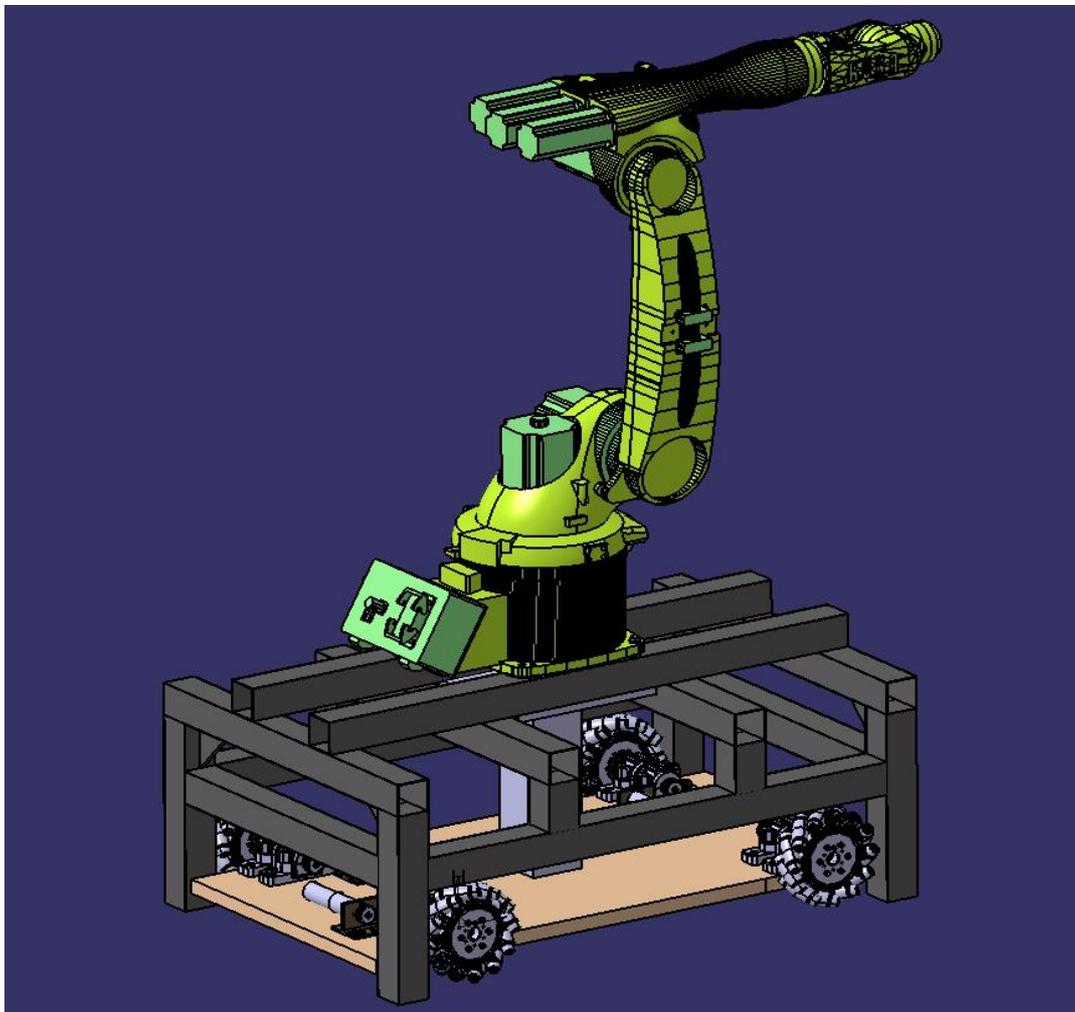


Projet IMA 4 : Commande d'un robot mobile holonome et manipulateur



Etudiant :
Brice Tcheussi /Stella Loembet

Encadrant :
Rochdi Merzouki

Table des matières

Introduction.....	3
Présentation du matériel.....	4
Modélisation du moteur brushless.....	6
Modélisation sur MATLAB-Simulink.....	7
Modélisation mécanique.....	13
Modélisation du châssis.....	13
Modèle géométrique du châssis.....	14
Modèle cinématique du châssis.....	15
Modèle dynamique du châssis.....	16
Modélisation du Kuka Kr5.....	18
Modèle géométrique direct.....	18
Modèle géométrique inverse.....	20
Modèle cinématique directe.....	20
Modèle cinématique inverse.....	22
Moment cinématique:.....	22
Modèle dynamique.....	23
Moment dynamique :.....	23
Calcul de l'énergie cinétique :.....	23
Simulation 3D.....	24
CONCLUSION.....	25

Introduction

La modélisation d'un système complexe représente l'une des phases essentielles de sa conception. En effet un modèle permet d'obtenir une représentation du système réel afin de mieux le contrôler et d'anticiper son comportement. C'est ainsi que dans le cadre du projet du semestre 8, nous avons choisis la commande du robot holonome et de son manipulateur. Notre mission a été de modéliser le moteur du châssis, mais nous devons aussi modéliser le bras manipulateur kuka kr5 et réaliser un modèle 3D montrant les différents déplacements et mouvements du châssis et du bras. Il sera donc question de présenter dans ce document les différentes étapes que nous avons suivies pour la modélisation tant sur le plan électrique (moteur et contrôleur) que mécanique (châssis et bras manipulateur). Nous présenterons également la validation de nos modèles sur Matlab simulink pour la partie électrique et sur simulateur 3D réalisé grâce à la librairie Opengl.

Présentation du matériel

I- Le robot kuka KR5 :

Le Kuka KR5 est souvent utilisé pour des tâches de soudage grâce à son anatomie qui permet de faire des travaux de précision. Ce robot se classe dans la catégorie des robots à charge faible. En effet il porter des charges de moins de cinq kilogrammes. Cependant, il est compact et accessible, on peut le fixer au sol ou au plafond. Nous avons identifié les différents corps du robot et leurs rotations respectives. Nous avons cinq corps qui font tous une rotation d'amplitude différente. Tous les corps de ce robot peuvent être assimilés à des cylindres. Le Kuka Kr 5 possède cinq degrés de liberté.



Figure 1: Kuka Kr5

II-Le châssis :

Le châssis est un robot holonome à trois degrés de liberté. Il portera le robot Kuka pour lui permettre de se déplacer plus aisément. Son contrôleur a été programmé, dans le cadre des projets IMA4 par un autre étudiant, de telle sorte que le robot soit commandable sans fils, grâce à une télécommande Wii, et de manière filaire grâce à un joystick.

