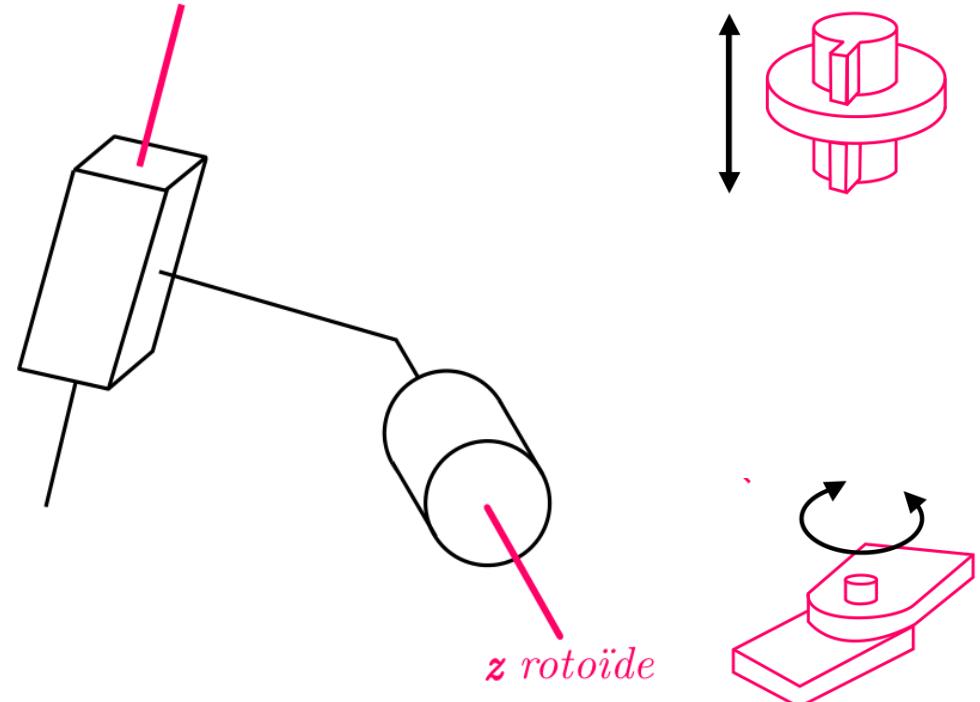
- Les segments sont numérotés dans l'ordre croissant, de la base (0) à l'effecteur (n).
- Le repère  $\mathcal{R}_i$  ( $O_i, \vec{X}_i \vec{Y}_i \vec{Z}_i$ ) est associé au segment  $i$ .
- L'axe  $\vec{Z}_i$  correspond à l'axe de l'articulation  $i$ . (pivot ou glissière).
- L'axe  $\vec{X}_i$  correspond à la perpendiculaire commune entre  $\vec{Z}_i$  et  $\vec{Z}_{i+1}$ . ( $\vec{X}_i = \vec{Z}_i \wedge \vec{Z}_{i+1}$ ).
- L'axe  $\vec{Y}_i$  est placé de façon à créer un repère orthogonal direct. ( $\vec{Y}_i = \vec{Z}_i \wedge \vec{X}_i$ ).
- L'origine  $O_i$  est située à intersection de  $\vec{X}_i$  et  $\vec{Z}_i$ .

## Règles de paramétrage

Numérotation des segments et liaisons & Définition des axes Z<sub>i</sub>

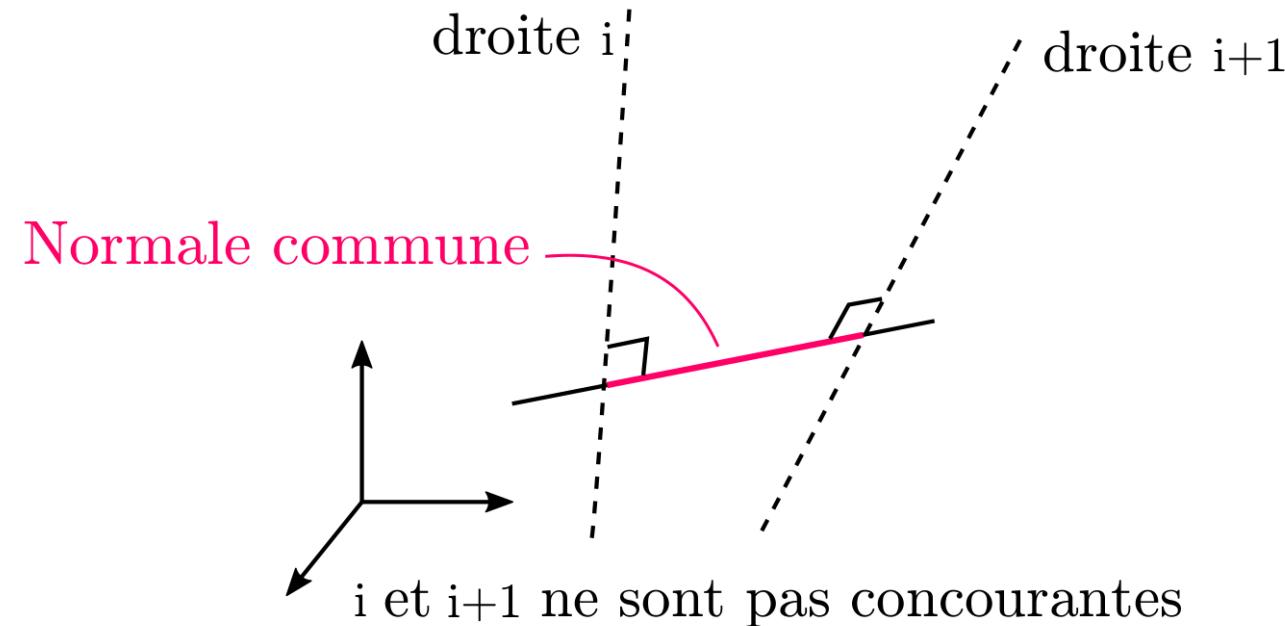


*z prismatique*



## Règles de paramétrage

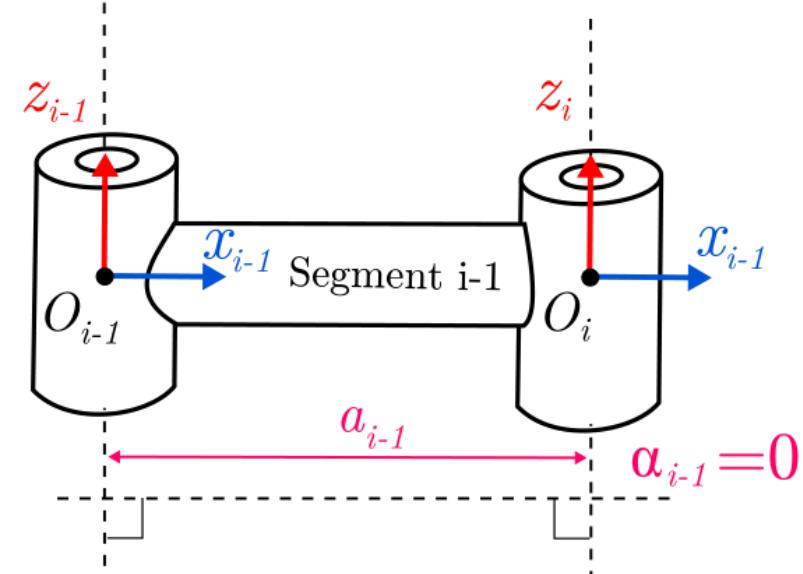
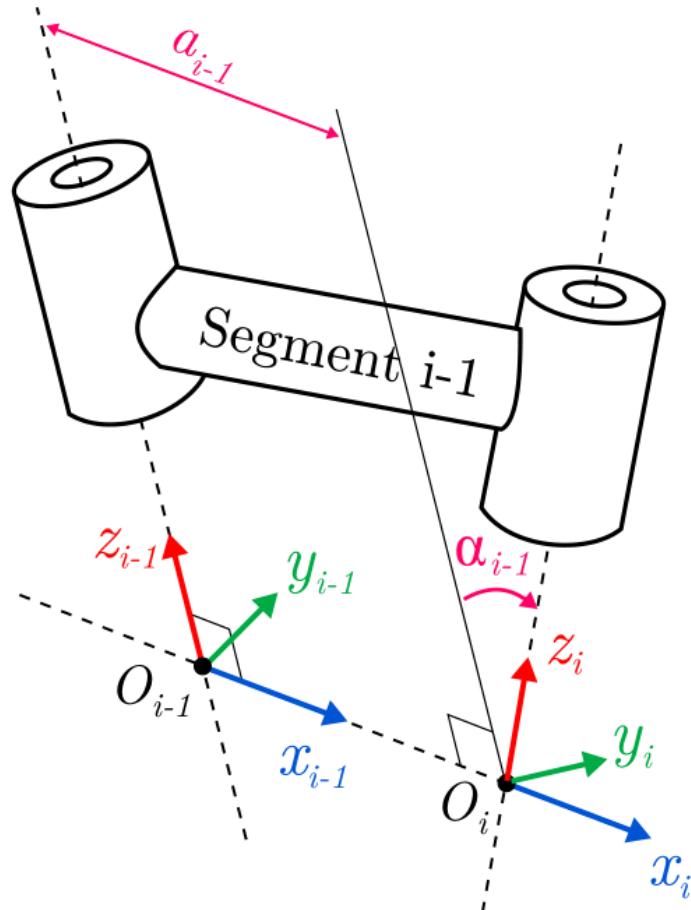
Numérotation des segments et liaisons & Définition des axes  $Z_i$



La **normale commune** entre deux droites est la droite qui contient le segment de distance minimale entre les deux droites.