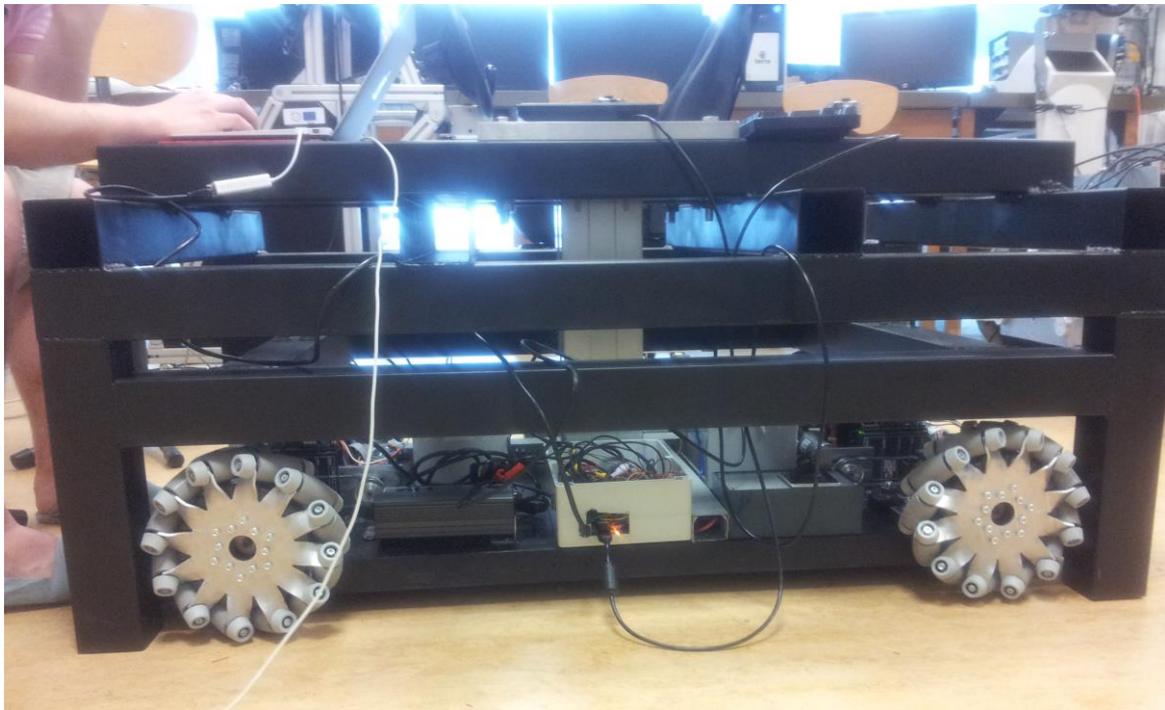


Figure 2:Châssis omnidirectionnel

Modélisation du moteur brushless

Une machine brushless est à la base une machine synchrone à aimants permanents et sans balais alimentée par une source de tension continue, elle nécessite donc un onduleur permettant la conversion de la tension continue en tension triphasé pour l'alimentation des phases statoriques. Elle est aussi conçue de façon à ce que les forces électromotrices (fem) induites au stator soient trapézoïdales. Pour obtenir un couple maximale, il suffit donc d'alimenter les phases pour lesquelles les forces électromotrices sont maximales ce qui conduit à une alimentation en créneaux des phases. D'où la nécessité d'adjoindre a la machine des capteurs de positions dans notre cas des capteurs hall (H1, H2 et H3) permettant de détecter la position de l'aimant nécessaire à l'aiguillage du courant dans les « bonnes » phases comme le montre la figure suivante :

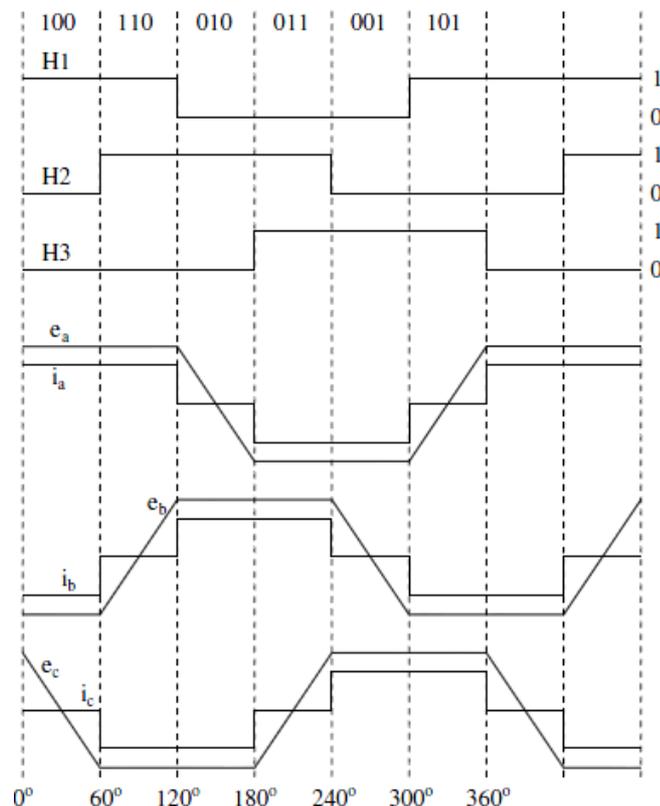


Figure 3: forces électromotrices et courants dans les phases

Modélisation sur MATLAB-Simulink

Les moteurs utilisés pour la traction des roues sont des moteurs brushless fabriqués par Maxon (ECmax-40). Ces moteurs sont couplés à des contrôleurs qui réalisent un asservissement en vitesse. Normalement la validation d'un modèle se fait en boucle ouverte mais puisque le moteur était couplé au contrôleur il nous a été recommandé de modéliser l'ensemble du système moteur+contrôleur. Nous avons réalisé la modélisation de l'ensemble sur MATLAB-Simulink en rentrant les paramètres du moteur obtenu de la fiche technique.

1	Assigned power rating	W	120
2	Nominal voltage	Volt	48.0
3	No load speed ¹⁾	rpm	10100
4	Stall torque ¹⁾	mNm	2090
5	Speed / torque gradient ¹⁾	rpm / mNm	4.89
6	No load current ¹⁾	mA	348
7	Terminal resistance phase to phase	Ohm	1.03
8	Max. permissible speed	rpm	12000
9	Max. continuous current at 5000 rpm ¹⁾	mA	4450
10	Max. continuous torque at 5000 rpm	mNm	168
11	Max. efficiency ¹⁾	%	84
12	Torque constant	mNm / A	44.8
13	Speed constant	rpm / V	213
14	Mechanical time constant	ms	5.2
15	Rotor inertia	gcm ²	101
16	Terminal inductance phase to phase	mH	0.204
17	Thermal resistance housing-ambient	K / W	3.5
18	Thermal resistance winding-housing	K / W	0.3
19	Thermal time constant winding	s	3.9
20	Thermal time constant stator	s	1140

Figure 4: Paramètres moteur brushless

Il a donc fallu représenter les différents éléments de la machine : capteur Hall, machine proprement dite, l'onduleur, la source d'énergie continue. Nous avons réalisé le modèle en boucle ouverte suivant sur Matlab :

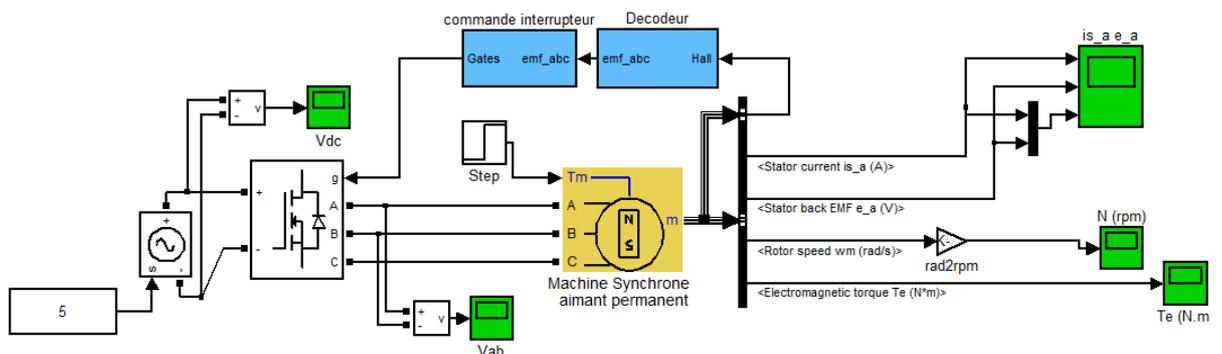


Figure 5: model brushless du moteur sans PI

Ce modèle ne pouvant être validé seul dû au couplage de la machine réelle au contrôleur, nous lui ajoutons donc un contrôleur PI avec une consigne de vitesse de 4800tr/m, valeur que nous avons utilisé lors des essais. L'idée est de considérer l'ensemble PI+Machine brushless comme étant le système à valider. Le réglage du PI nous a donné le gain proportionnel, $K_p=0.00007$ et le gain intégral $K_I=0.4$.

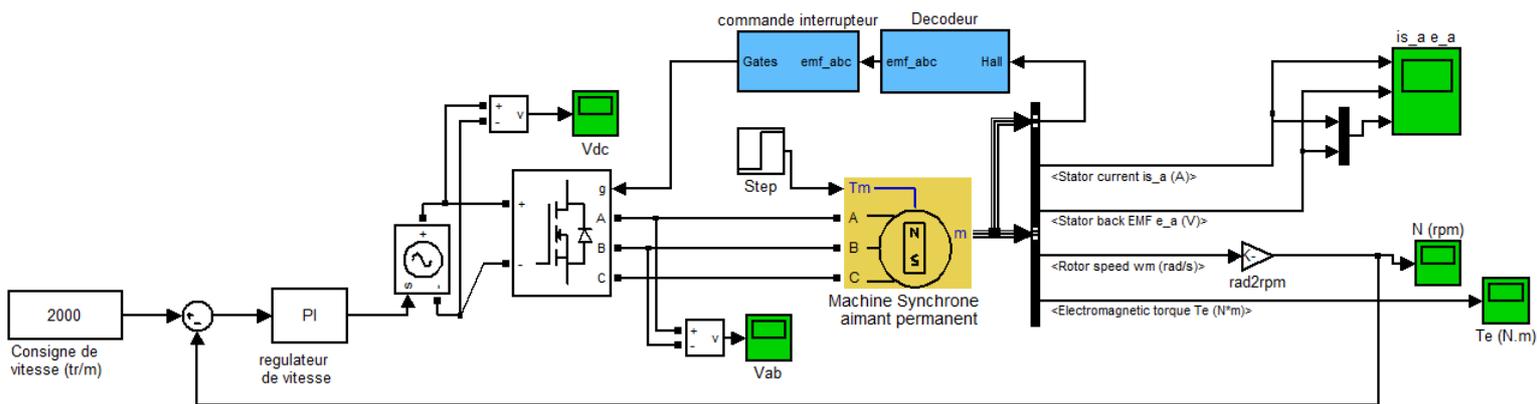


Figure 6: Modèle du moteur avec PI

Afin de valider ce modèle de simulation nous avons prélevé des grandeurs caractéristiques telles que le couple électromagnétique, la vitesse, le courant et la force électromotrice des phases statorique. Elles nous ont permis d'analyser le fonctionnement du système et de les comparer à celles du système réel. Les courants et forces électromotrices dans les phases du modèle sont des courbes en créneaux et trapézoïdales respectivement et caractérisent une machine brushless. Ainsi pour la phase A nous avons :



Figure 7: force électromotrice trapézoïdale et courant en craineau dans la phase A

En simulant le système sur Matlab-Simulink nous obtenons la courbe de vitesse suivante:

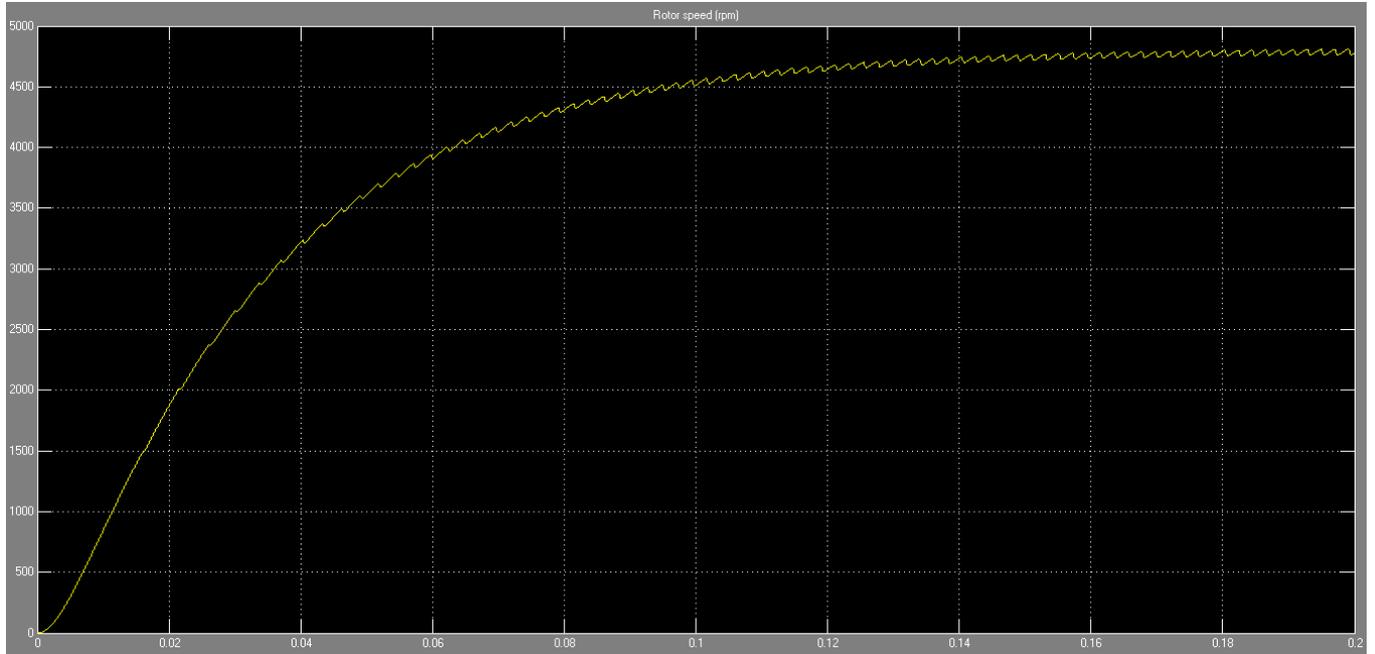


Figure 8: Courbe de vitesse du modèle

A partir du logiciel fourni par le fabricant on a aussi prélevé la courbe de vitesse du moteur.