# Règles de paramétrage

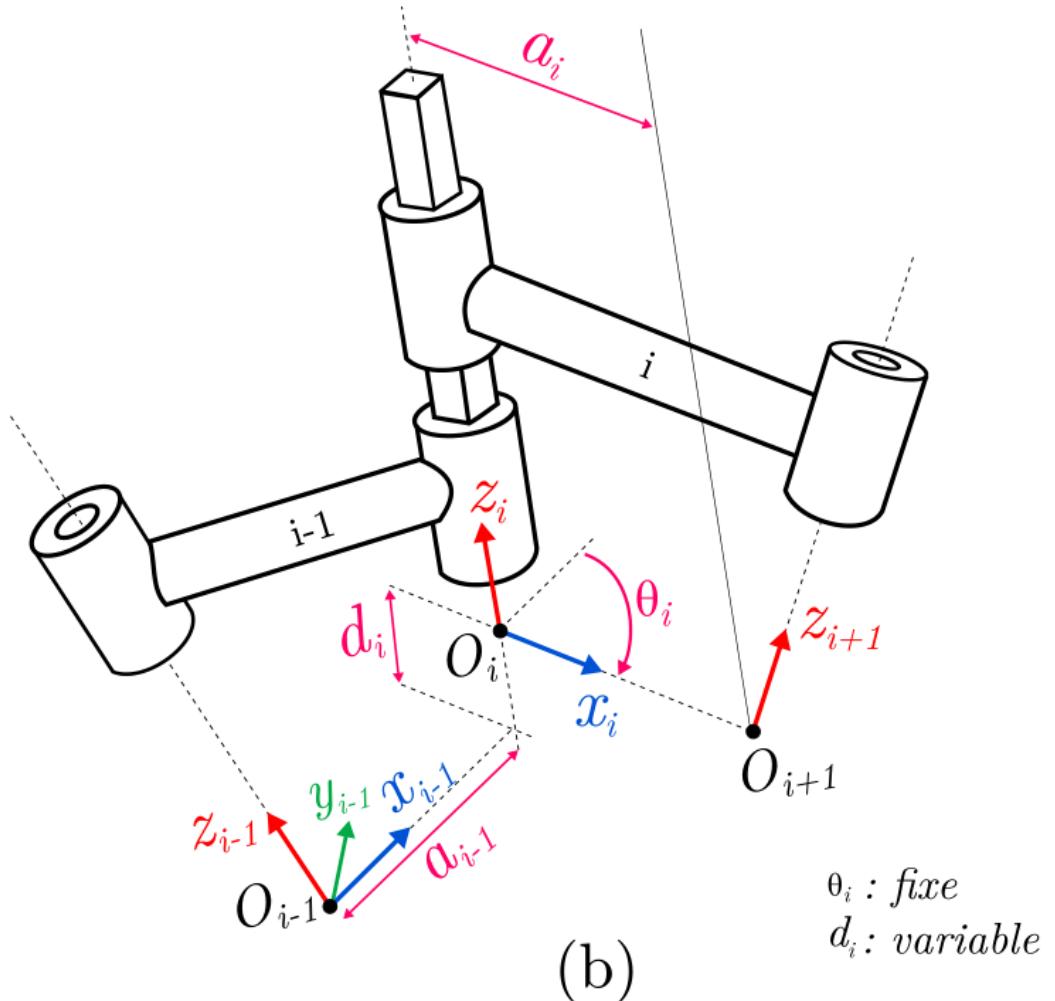
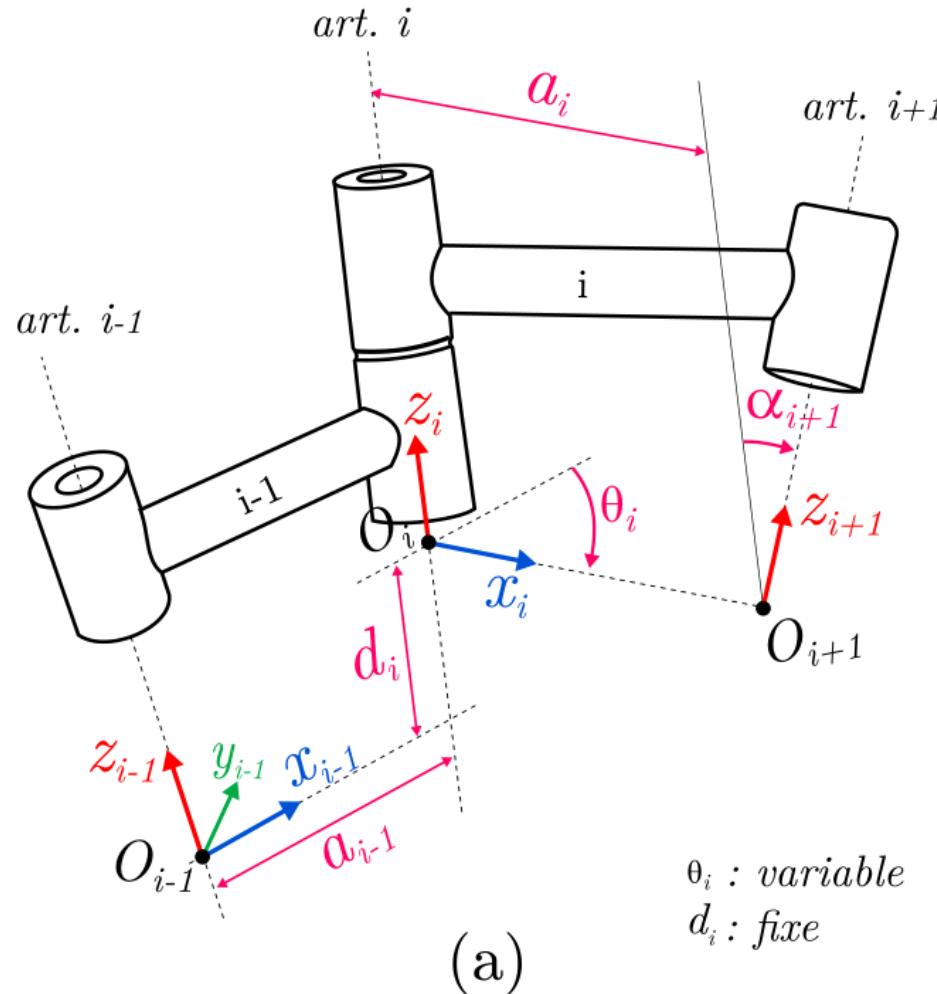
Paramétrage d'un segment (convention DH-KK)



(cas où axes i-1 et i parallèles)

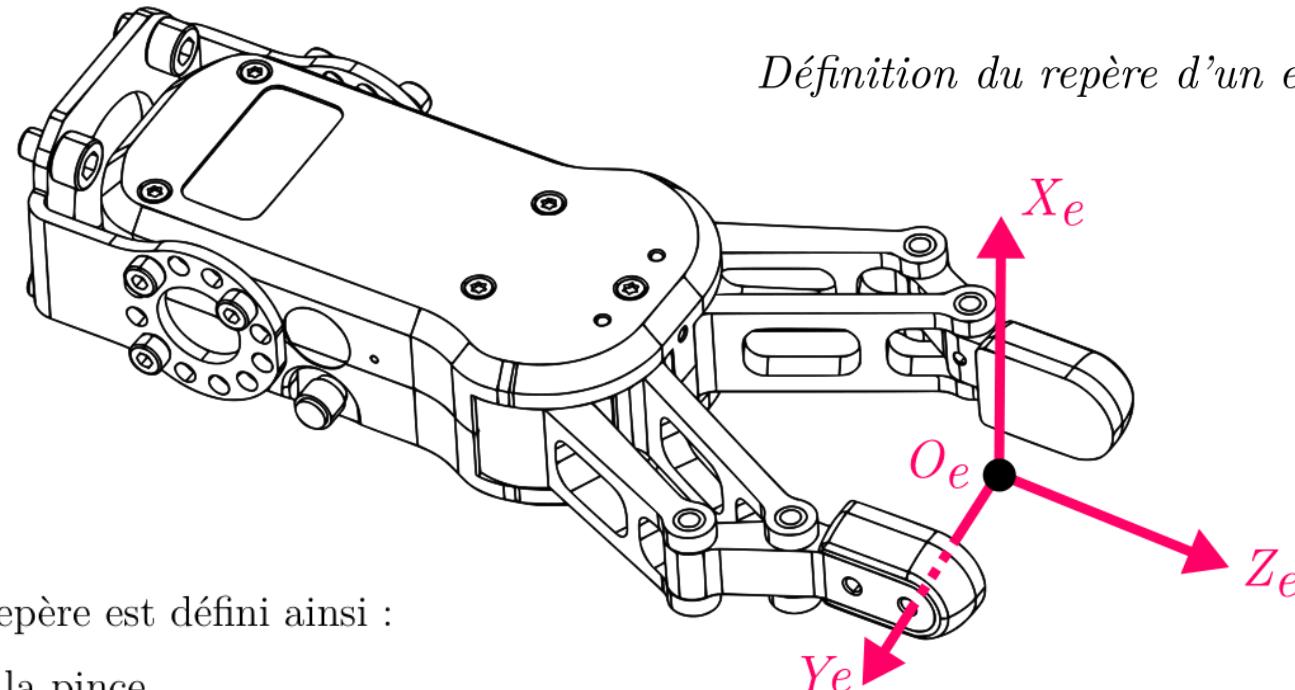
## Règles de paramétrage

Paramétrage (a) d'une liaison pivot, (b) d'une liaison glissière. (convention DH-KK)



# Règles de paramétrage

## Effecteur pince



Définition du repère d'un effecteur de type pince.

Si l'effecteur est une **pince** le repère est défini ainsi :

- Origine  $O_e$  : au centre de la pince
- $\vec{Z}_e$  : en direction de l'objet à attraper.
- $\vec{Y}_e$  : orthogonal à  $\vec{Z}_e$ , dans le plan de glissement des becs de la pince.
- $\vec{X}_e$  : orthogonal au deux autres axes pour avoir un repère orthogonal direct ( $\vec{X}_e = \vec{Y}_e \wedge \vec{Z}_e$ ).

## Convention DH modifiée (DH-KK)

Le passage de  $(\mathcal{R}_{i-1})$  à  $(\mathcal{R}_i)$  s'exprime en fonction des quatre paramètres DH modifiés suivants :  $a_{i-1}$  (excentricité),  $\alpha_{i-1}$  (torsion),  $d_i$  (longueur),  $\theta_i$  (angle).