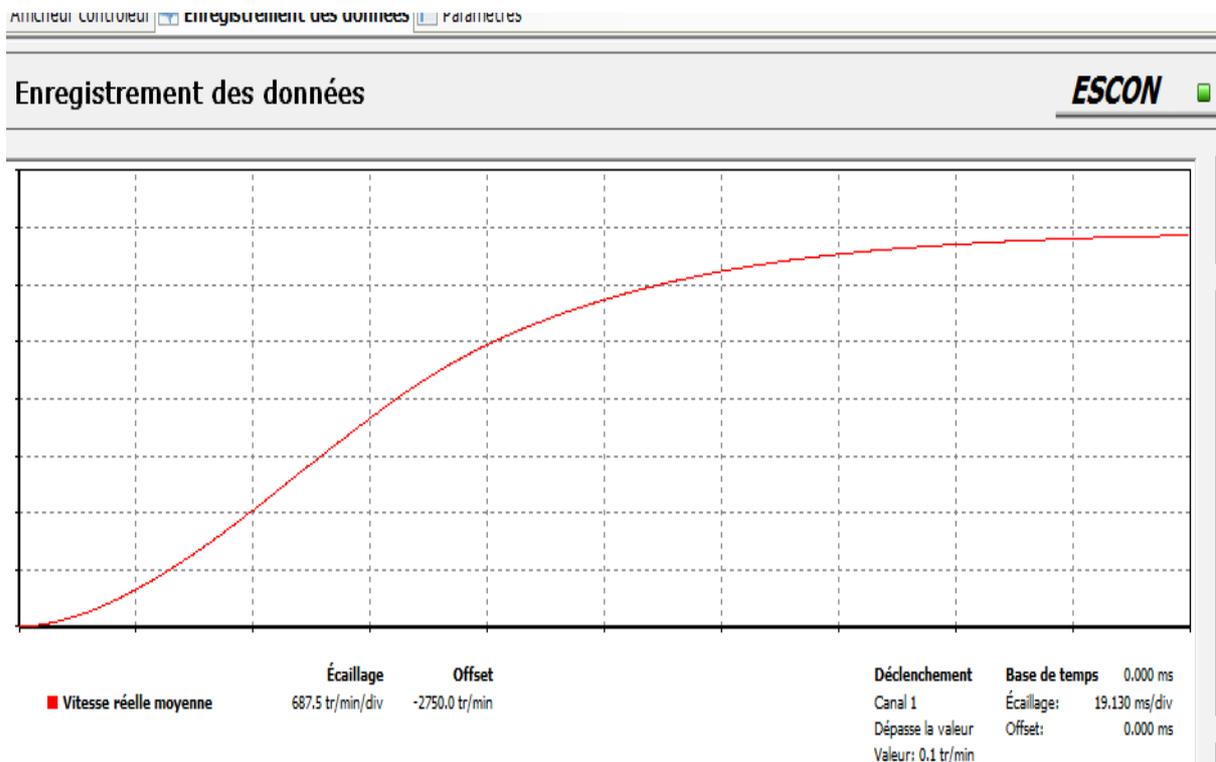


Figure 9: courbe de vitesse du système réelle moteur+contrôleur

Le modèle en boucle fermé a une constante de temps de 0.06s et un temps de réponse à 95% de 0.1s comme celui obtenu du système réelle.

Nous avons aussi pu vérifier les fonctions de régulation et asservissement de cette commande en y ajoutant une perturbation matérialisée par une augmentation du couple résistant de 0.5Nm à l'instant $t=0.15s$. Le correcteur nous permet de revenir à la vitesse de consigne.



Figure 10: Réjection de perturbation

Nous constatons que le courant dans les phases et par conséquent le couple électromagnétique augmente de façon à maintenir la vitesse constante comme nous le montre les courbes suivantes :

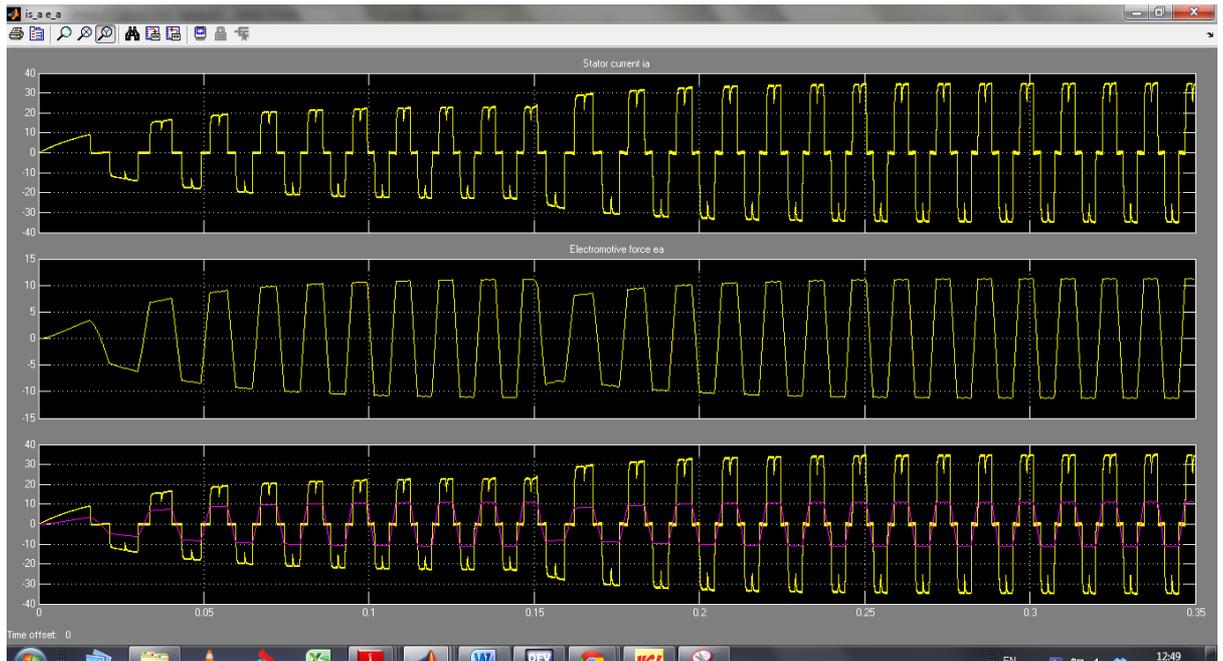


Figure 11: Augmentation du courant et couple due à une augmentation du couple résistant

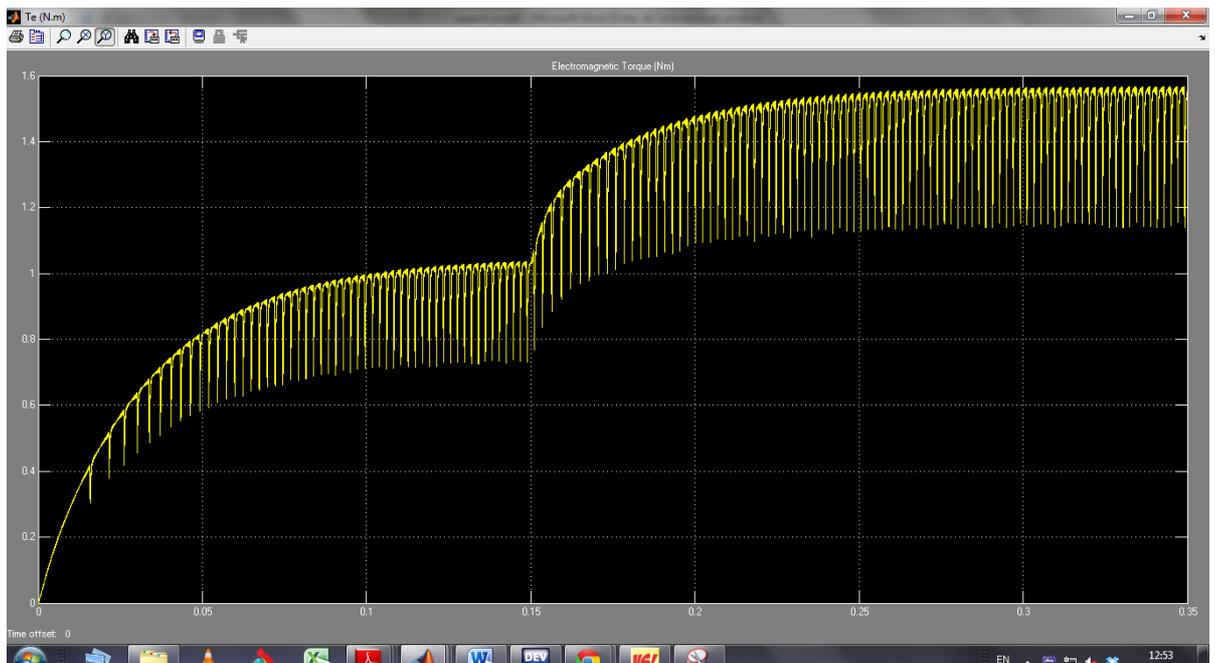


Figure 12: Augmentation du couple électromagnétique suite à une augmentation du couple résistant

Il est néanmoins important de mentionner que la validation absolue d'un modèle se fait en boucle ouverte et aurait nécessité, dans notre cas, une connaissance des paramètres du correcteur utilisé par le fabricant ce qui ne nous a pas été donné par ce dernier. Il est également important de souligner que la mise en équation du système permet de lui appliquer des algorithmes de

commande avancés mais le fait que le moteur soit déjà couplé à un correcteur rend cette modélisation presque inutile et non-économique.

Modélisation mécanique

Modélisation du châssis

Le caractère omnidirectionnel du châssis et par conséquent du robot est assuré par les roues mecanum montées sur sa plateforme. Sur la circonférence des roues se trouvent des galets inclinés et pouvant pivoter autour de leur axe respectif.



Figure 13: Roue mecanum

La configuration de roues permet au châssis de faire une rotation autour de lui-même ainsi que des translations latérale et longitudinale comme nous le montre les figures suivantes :

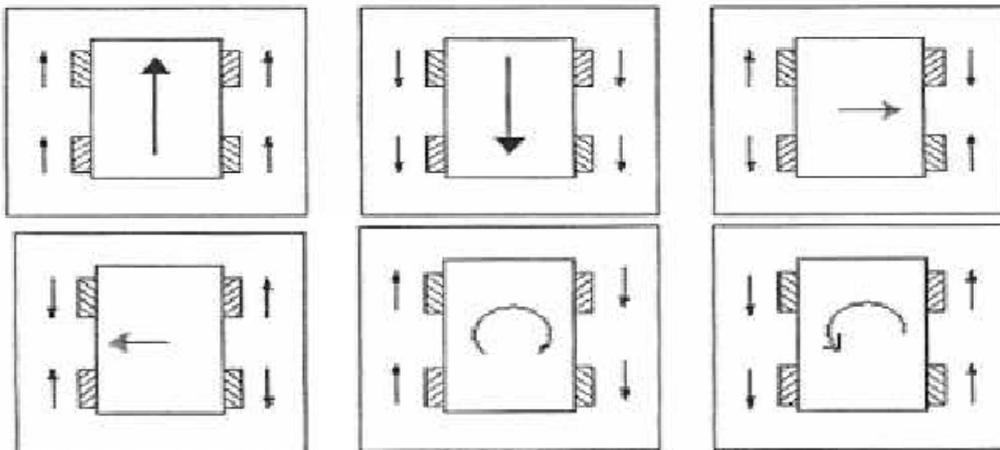


Figure 14: Déplacements du châssis

Modèle géométrique du châssis

On considère la figure suivante :

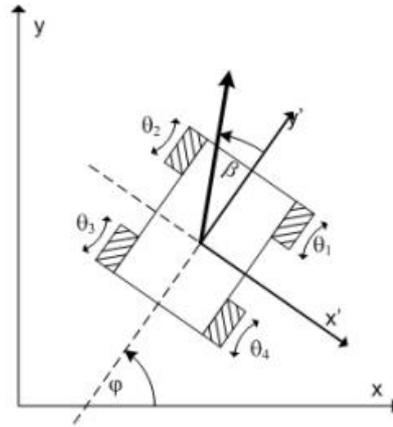


Figure 15 : plateforme du châssis et ses coordonnées

La vitesse développée à chaque roue dans le repère R' (X', Y', Z') lié au châssis

$$\mathbf{v}_i = \text{SIG}(\theta_i) K_i r \dot{\theta}_i \sin \alpha$$

s'écrit :

Avec ;

$\text{SIG}(\theta_i)$ le sens de rotation de la roue (trigo = -, antitrig = +)

K_i la constante de la roue,

r la constante de la roue

α l'inclinaison angulaire de la roue et

$\dot{\theta}$ est la vitesse angulaire de la roue

Nous modélisons ici les déplacements du châssis, soit les translations longitudinale et latérale et la rotation. Le but est de déterminer la position (x',y') du châssis à partir du déplacement angulaires des roues. Pour cela nous allons nous placer dans les différentes configurations de déplacement du châssis accessibles dans le cadre de ce projet .