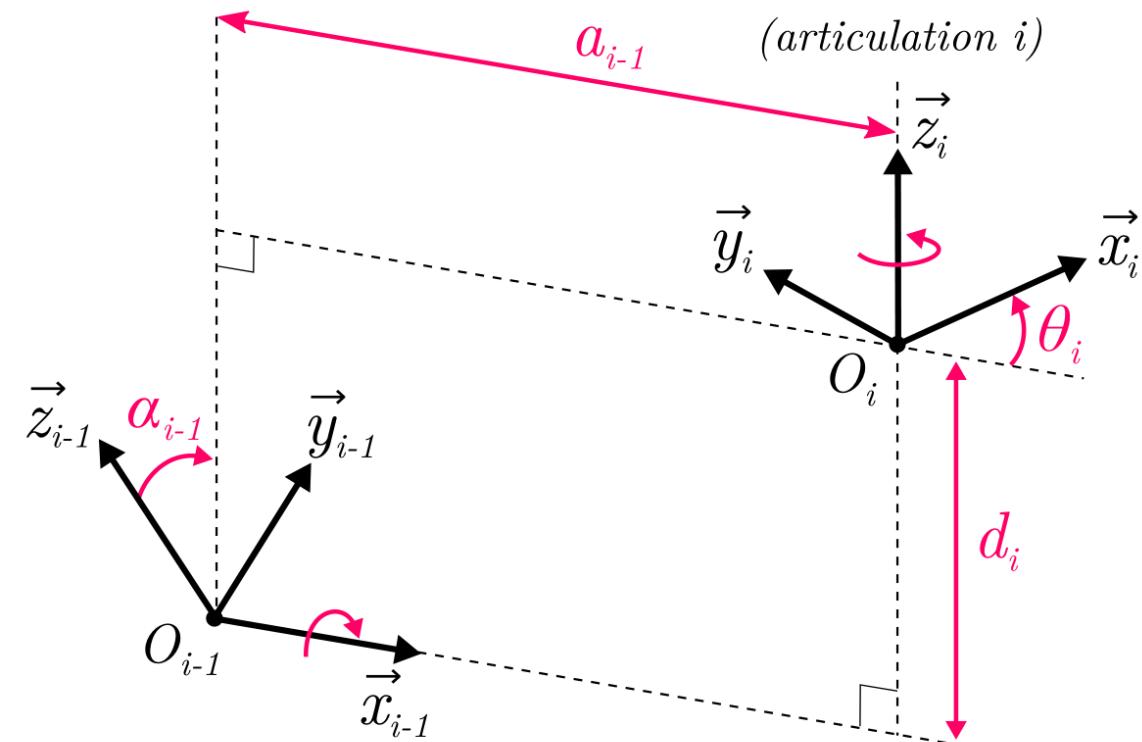
$a_{i-1}$  : Distance de  $Z_{i-1}$  vers  $Z_i$ , le long de  $X_{i-1}$ .

$\alpha_{i-1}$  : Angle entre  $Z_{i-1}$  et  $Z_i$ , autour de l'axe  $X_{i-1}$ .

$d_i$  : Distance de  $X_{i-1}$  vers  $X_i$ , le long de  $Z_i$ .

$\theta_i$  : Angle entre  $X_{i-1}$  et  $X_i$ , autour de l'axe  $Z_i$ .

Paramètres de la convention DH-KK.



# Convention DH modifiée (DH-KK)

## Algorithme de paramétrage

Symbol	Name	Description
$a_{i-1}$	Link Length	$Z_{i-1} \xrightarrow[\text{@ } X_{i-1}]{\perp, \text{distance}} Z_i$
$\alpha_{i-1}$	Twist Angle	$Z_{i-1} \xrightarrow[\text{@ } X_{i-1}]{\curvearrowright, \text{rotation}} Z_i$
$d_i$	Joint Offset	$X_{i-1} \xrightarrow[\text{@ } Z_i]{\perp, \text{distance}} X_i$
$\theta_i$	Joint Angle	$X_{i-1} \xrightarrow[\text{@ } Z_i]{\curvearrowright, \text{rotation}} X_i$

Paramètres de la convention DH-KK.

# Convention DH modifiée (DH-KK)

## Algorithme de paramétrage

Étape 1 **Identification des segments** : Chaque repère ( $\mathcal{R}_i$ ) est lié au segment  $i$   
**Identification des articulations de 1 à n** : L'axe  $Z_i$  du repère ( $\mathcal{R}_i$ ) coïncide avec l'articulation  $i$ . Le segment  $i$  possède 2 axes :  $Z_i$  et  $Z_{i+1}$ . L'axe  $Z_i$  est lié à l'articulation  $i$  et l'axe  $Z_{i+1}$  est lié à l'articulation  $i + 1$ .

Étape 2 **Choisir**  $Z_i$  le long de l'axe des articulations  $i$ .

Étape 3 **Identifier la normale commune entre**  $Z_i$  **et**  $Z_{i+1}$ . L'origine du repère  $O_i$  est située à l'intersection de la normale commune  $a_i$  et l'axe  $Z_i$ .

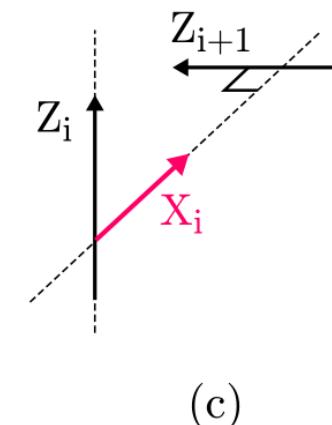
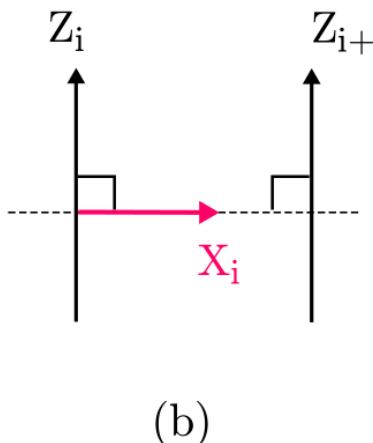
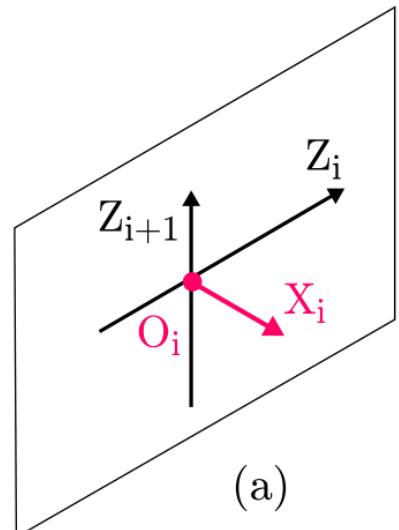
- Si  $Z_i$  et  $Z_{i+1}$  sont concourants,  $O_i$  est située au point d'intersection.
- Si  $Z_i$  et  $Z_{i+1}$  sont parallèles, le choix de l'origine  $Z_i$  est arbitraire (on choisit généralement une solution qui donne  $d_i=0$ ).
- De même si il s'agit d'une articulation prismatique, la liberté est donnée quand à la position de l'origine du repère.

# Convention DH modifiée (DH-KK)

## Algorithme de paramétrage

Étape 4 Choisir  $X_i$  le long de la normale commune  $a_i$  et dirigé de  $Z_i$  vers  $Z_{i+1}$ .

- Si  $Z_i$  et  $Z_{i+1}$  s'intersectent,  $X_i$  est perpendiculaire au plan contenant les deux axes et le choix de la direction de  $X_i$  est libre.
- Si  $Z_i$  et  $Z_{i+1}$  sont parallèles,  $X_i$  est choisi de tel sorte qu'il intersecte  $Z_{i-1}$ .



(a)  $\vec{Z}_i$  et  $\vec{Z}_{i+1}$  s'intersectent. (b,c)  $\vec{Z}_i$  et  $\vec{Z}_{i+1}$  sont parallèles.

# Convention DH modifiée (DH-KK)

## Algorithme de paramétrage

Étape 5 **Choisir**  $Y_i$  pour obtenir un trièdre direct avec  $Z_i$  et  $X_i$  soit  $Y_i = Z_i \wedge X_i$ . (Généralement on ne représente pas les axes  $Y_i$  pour ne pas encombrer le schéma)

Étape 6 **Attribution du repère de Base ( $\mathcal{R}_0$ )** : Le repère de base est lié au segment 0. Le repère est placé arbitrairement mais le choix le plus simple consiste à prendre  $(\mathcal{R}_0)$  confondu avec  $(\mathcal{R}_1)$  quand  $q_1 = 0$ . (On a alors  $d_0=0$  et  $\alpha_0=0$ ,  $r_1=0$  si l'articulation est rotatoire et  $\theta_1=0$  si l'articulation est prismatique)

# Convention DH modifiée (DH-KK)

## Algorithme de paramétrage

### Étape 7 Assignation du repère de l'organe terminal $n$ :

Si l'articulation  $n$  est rotatoire, la direction de  $x_n$  est choisie le long de  $x_{n-1}$  quand  $\theta_n = 0$  et l'origine du repère  $n$  est choisie telle que  $r_n = 0$ .

Si l'articulation  $n$  est prismatique, la direction de  $x_n$  est choisie telle que  $\theta_n = 0$  et l'origine du repère  $n$  est définie à l'intersection de  $x_{n-1}$  et  $z_n$  tel que  $r_n = 0$ .

### Étape 8 Remplissage du tableau des paramètres

	segment $i$	$\sigma_i$	$a_{i-1}$	$\alpha_{i-1}$	$d_i$	$\theta_i$
${}^0T_1$	1					
${}^1T_2$	2					
$\dots$	$\dots$					
${}^{n-1}T_n$	$n$					

# Convention DH modifiée (DH-KK)

Variables articulaires :  $q_i$

Le nombre de variables articulaires est égale au nombre d'axes du robot. La variable articulaire  $q_i$ , associée à chaque articulation  $i$ , définit la position relative de l'actionneur (moteur au niveau de la liaison).  $q_i$  est soit  $\theta_i$ , soit  $d_i$ , selon que cette articulation est respectivement de type rotatoire ou prismatique :

$$q_i = (1 - \sigma_i)\theta_i + \sigma_i d_i$$

articulation $i$	$\sigma_i$	$q_i$	constantes
ratoire	0	$\theta_i$	$\alpha_{i-1}, a_{i-1}, d_i$
prismatique	1	$d_i$	$\alpha_{i-1}, a_{i-1}, \theta_i$

## Matrice de passage homogène Denavit-Hartenberg

La matrice de transformation homogène (position et orientation) entre 2 repères adjacents  $\{\mathcal{R}_{i-1}\}$ ,  $\{\mathcal{R}_i\}$  peut être décomposée en 4 transformations élémentaires :

$\mathbf{Trans}_{x_{i-1}}(a_{i-1})$  : Translation le long de  $\mathbf{X}$  d'une distance  $a$ .

$\mathbf{Rot}_{x_{i-1}}(\alpha_{i-1})$  : Rotation autour de  $\mathbf{X}$  d'un angle  $\alpha$ .

$\mathbf{Trans}_{z_i}(d_i)$  : Translation le long de  $\mathbf{Z}$  d'une distance  $d$ .