- Translation longitudinale du châssis

La translation longitudinale du châssis est obtenue lorsque les quatre roues tournent dans le même sens. De plus, les vitesses angulaires des roues sont les mêmes ($\dot{\theta}_1 = \dot{\theta}_2 = \dot{\theta}_3 = \dot{\theta}_4$). En faisant une projection de ces vitesses dans le repère (X',Y') du châssis on obtient que les composantes suivant X' s'annulent et le châssis se déplace suivant Y' à la vitesse:

$$v_{y'} = \pm K_i r \dot{\theta}_i (\sin \alpha)^2$$

D'où une translation longitudinale dans le repère (X', Y') de :

$y' = + K_i r \theta (\sin \alpha)^2$; mouvement avant : les roues tournent vers l'avant

$y' = - K_i r \theta (\sin \alpha)^2$; mouvement arrière : les roues tournent vers l'arrière

- Translation latérale du châssis

Ce déplacement est obtenu lorsque les roues diagonalement opposées tournent dans le même sens et dans le sens contraire des deux autres roues. En faisant une projection de ces vitesses dans le repère (X',Y') du châssis on obtient que les composantes suivant Y' s'annulent et le châssis se déplace suivant X' à la vitesse :

$$v_{x'} = \pm K_i r \dot{\theta}_i \sin \alpha \cos \alpha$$

D'où une translation latéral dans le repère (X', Y') de :

$$\begin{aligned} x' &= + K_i r \theta \sin \alpha \cos \alpha ; \text{ mouvement gauche : } \theta_1 > 0, \theta_2 < 0 \\ x' &= - K_i r \theta \sin \alpha \cos \alpha ; \text{ mouvement droite : } \theta_2 > 0 \text{ et } \theta_1 < 0 \end{aligned}$$

- Rotation du châssis

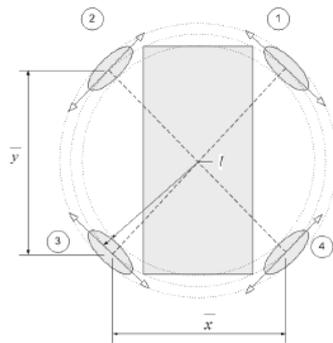


Figure 16: mouvement de rotation

Ce mouvement est obtenu lorsque les roues droites et gauches tournent en sens opposé. Il est important de connaître le rayon de braquage. Le mouvement de rotation par rapport au centre de masse est réalisé lorsque toutes les roues ont des moments dans le même sens par rapport au centre de masse. Ainsi on peut écrire :

$$\begin{aligned} \dot{\varphi} &= \frac{K_i r \dot{\theta} \sin \alpha}{l} \text{ avec } l = \frac{\sqrt{x^2 + y^2}}{2} \\ \Rightarrow \varphi &= \frac{K_i r \theta \sin \alpha}{l} \end{aligned}$$

Modèle cinématique du châssis

La vitesse de chacune des roues dans la direction d'orientation des galets s'écrit :

$$v_i = \text{SIG}(\theta_i) K_i r \dot{\theta}_i \sin \alpha$$

Cette vitesse est dans la même direction que l'orientation des galets. Ainsi on trouve les composantes de cette vitesse dans le repère (X', Y', Z') lié au châssis comme suit :

-suivant X'

$$\begin{aligned}\sum_{i=1}^4 \mathbf{v}_{ix'} &= \sum_{i=1}^4 \mathbf{v}_i \cos \alpha \\ &= \sum_{i=1}^4 (-1)^i \text{SIG}(\theta_i) K_i r \dot{\theta}_i \sin \alpha \cos \alpha\end{aligned}$$

-suivant Y'

$$\begin{aligned}\sum_{i=1}^4 \mathbf{v}_{iy'} &= \sum_{i=1}^4 \mathbf{v}_i \sin \alpha \\ &= \sum_{i=1}^4 (-1)^i \text{SIG}(\theta_i) K_i r \dot{\theta}_i (\sin \alpha)^2\end{aligned}$$

On déduit donc la vitesse résultante :

$$\mathbf{v} = \sum_{i=1}^4 v_{ix'} \mathbf{i}' + \sum_{i=1}^4 v_{iy'} \mathbf{j}'$$

En revenant dans le repère Galiléen, on obtient :

$$\mathbf{v}_x = \sin \varphi \sum_{i=1}^4 \mathbf{v}_{ix'} + \cos \varphi \sum_{i=1}^4 \mathbf{v}_{iy'}$$

$$\mathbf{v}_y = \cos \varphi \sum_{i=1}^4 \mathbf{v}_{ix'} + \sin \varphi \sum_{i=1}^4 \mathbf{v}_{iy'}$$

La direction du déplacement dans le repère Galiléen est alors :

$$\beta = \arctan \left(\frac{\mathbf{v}_y}{\mathbf{v}_x} \right)$$

Modèle dynamique du châssis

On considère la figure suivante qui montre les forces qui agissent sur un galet en contact avec le sol:

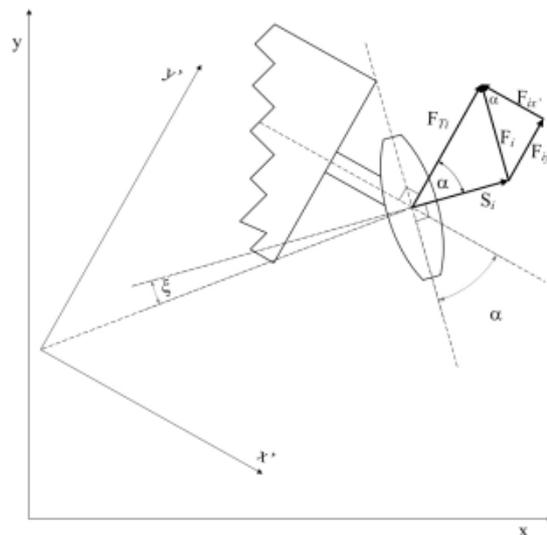


Figure 17: forces s'exerçant sur un galet en contact avec le sol

Soit F_{Ti} la force développée à la circonférence de la roue i , de rayon r due au couple moteur T_i ,

$$T_i = r \mathbf{F}_{T_i}$$

Cette force se décompose en une composante effective, \mathbf{F}_i et ineffective, \mathbf{S}_i .

$$\mathbf{F}_i = \mathbf{F}_{T_i} \sin \alpha, \quad \mathbf{S}_i = \mathbf{F}_{T_i} \cos \alpha$$

La force effective est celle qui provoque le mouvement, et s'exprime dans le repère (X',Y') par :

$$\mathbf{F}_i = \mathbf{F}_{ix'} \mathbf{i}' + \mathbf{F}_{iy'} \mathbf{j}'$$

La force résultante issue des quatre roues motrices dans la direction X' est

$$\begin{aligned} \sum_{i=1}^4 \mathbf{F}_{ix'} &= \sum_{i=1}^4 \mathbf{F}_i \cos \alpha \\ &= \sum_{i=1}^4 (-1)^i (\text{SIG}(\theta_i)) K_i \mathbf{F}_{T_i} \sin \alpha \cos \alpha \end{aligned}$$

Et suivant Y' :

$$\sum_{i=1}^4 \mathbf{F}_{iy'} = \sum_{i=1}^4 \mathbf{F}_i \sin \alpha = \sum_{i=1}^4 (\text{SIG}(\theta_i)) K_i \mathbf{F}_{T_i} (\sin \alpha)^2$$

Dans le repère galiléen la seconde loi de Newton peut alors s'écrire :

$$\begin{aligned} m\ddot{x} &= \sum_{i=1}^4 \left\{ \begin{array}{l} ((-1)^i (\text{SIG}(\theta_i)) K_i \mathbf{F}_{T_i} \sin \alpha \cos \alpha \sin \varphi) \\ + ((\text{SIG}(\theta_i)) K_i \mathbf{F}_{T_i} (\sin \alpha)^2 \cos \varphi) \end{array} \right\} \\ m\ddot{y} &= \sum_{i=1}^4 \left\{ \begin{array}{l} ((-1)^i (\text{SIG}(\theta_i)) K_i \mathbf{F}_{T_i} \sin \alpha \cos \alpha \cos \varphi) \\ + ((\text{SIG}(\theta_i)) K_i \mathbf{F}_{T_i} (\sin \alpha)^2 \sin \varphi) \end{array} \right\} \end{aligned}$$

Le couple, T , développé pour changer l'orientation du châssis est alors donné par :

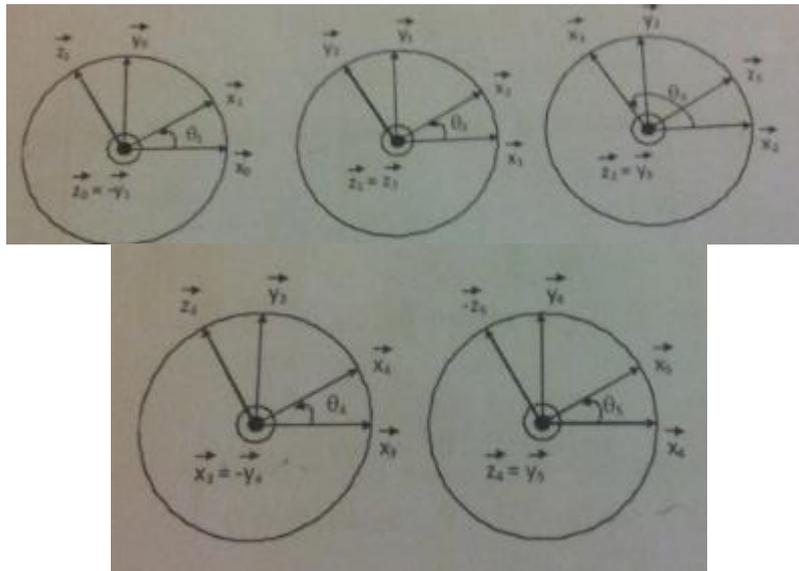
$$T = I \ddot{\varphi}$$

avec $I =$ inertie de masse du châssis et $\ddot{\varphi} =$ l'accélération angulaire

Modélisation du Kuka Kr5

Modèle géométrique direct

Pour faire le modèle il est nécessaire d'identifier les corps les en mouvement et définir leurs repères. Ceci nous permet d'obtenir nous avons établi les paramètres de Denavit Hartenberg. A partir des différents degrés de liberté du robot nous avons pu déterminés les projections des repères ci-dessous.



A l'aide des différentes projections des repères du robot Kuka, nous avons pu calculer ces paramètres.

Liaisons	Alpha	Théta	D (mm)	A (mm)
1	$-\pi/2$	θ_1	-400	180
2	0	θ_2	0	-600
3	$\pi/2$	θ_3	0	110
4	$-\pi/2$	θ_4	-620	0
5	$\pi/2$	θ_5	0	0
6	0	θ_6	0	0

Ces paramètres nous servent pour le calcul des matrices de passage, qui nous permette de passage d'un repère à un autre. Une fois les paramètres calculés, nous pouvons calculer les matrices de passages, à l'aide de l'outil de calcul Matlab. Voici le code et les résultats.

```

%Matrice T_0_1
T01=[cos(t1), -cosd(alpha1)*sin(t1), sind(alpha1)*sin(t1), a1*cos(t1);
      sin(t1),  cosd(alpha1)*cos(t1), -cos(t1)*sind(alpha1), a1*sin(t1);
      0,      sind(alpha1),      cosd(alpha1),      d1;
      0,      0,      0,      1 ]

%Matrice T_1_2
T12=[cos(t2), -cosd(alpha2)*sin(t2), sind(alpha2)*sin(t2), a2*cos(t2);
      sin(t2),  cosd(alpha2)*cos(t2), -cos(t2)*sind(alpha2), a2*sin(t2);
      0,      sind(alpha2),      cosd(alpha2),      d2;
      0,      0,      0,      1 ]

%Matrice T_2_3
T3 = t3-pi/2;
T23=[cos(T3), -cosd(alpha3)*sin(T3), sind(alpha3)*sin(T3), a3*cos(T3);
      sin(T3),  cosd(alpha3)*cos(T3), -cos(T3)*sind(alpha3), a3*sin(T3);
      0,      sind(alpha3),      cosd(alpha3),      d3;
      0,      0,      0,      1 ]

%Matrice T_3_4
T34=[cos(t4), -cosd(alpha4)*sin(t4), sind(alpha4)*sin(t4), a4*cos(t4);
      sin(t4),  cosd(alpha4)*cos(t4), -cos(t4)*sind(alpha4), a4*sin(t4);
      0,      sind(alpha4),      cosd(alpha4),      d4;
      0,      0,      0,      1 ]

%Matrice T_4_5
T45=[cos(t5), -cosd(alpha5)*sin(t5), sind(alpha5)*sin(t5), a5*cos(t5);
      sin(t5),  cosd(alpha5)*cos(t5), -cos(t5)*sind(alpha5), a5*sin(t5);
      0,      sind(alpha5),      cosd(alpha5),      d5;
      0,      0,      0,      1 ]

%Matrice T_5_6
T56=[cos(t6), -cosd(alpha6)*sin(t6), sind(alpha6)*sin(t6), a6*cos(t6);
      sin(t6),  cosd(alpha6)*cos(t6), -cos(t6)*sind(alpha6), a6*sin(t6);
      0,      sind(alpha6),      cosd(alpha6),      d6;
      0,      0,      0,      ,1 ]

```

```

%Matrice de transformation
T = T01*T12*T23*T34*T45*T56

```

Le point terminal du bras est confondu au centre du repère terminal O_6 .

Les coordonnées du point terminal sont donc $[0 \ 0 \ 0 \ 1]$ dans le repère (X_6, Y_6, Z_6) .

Voici le code Matlab:

```

%Position de l'effecteur
eff = [0;0;0;1];
xyz = T*eff;

```

Nous tenons à souligner que ces modèles seront validés avec le simulateur.

Vous noterez que nous avons utilisé « cosd » et « sind » pour préciser que nos angles sont en degré. Nous avons utilisé ces fonctions afin d'avoir une

meilleure rigueur dans les calculs. En effet, Matlab fait une approximation à 10 pour la valeur de pi ce qui fait que le $\cos(\pi/2)$ est différente de zéro. Cependant avec le cosinus décimal, le $\cosd(90)$ nous donne bien zéro.