$\mathbf{Rot}_{z_i}(\theta_i)$  : Rotation autour de  $\mathbf{Z}$  d'un angle  $\theta$ .

$$\mathbf{Trans}_{x_{i-1}}(a_{i-1}) = \begin{bmatrix} I & \vec{t} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & a_{i-1} \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{Rot}_{x_{i-1}}(\alpha_{i-1}) = \begin{bmatrix} R & \vec{0} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & C\alpha_{i-1} & -S\alpha_{i-1} & 0 \\ 0 & S\alpha_{i-1} & C\alpha_{i-1} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{Trans}_{z_i}(d_i) = \begin{bmatrix} I & \vec{t} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{Rot}_{z_i}(\theta_i) = \begin{bmatrix} R & \vec{0} \end{bmatrix} = \begin{bmatrix} C\theta_i & -S\theta_i & 0 & 0 \\ S\theta_i & C\theta_i & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

## Matrice de passage homogène Denavit-Hartenberg

Le produit des matrices de passage successives donne l'expression de la matrice de transformation qui amène le repère  $\{\mathcal{R}_{i-1}\}$  au repère  $\{\mathcal{R}_i\}$  :

$${}^{i-1}\mathbf{T}_i = \mathbf{Trans}_{x_{i-1}}(a_{i-1}) \cdot \mathbf{Rot}_{x_{i-1}}(\alpha_{i-1}) \cdot \mathbf{Trans}_{z_i}(d_i) \cdot \mathbf{Rot}_{z_i}(\theta_i)$$

$${}^{i-1}\mathbf{T}_i = \begin{bmatrix} 1 & 0 & 0 & a_{i-1} \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & C\alpha_{i-1} & -S\alpha_{i-1} & 0 \\ 0 & S\alpha_{i-1} & C\alpha_{i-1} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} C\theta_i & -S\theta_i & 0 & 0 \\ S\theta_i & C\theta_i & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^{i-1}\mathbf{T}_i = \begin{bmatrix} C\theta_i & -S\theta_i & 0 & a_{i-1} \\ S\theta_i C\alpha_{i-1} & C\theta_i C\alpha_{i-1} & -S\alpha_{i-1} & -d_i S\alpha_{i-1} \\ S\theta_i S\alpha_{i-1} & C\theta_i S\alpha_{i-1} & C\alpha_{i-1} & d_i C\alpha_{i-1} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^{i-1}\mathbf{T}_i = \begin{bmatrix} {}^{i-1}\mathbf{R}_i & {}^{i-1}\mathbf{O}_i \\ 0 & 1 \end{bmatrix}$$

# Matrice de passage homogène Denavit-Hartenberg

## Transformation inverse

La transformation inverse  ${}^i\mathbf{T}_{i-1}$  est donnée par :

$${}^i\mathbf{T}_{i-1} = \mathbf{Rot}_{z_i}(-\theta_i) \cdot \mathbf{Trans}_{z_i}(-d_i) \cdot \mathbf{Rot}_{x_{i-1}}(-\alpha_{i-1}) \cdot \mathbf{Trans}_{x_{i-1}}(-a_{i-1})$$

$${}^i\mathbf{T}_{i-1} = \begin{bmatrix} {}^{i-1}\mathbf{R}_i^T & -a_{i-1} C\theta_i \\ 0 & a_{i-1} S\theta_i \\ 0 & -d_i \\ 0 & 0 & 1 \end{bmatrix}$$

# Matrice de passage homogène Denavit-Hartenberg

Comparaisons des paramètres de Denavit-Hartenberg classiques et modifiés

La méthode de Denavit-hartenberg "originale" est bien adaptée pour des structures ouvertes simples, mais présente des ambiguïtés lorsqu'elle est appliquée sur des robots à structures fermées ou arborescentes. La variante dite "modifiée" permet de définir les paramètres à partir de deux solides seulement (trois axes caractéristiques de liaison) au lieu de trois pour la convention DH classique.

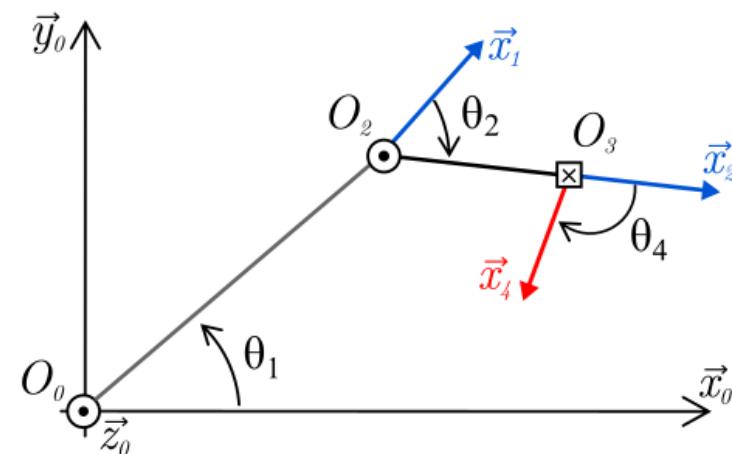
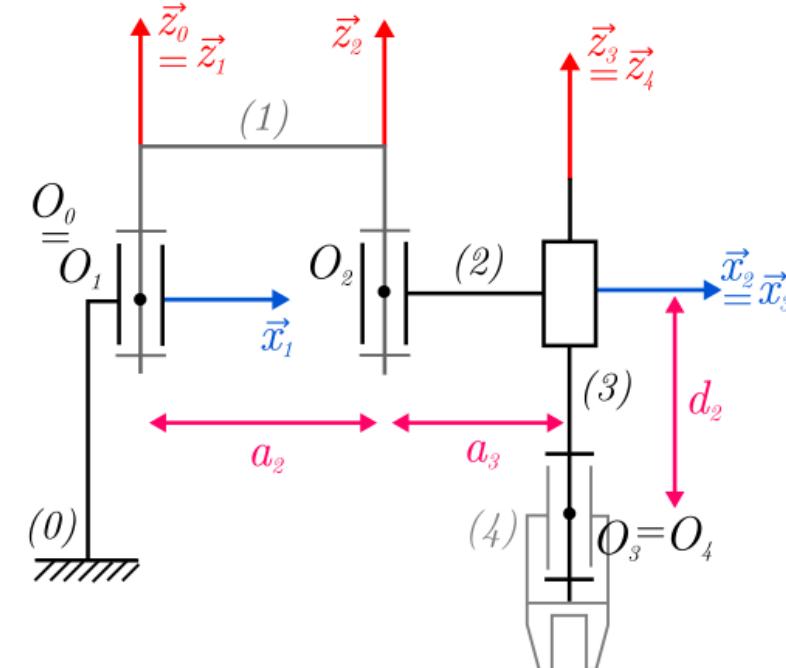
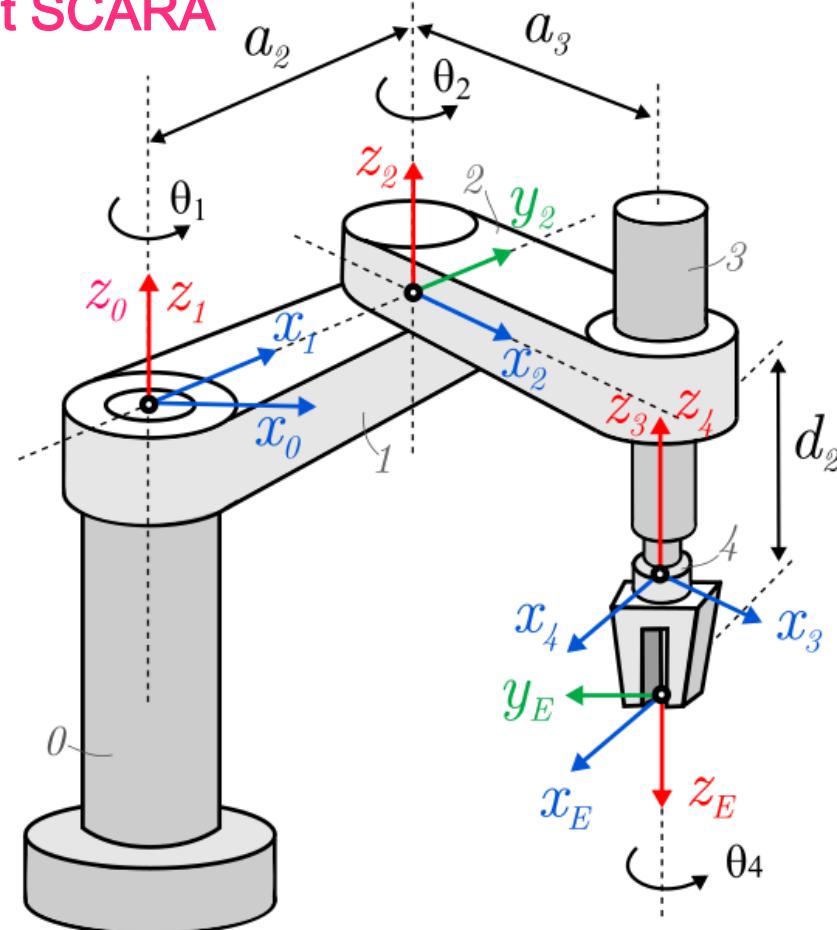
# Matrice de passage homogène Denavit-Hartenberg

Comparaisons des paramètres de Denavit-Hartenberg classiques et modifiés

Paramètres	Convention originale DH	Convention modifiée DH-KK
axe de liaison	$z_{i-1}$ pour l'articulation $i$	$z_i$ pour l'articulation $i$
longueur : $a_i$	distance de $O_i$ à l'intersection de $z_{i-1}$ et $x_i$ , le long de $x_i$	distance de $z_i$ à $z_{i+1}$ , le long de $x_i$
torsion : $\alpha_i$	angle de $z_{i-1}$ à $z_i$ , autour de $x_i$	angle de $z_i$ à $z_{i+1}$ , autour de $x_i$
longueur : $d_i$	distance de $O_{i-1}$ à l'intersection de $z_{i-1}$ et $x_i$ , le long de $z_{i-1}$	distance de $x_{i-1}$ à $x_i$ , le long de $z_i$
angle : $\theta_i$	angle de $x_{i-1}$ à $x_i$ , autour de $z_{i-1}$	angle de $x_{i-1}$ à $x_i$ , autour de $z_i$
$i-1T_i$	$\begin{bmatrix} C\theta_i & -C\alpha_i S\theta_i & S\alpha_i S\theta_i & d_i C\theta_i \\ S\theta_i & C\alpha_i C\theta_i & -S\alpha_i C\theta_i & d_i S\theta_i \\ 0 & S\alpha_i & C\alpha_i & a_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$\begin{bmatrix} C\theta_i & -S\theta_i & 0 & a_{i-1} \\ S\theta_i C\alpha_{i-1} & C\theta_i C\alpha_{i-1} & -S\alpha_{i-1} & -d_i S\alpha_{i-1} \\ S\theta_i S\alpha_{i-1} & C\theta_i S\alpha_{i-1} & C\alpha_{i-1} & d_i C\alpha_{i-1} \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$iT_{i-1}$	$\begin{bmatrix} C\theta_i & S\theta_i & 0 & -d_i \\ -C\alpha_i S\theta_i & C\alpha_i C\theta_i & S\alpha_i & -a_i S\alpha_i \\ S\alpha_i S\theta_i & -S\alpha_i C\theta_i & C\alpha_i & -a_i C\alpha_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$\begin{bmatrix} C\theta_i & S\theta_i C\alpha_{i-1} & S\theta_i S\alpha_{i-1} & -a_{i-1} C\theta_i \\ -S\theta_i & C\theta_i C\alpha_{i-1} & C\theta_i S\alpha_{i-1} & a_{i-1} S\theta_i \\ 0 & -S\alpha_{i-1} & C\alpha_{i-1} & -d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$

# Paramétrage

Exemple Robot SCARA



# Paramétrage

## Exemple Robot SCARA

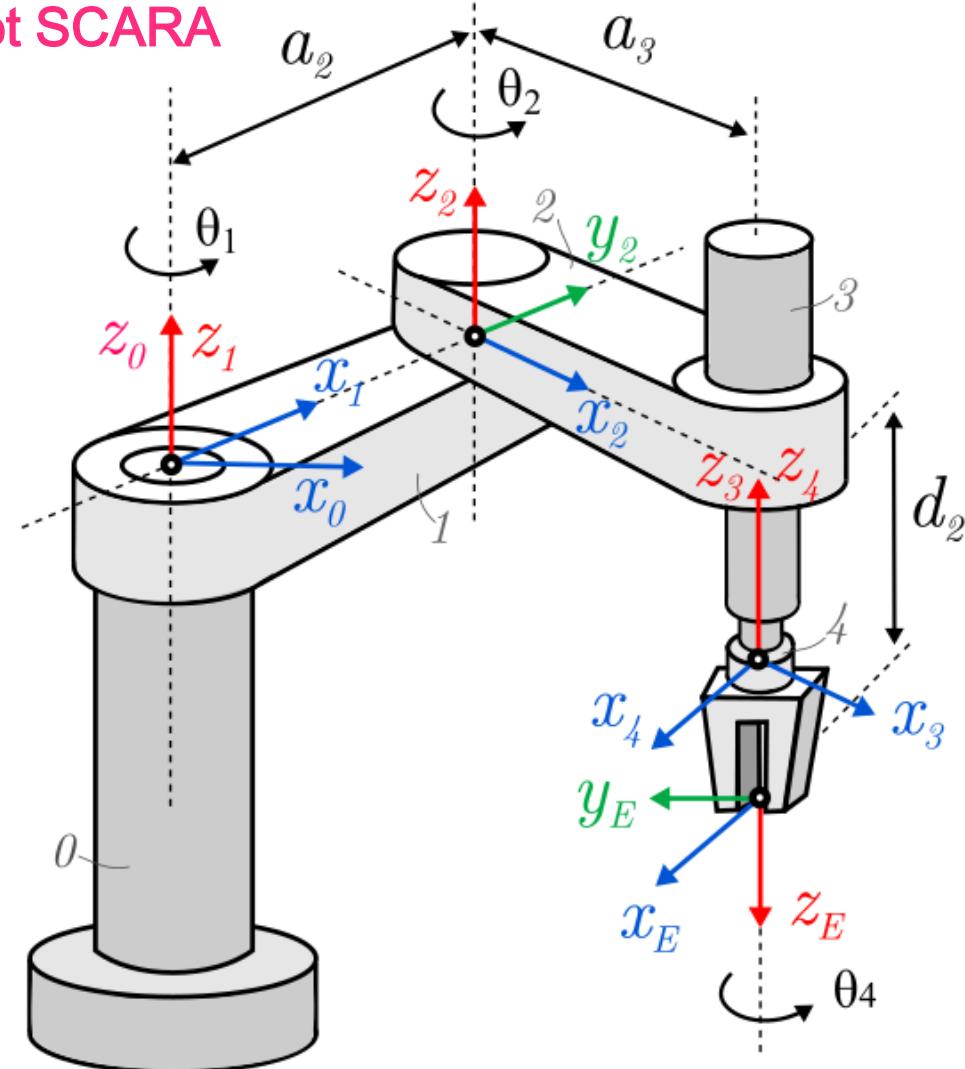


Tableau des paramètres DHKK - Robot Scara

$i$	$\sigma_i$	$a_{i-1}$	$\alpha_{i-1}$	$d_i$	$\theta_i$
1	0	0	0	0	$\theta_1$
2	0	$a_2$	0	0	$\theta_2$
3	1	$a_3$	0	$d_2$	0
4	0	0	0	0	$\theta_4$

# Paramétrage

## Stäubli RX-90

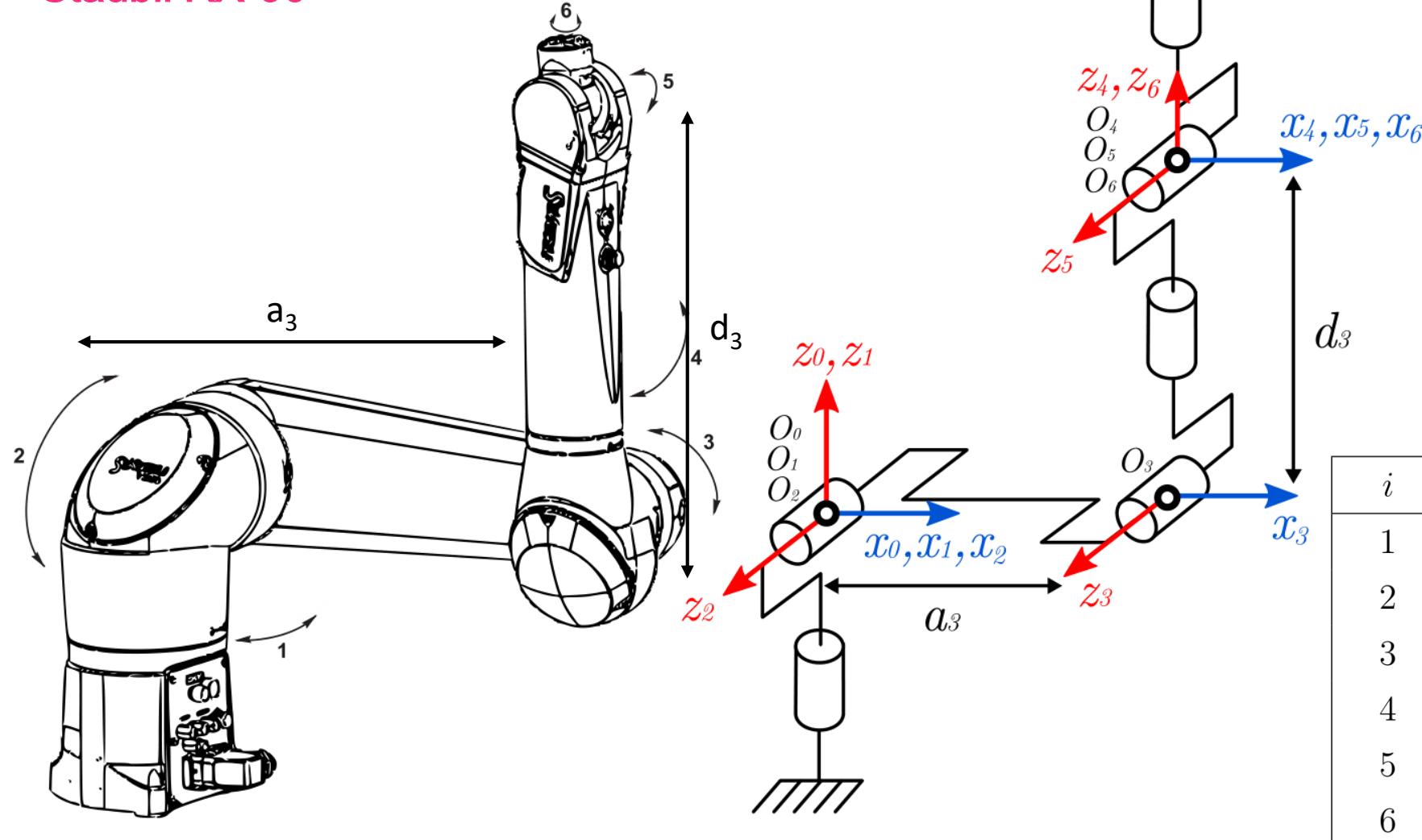
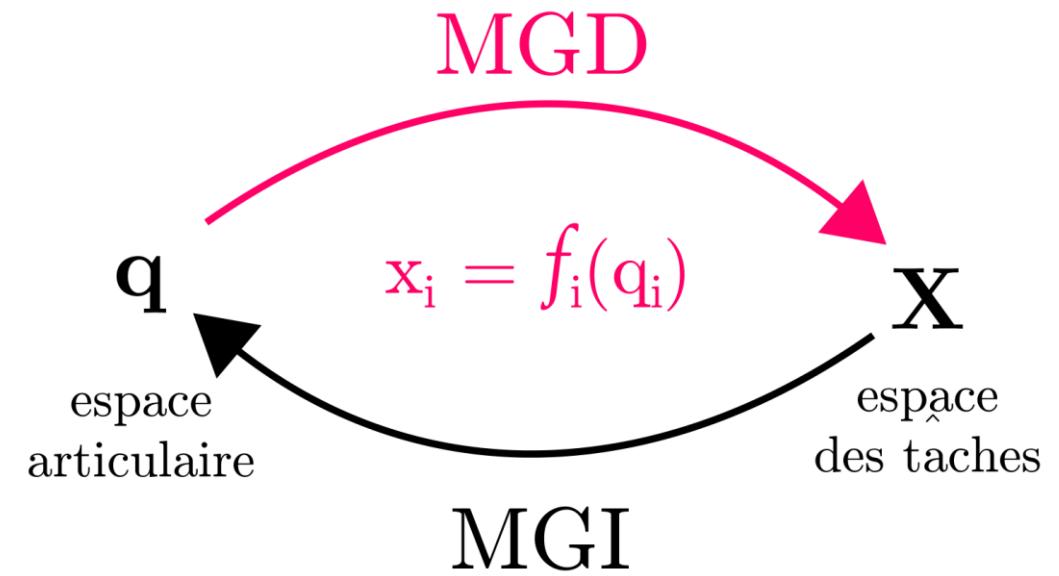
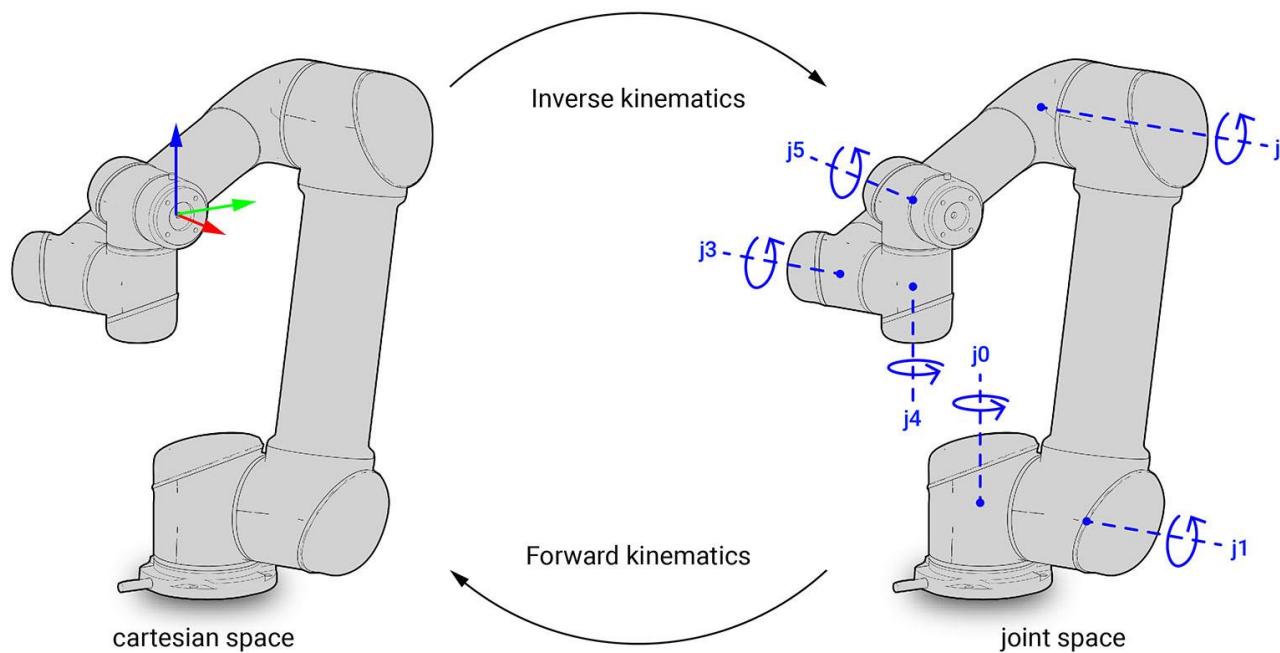


Tableau des paramètres DHKK  
Robot Stäubli RX-90

$i$	$\sigma_i$	$a_{i-1}$	$\alpha_{i-1}$	$d_i$	$\theta_i$
1	0	0	0	0	$\theta_1$
2	0	0	$\frac{\pi}{2}$	0	$\theta_2$
3	0	$a_3$	0	0	$\theta_3$
4	0	0	$-\frac{\pi}{2}$	$d_3$	$\theta_4$
5	0	0	$\frac{\pi}{2}$	0	$\theta_5$
6	0	0	$-\frac{\pi}{2}$	0	$\theta_6$

## Modèle Géométrique Direct : MGD

Le MGD est l'ensemble des relations qui donne la *pose* de l'organe terminal  $\mathbf{X}$  (*les coordonnées opérationnelles*) en fonction de la configuration du robot définie par les variables articulaires  $\mathbf{q}$  des différentes liaisons.

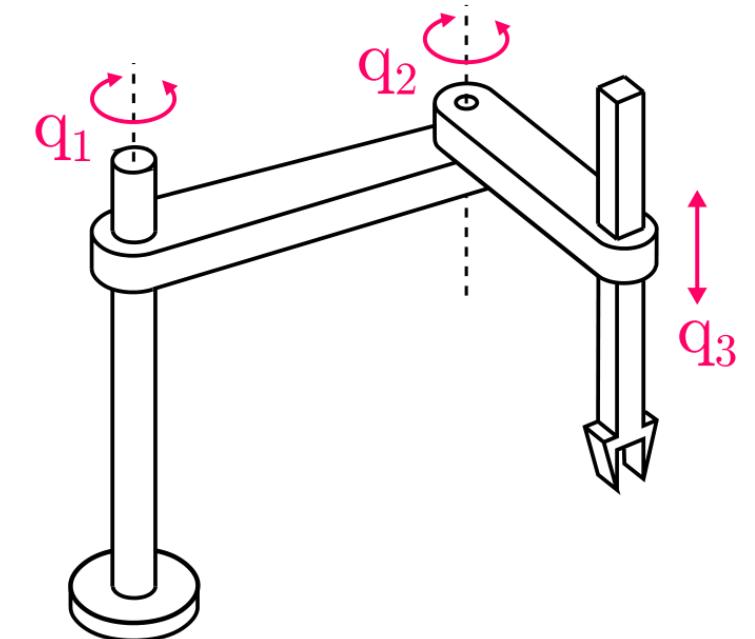


## Modèle Géométrique Direct : MGD

$$\mathbf{X} = f(\mathbf{q}) = \begin{bmatrix} \text{position de l'effecteur dans } \{\mathcal{R}_0\} \\ \text{orientation de } (\mathcal{R}_n) \text{ par rapport à } \{\mathcal{R}_0\} \end{bmatrix}$$

$$\mathbf{q} = \begin{bmatrix} q_1 \\ q_2 \\ \dots \\ q_n \end{bmatrix} \quad \mathbf{X} = \begin{bmatrix} x_1 \\ x_2 \\ \dots \\ x_m \end{bmatrix}_{(0)}$$

$$q_i = \begin{cases} \theta & \text{pour une liaison rotatoire} \\ a_i & \text{pour une liaison prismatique} \end{cases}$$



## Modèle Géométrique Direct : MGD

Le contenu du vecteur  $\vec{X}$  peut être défini de plusieurs façons, suivant la représentation choisie :

- Avec la position cartésienne et les cosinus directeurs pour l'orientation :

$$\mathbf{X} = [P_x, P_y, P_z, s_x, s_y, s_z, n_x, n_y, a_z, a_y, a_z]^T$$

- Avec la position cartésienne et les angles RPY pour l'orientation :

$$\mathbf{X} = [P_x, P_y, P_z, \varphi, \theta, \psi]^T$$

## Modèle Géométrique Direct : MGD

Avec le MGD dans le cas d'un robot série, si les variables des articulations sont connues, la pose de l'effecteur peut être calculée d'une **manière unique** en utilisant la matrice de transformation homogène  ${}^0T_n$  définie par :

$${}^0T_n = {}^0T_1(q_1) {}^1T_2(q_2) \dots {}^{n-1}T_n(q_n)$$

En résumé : Le calcul du MGD conduit à identifier la matrice de transformation entre le repère de référence  $\{\mathcal{R}_0\}$  et celui de l'effecteur  $\{\mathcal{R}_n\}$  en identifiant les paramètres de la matrice  ${}^0T_n$  :

$${}^0T_n = \begin{bmatrix} s_x & n_x & a_x & P_x \\ s_y & n_y & a_y & P_y \\ s_z & n_z & a_z & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{avec } P_x, P_y, P_z \text{ coordonnées du point P dans } \{\mathcal{R}_0\}$$

## Repère global (WORLD) - Repère de l'outil (TOOL)

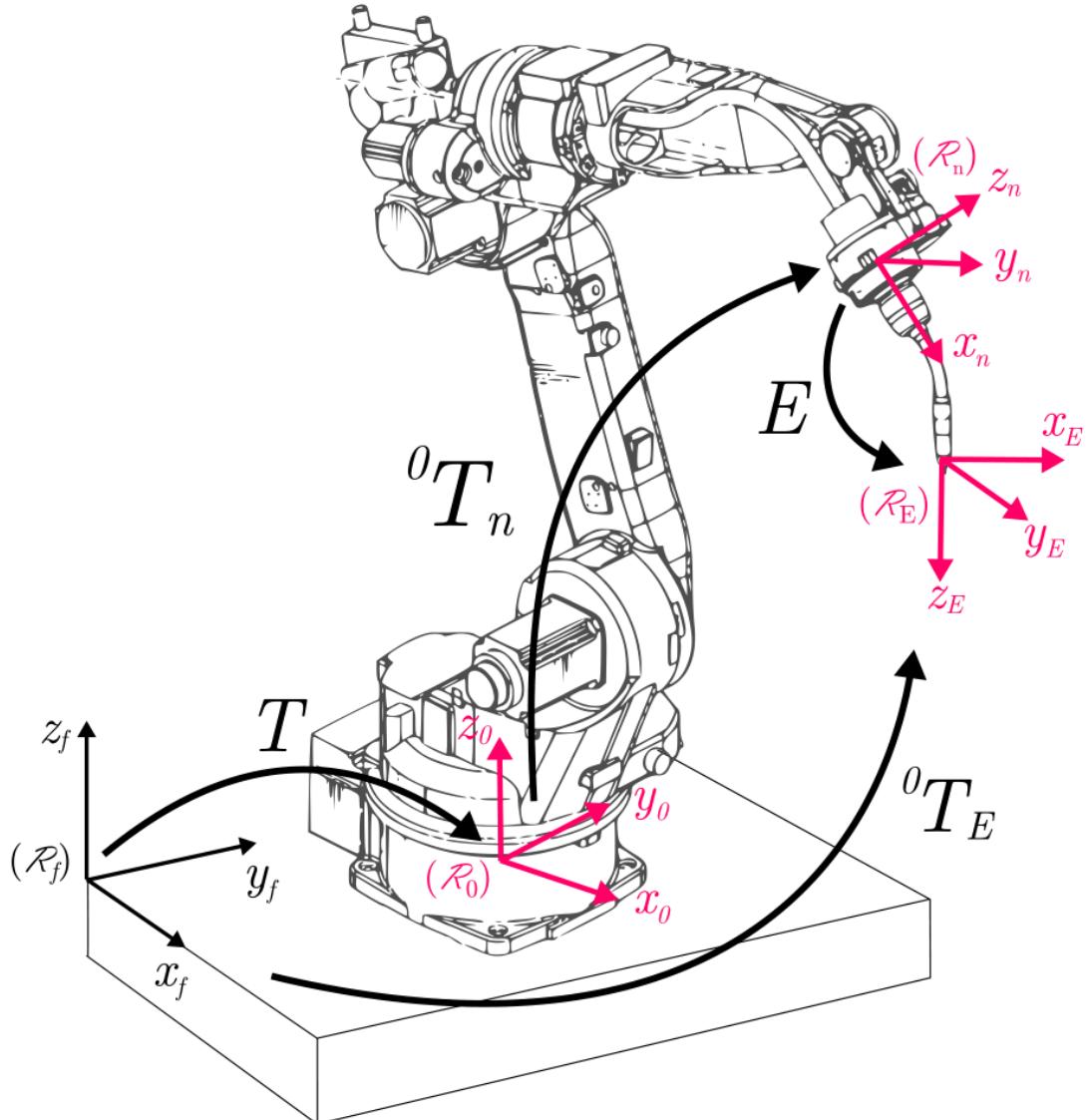
Généralement on définit un repère global (WORLD frame) ( $\mathcal{R}_f$ ) qui peut être différent du repère de base  $\{\mathcal{R}_0\}$  associé au bâti du robot. Un repère associé à l'outil (TOOL frame) ( $\mathcal{R}_E$ ) est aussi souvent utilisé (pratique lorsque l'on change souvent d'outil avec des dimensions différentes).

Les commandes de programmation de certains robots font apparaître :

- $Z = {}^fT_0$  : la matrice de transformation qui définit  $\{\mathcal{R}_0\}$  dans le repère de référence ( $\mathcal{R}_f$ )
- $E = {}^nT_E$  : la matrice de transformation qui définit  $(\mathcal{R}_E)$  dans le repère du terminal ( $\mathcal{R}_n$ )

Le repérage de l'outil est définie directement dans le repère global par :

$${}^nT_E = Z \ {}^0T_n(q) E$$



# Modèle Géométrique Direct : MGD

## Exemple Robot SCARA

A partir du tableau des paramètres DHKK on peut calculer les matrices de transfert :

$${}^0T_1 = \begin{bmatrix} C\theta_1 & -S\theta_1 & 0 & 0 \\ S\theta_1 & C\theta_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^1T_2 = \begin{bmatrix} C\theta_2 & -S\theta_2 & 0 & a_2 \\ S\theta_2 & C\theta_2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^2T_3 = \begin{bmatrix} 1 & 0 & 0 & a_3 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^3T_4 = \begin{bmatrix} C\theta_4 & -S\theta_4 & 0 & 0 \\ S\theta_4 & C\theta_4 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

# Modèle Géométrique Direct : MGD

## Exemple Robot SCARA

$$\begin{aligned} U_4 &= I_4 \\ U_3 &= {}^3T_4 U_4 = {}^3T_4 \end{aligned}$$

$$U_2 = {}^2T_3 U_3 = {}^2T_3 {}^3T_4 = {}^2T_4 = \begin{bmatrix} C\theta_4 & -S\theta_4 & 0 & a_3 \\ S\theta_4 & C\theta_4 & 0 & 0 \\ 0 & 0 & 1 & -d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$U_1 = {}^1T_2 U_2 = {}^1T_2 {}^2T_3 {}^3T_4 = {}^1T_4 = \begin{bmatrix} C(\theta_2 + \theta_4) & -S(\theta_2 + \theta_4) & 0 & a_3 C\theta_2 + a_2 \\ S(\theta_2 + \theta_4) & C(\theta_2 + \theta_4) & 0 & a_3 S\theta_2 \\ 0 & 0 & 1 & -d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\begin{aligned} U_0 &= {}^0T_1 U_1 = {}^0T_1 {}^1T_2 {}^2T_3 {}^3T_4 = {}^0T_4 \\ &= \begin{bmatrix} C(\theta_1 + \theta_2 + \theta_4) & -S(\theta_1 + \theta_2 + \theta_4) & 0 & a_3 C(\theta_1 + \theta_2) + a_2 C(\theta_1) \\ S(\theta_1 + \theta_2 + \theta_4) & C(\theta_1 + \theta_2 + \theta_4) & 0 & a_3 S(\theta_1 + \theta_2) + a_2 S(\theta_1) \\ 0 & 0 & 1 & -d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{aligned}$$

# Modèle Géométrique Direct : MGD

## Exemple Robot SCARA - Python

```
from math import * # module needed to use cosinus, sinus ... functions
import numpy as np # Python doesn't have a built-in type for matrices. However NumPy is a package for scientific computing
from sympy import * # symbolic computation
```

```
def T(ai, alpha, di, theta):
    return np.array([[cos(theta), -sin(theta), 0, ai],
                    [sin(theta)*cos(alpha), cos(theta)*cos(alpha), -sin(alpha), -sin(alpha)*di],
                    [sin(theta)*sin(alpha), cos(theta)*sin(alpha), cos(alpha), cos(alpha)*di],
                    [0, 0, 0, 1]])
```

# Modèle Géométrique Direct : MGD

## Exemple Robot SCARA - Python

```
a1 = 0          # a i-1
alpha1 = 0      # alpha i-1
d1 = 0          # d i
t1 = symbols('t1') # theta i

a2 = symbols('a2')
alpha2 = 0
d2 = 0
t2 = symbols('t2')

a3 = symbols('a3')
alpha3 = 0
d3 = symbols('d2')
t3 = 0

a4 = 0
alpha4 = 0
d4 = 0
t4 = symbols('t4')

T01 = T(a1, alpha1, d1, t1)
T12 = T(a2, alpha2, d2, t2)
T23 = T(a3, alpha3, d3, t3)
T34 = T(a4, alpha4, d4, t4)

U4 = np.identity(4)    # I4
U3 = T34 @ U4
U2 = T23 @ U3
U1 = T12 @ U2
U0 = T01 @ U1

print(simplify(U0 @ np.array([0,0,0,1])).subs({t1:0, t2:0, t4:0}))
```

...: print(simplify(U0 @ np.array([0,0,0,1])).subs({t1:0, t2:0, t4:0})) #subs permet d'assigner une valeur numérique à une variable symbolique  
[1.0\*a2 + 1.0\*a3, 0, -1.0\*d2, 1.0]

En position  $q_1 = 0, q_2 = 0, q_4 = 0$

$X = a_2 + a_3$

$Y = 0$

$Z = -d_2$

# Modèle Géométrique Direct : MGD

## Exemple Robot SCARA

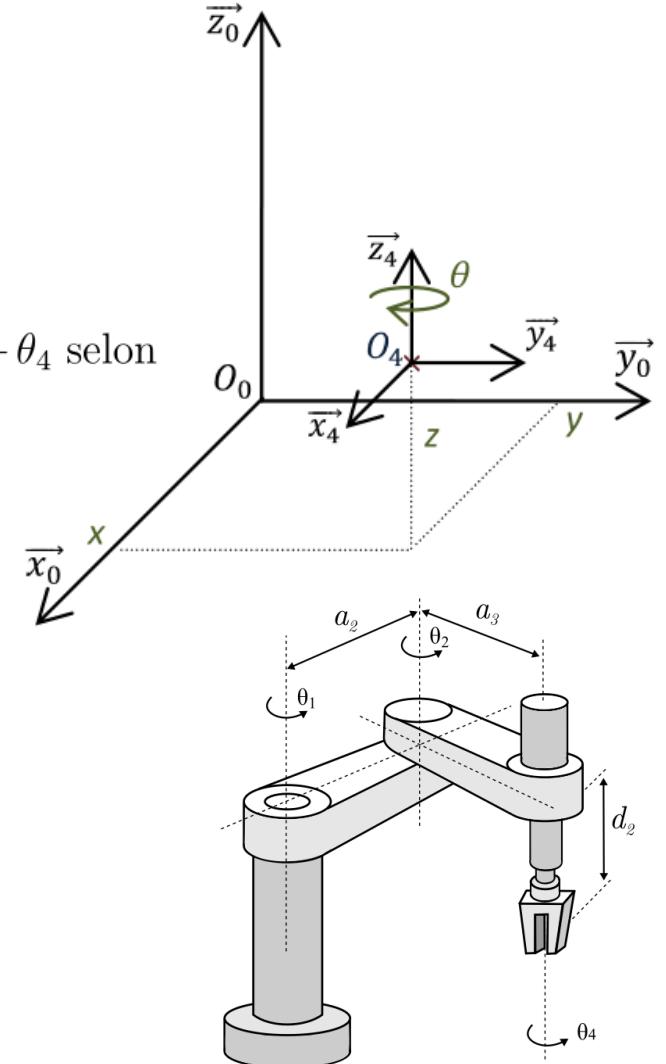
$$U_0 = \begin{bmatrix} C(\theta_1 + \theta_2 + \theta_4) & -S(\theta_1 + \theta_2 + \theta_4) & 0 & a_3C(\theta_1 + \theta_2) + a_2C(\theta_1) \\ S(\theta_1 + \theta_2 + \theta_4) & C(\theta_1 + \theta_2 + \theta_4) & 0 & a_3S(\theta_1 + \theta_2) + a_2S(\theta_1) \\ 0 & 0 & 1 & -d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Toutes les rotations se font selon l'axe z. L'angle d'orientation  $\theta_z$  vaut alors :  $\theta = \theta_1 + \theta_2 + \theta_4$  selon la direction  $z_0$ .

$$T_{04} = \begin{pmatrix} \cos(\theta) & -\sin(\theta) & 0 & x \\ \sin(\theta) & \cos(\theta) & 0 & y \\ 0 & 0 & 1 & z \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Donc MGD :

$$\begin{cases} x = a_2 \cdot \cos(\theta_1) + a_3 \cdot \cos(\theta_1 + \theta_2) \\ y = a_2 \cdot \sin(\theta_1) + a_3 \cdot \sin(\theta_1 + \theta_2) \\ z = -d_2 \\ \theta = \theta_1 + \theta_2 + \theta_4 \end{cases}$$



## MGD robot SCARA – Vérification géométrique

On peut vérifier aisément (car c'est un cas simple) géométriquement à l'aide de la vue de dessus :

$$O_oA =$$

$$AB =$$

$$O_oB =$$

$$O_oC =$$

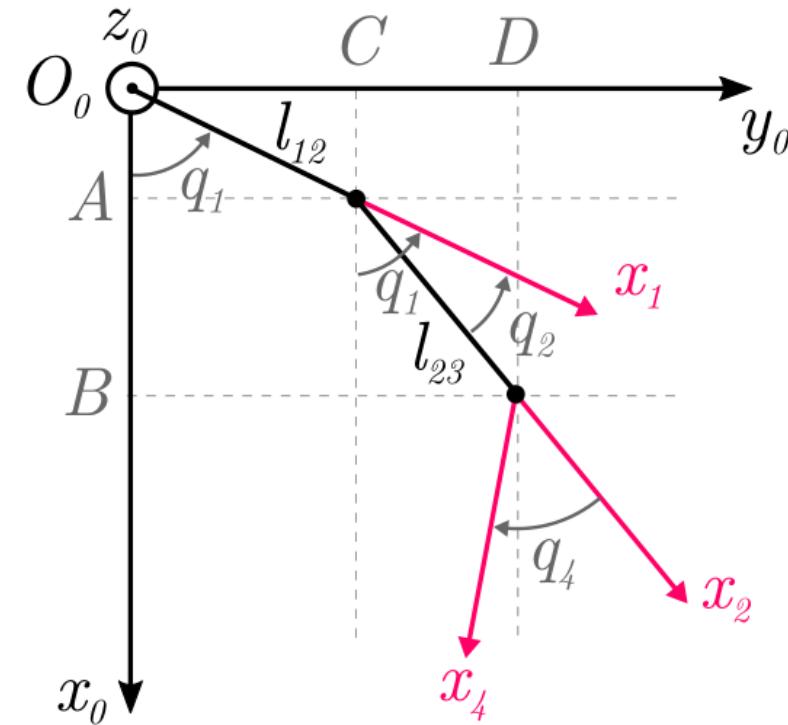
$$CD =$$

$$O_oD =$$

$$z =$$

$$\theta =$$

avec  $q_2$  et  $q_4 < 0$ .



## MGD robot robot Stäubli RX-90

A partir du tableau des paramètres DHKK on peut calculer les matrices de transfert :

$${}^0T_1 = \begin{bmatrix} C\theta_1 & -S\theta_1 & 0 & 0 \\ S\theta_1 & C\theta_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^1T_2 = \begin{bmatrix} C\theta_2 & -S\theta_2 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ S\theta_2 & C\theta_2 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^2T_3 = \begin{bmatrix} & & & \\ & & & \\ & & & \\ & & & \end{bmatrix}$$

$${}^3T_4 = \begin{bmatrix} & & & \\ & & & \\ & & & \\ & & & \end{bmatrix} \quad {}^4T_5 = \begin{bmatrix} C\theta_5 & -S\theta_5 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ S\theta_5 & C\theta_5 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^5T_6 = \begin{bmatrix} C\theta_6 & -S\theta_6 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ S\theta_6 & C\theta_6 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

## MGD robot robot Stäubli RX-90

$$U_6 = I_6$$

$$U_5 = {}^5T_6$$

$$U_4 = {}^4T_6 = {}^4T_5 U_5 = \begin{bmatrix} C\theta_5 C\theta_6 & -C\theta_5 S\theta_6 & -S\theta_5 & 0 \\ S\theta_6 & C\theta_6 & 0 & 0 \\ S\theta_5 S\theta_6 & -S\theta_5 S\theta_6 & C\theta_5 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$U_3 = {}^3T_6 = {}^3T_4 U_4 = \begin{bmatrix} C\theta_4 C\theta_5 S\theta_4 C\theta_6 & -C\theta_4 C\theta_5 S\theta_6 & -C\theta_4 S\theta_5 & 0 \\ S\theta_5 C\theta_6 & -S\theta_5 S\theta_6 & C\theta_5 & d_3 \\ -S\theta_4 S\theta_5 S\theta_6 - C\theta_4 S\theta_6 & S\theta_4 C\theta_5 S\theta_6 & S\theta_4 S\theta_5 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

:

$$U_0 = {}^0T_6 = {}^0T_1 U_1 = \begin{bmatrix} s_x & n_x & a_x & P_x \\ s_y & n_y & a_y & P_y \\ s_z & n_z & a_z & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

## MGD robot robot Stäubli RX-90

$$U_0 = {}^0T_6 = {}^0T_1 U_1 = \begin{bmatrix} s_x & n_x & a_x & P_x \\ s_y & n_y & a_y & P_y \\ s_z & n_z & a_z & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$S_x =$$

$$S_y =$$

$$S_z =$$

$$n_x =$$

$$n_y =$$

$$n_z =$$

$$a_x =$$

$$a_y =$$

$$a_z =$$

$$P_x =$$

$$P_y =$$

$$P_z =$$

## Poignet sphérique

Énormément de robots possèdent un poignet **sphérique**. C'est à dire que les axes des trois dernières articulations s'intersectent en un point.

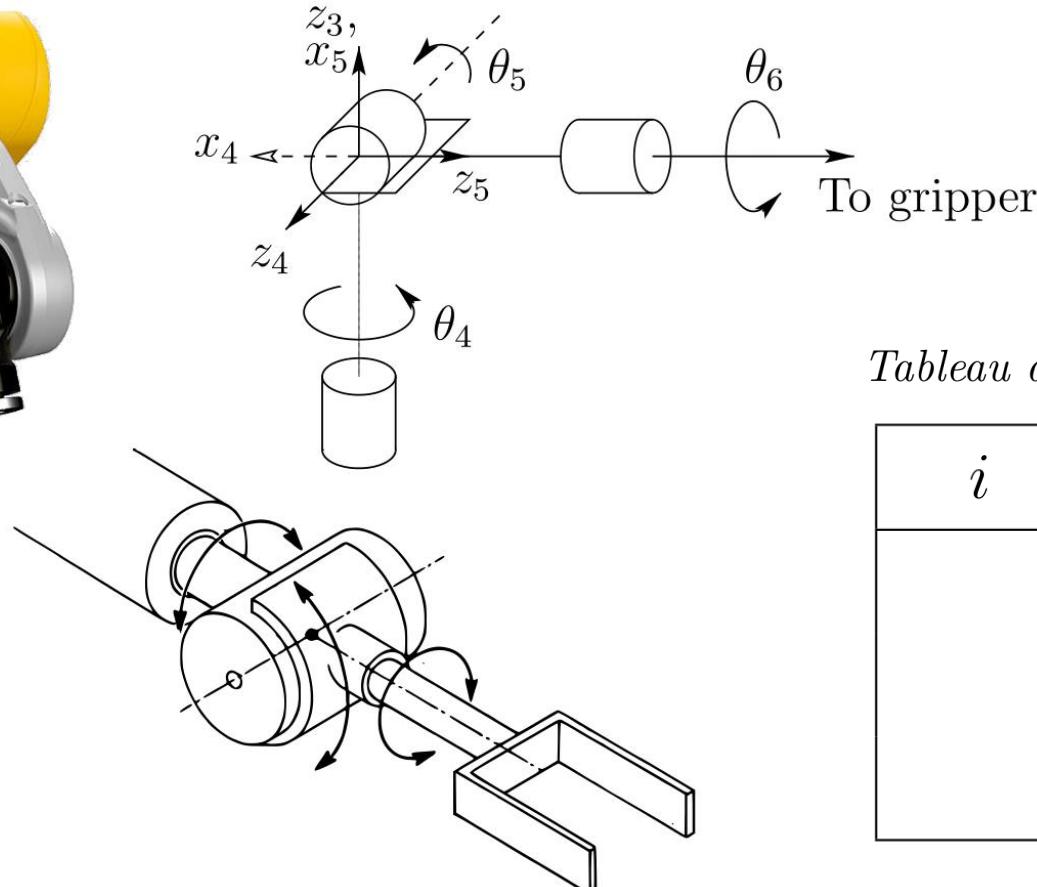
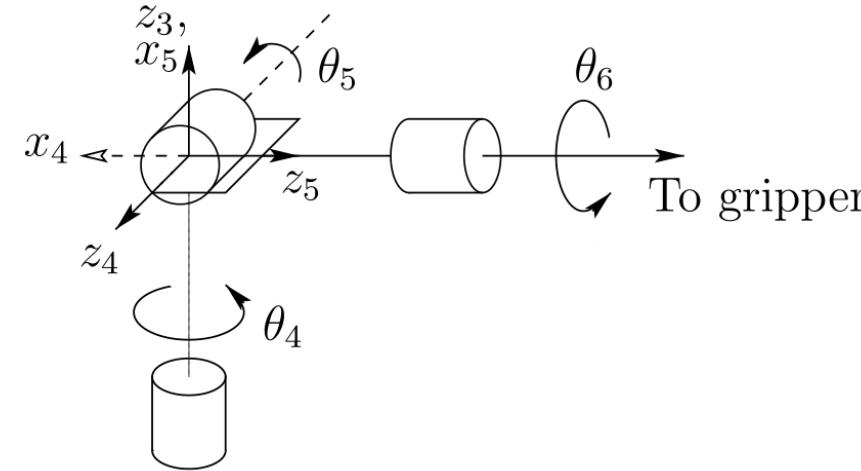
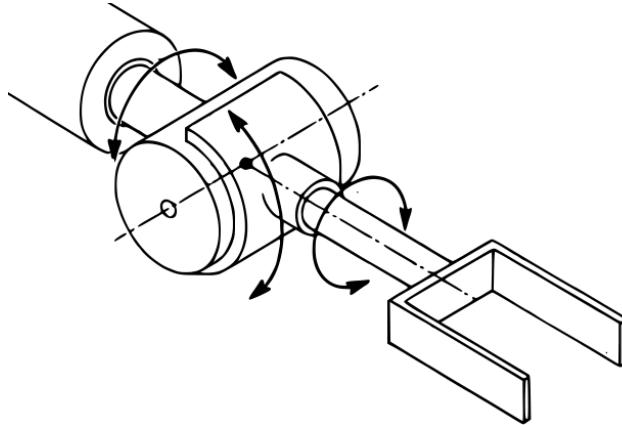


Tableau des paramètres DHKK - Poignet sphérique

$i$	$a_{i-1}$	$\alpha_{i-1}$	$d_i$	$\theta_i$

## Poignet sphérique



$${}^3T_4 = \begin{bmatrix} c4 & 0 & -s4 & 0 \\ s4 & 0 & c4 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^4T_5 = \begin{bmatrix} c5 & 0 & s5 & 0 \\ s5 & 0 & -c5 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^5T_6 = \begin{bmatrix} c6 & -s6 & 0 & 0 \\ s6 & c6 & 0 & 0 \\ 0 & 0 & 1 & d_6 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^3T_6 = {}^3T_4 {}^4T_5 {}^5T_6 = \begin{bmatrix} c4c5c6 - s4s6 & -c4c5s6 - s4c6 & c4s5 & c4s5d_6 \\ s4c5c6 + c4s6 & -s4c5s6 + c4c6 & s4s5 & s4s5d_6 \\ -s5c6 & s5s6 & c5 & c5d_6 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$