La matrice de passage T nous sert pour le calcul du modèle géométrique inverse.

Modèle géométrique inverse

Le but de ce modèle est d'exprimer le déplacement angulaire des articulations en fonction des coordonnées cartésiennes de l'effecteur dans le repère galiléen contrairement au modèle plus haut dont le but était d'exprimer les coordonnées cartésien de l'effecteur en fonction des angles de rotation des articulations. Le modèle géométrique inverse permet donc, à partir d'une position donnée, de retrouver les angles des différentes articulations nécessaires. A cet effet, nous avons expérimenté la méthode de Paul, qui est une méthode de calcul nous permettant de calculer ce modèle. Malheureusement cette méthode est fastidieuse en termes de temps de calculs car il faut trouver la représentation du repère lié à l'organe terminal.

Par conséquent, nous avons opté pour une seconde méthode, où l'on pose des contraintes sur quatre articulations en les bloquant (annulation des angles associés) et on retrouve les deux autres angles. Il est clair qu'une telle simplification ne permettra pas d'accéder à tous les points normalement accessibles par le KUKA Kr5 mais les angles que nous n'annulons pas sont prépondérants au mouvement du KUKA. Nous obtenons alors :

$$\begin{aligned}\theta_1 &= \text{atan}(Y/X); \\ \theta_3 &= \text{asin}(-1*(Z-1000)/185); \end{aligned}$$

Modèle cinématique directe

Ce modèle calcule les vitesses linéaires à partir des vitesses articulaires. Pour calculer ce modèle, il faut calculer les vitesses angulaires qui sont les dérivées des angles de rotations portés par l'axe autour duquel s'effectue la rotation.

Supposons une rotation du repère 1 par rapport au repère 0 d'angle θ portée par l'axe z. La vitesse angulaire représente le moment du torseur cinématique et

Sa vitesse angulaire est :

$$\vec{\omega}(1/0) = \begin{pmatrix} 0 \\ 0 \\ \dot{\theta} \end{pmatrix}$$

Par analogie, la formule générale est la suivante:

$$\vec{\omega}(i/0) = \sum_1^n \vec{\omega}(i/i-1)$$

Après avoir déterminé la vitesse angulaire, la vitesse linéaire s'obtient à partir du torseur cinématique.

$$\vec{\tau} \begin{pmatrix} \vec{\omega}(i/0) \\ \vec{V}(O_{i-1}) \end{pmatrix} \Rightarrow \vec{V}(O_i) = \vec{V}(O_{i-1}) + \vec{\omega}(i/0) \wedge \overrightarrow{O_{i-1}O_i}$$

Voici le résultat après calcul sur Matlab :

% pour plus de clarté nous affichons les vitesses articulaires pour les trois premiers repères (1, 2 et 3) par rapport au repère zéro. Le résultat est affiché sous forme vectorielle.

```
w_1 =
    0
    0
    t_p_1
```

```
w_2 =
 -t_p_2*sin(t1)
 t_p_2*cos(t1)
    t_p_1
```

```
w_3 =
 (-t_p_2-t_p_3)*sin(t1)
 cos(t1)*(t_p_2+t_p_3)
    t_p_1
```

$$v_0 =$$

$$\begin{matrix} 0 \\ 0 \\ 0 \end{matrix}$$

$$v_1 =$$

$$\begin{matrix} -t_{p_1} * d1 \\ t_{p_1} * a1 \\ 0 \end{matrix}$$

$$v_2 =$$

$$\begin{matrix} -t_{p_1} * d1 - t_{p_1} * a2 \\ t_{p_1} * a1 \\ -t_{p_2} * \sin(t1) * a2 \end{matrix}$$

$$v_3 =$$

$$\begin{matrix} -t_{p_1} * d1 - t_{p_1} * a2 + \cos(t1) * (t_{p_2} + t_{p_3}) * a3 \\ t_{p_1} * a1 - (-t_{p_2} - t_{p_3}) * \sin(t1) * a3 \\ -t_{p_2} * \sin(t1) * a2 \end{matrix}$$

Modèle cinématique inverse

Il nous permet de retrouver, à partir de la vitesse linéaire du point terminal, les vitesses et angles articulaires. Mais dans notre étude, il ne nous servira pas tellement. Malheureusement nous ne l'avons pas calculé.

Moment cinématique:

Le moment cinématique décrit l'état général de la rotation du système physique. On peut l'assimiler à la quantité de mouvements. On le calcule à partir de la matrice d'inertie et de la vitesse articulaire par rapport au repère zéro.

$$\vec{K}(O_i, i/O) = I_i \cdot \vec{\omega}(i/O)$$

Voici les résultats après calcul sur Matlab :

K1 =

$$\begin{matrix} -2641/500 * t_{p_1} \\ -1021/125 * t_{p_1} \\ 6215631590794461/137438953472 * t_{p_1} \end{matrix}$$

K2 =

$$\begin{aligned} & -1089307816272855/34359738368*t_{p_2}*\sin(t1)-1/200*t_{p_2}*\cos(t1)-3/500*t_{p_1} \\ & 1/200*t_{p_2}*\sin(t1)+2832117377901003/137438953472*t_{p_2}*\cos(t1)+4739939102 \\ & 005133/549755813888*t_{p_1} \\ & 3/500*t_{p_2}*\sin(t1)+4739939102005133/549755813888*t_{p_2}*\cos(t1)+1971023714 \\ & 052997/137438953472*t_{p_1} \end{aligned}$$

Modèle dynamique

Ce modèle définit les équations du mouvement du robot, qui permettent d'établir les relations entre les couples ou forces exercés par les actionneurs et les positions, vitesses et accélérations des articulations.

Moment dynamique :

H1 =

$$\begin{aligned} & -2641/500*a1 \\ & -1021/125*a1 \\ & 6215631590794461/137438953472*a1 \end{aligned}$$

H2=

$$\begin{aligned} & -1089307816272855/34359738368*\omega_2*a1*\sin(\theta_1)- \\ & 1/200*\omega_2*a1*\cos(\theta_1)-3/500*\omega_1*a2 \\ & 1/200*\omega_2*a1*\sin(\theta_1)+2832117377901003/137438953472*\omega_2*a1*\cos(\theta_1)+ \\ & 4739939102005133/549755813888*\omega_1*a2 \\ & 3/500*\omega_2*a1*\sin(\theta_1)+4739939102005133/549755813888*\omega_2*a1*\cos(\theta_1)+ \\ & 1971023714052997/137438953472*\omega_1*a2 \end{aligned}$$

Calcul de l'énergie cinétique :

Ec_1 =

$$\begin{aligned} & 41317/1000+180*(\omega_1^2*\sin(\theta_1)^2/(\cos(\theta_1)^2+\sin(\theta_1)^2)^2+\omega_1^2 \\ & *\cos(\theta_1)^2/(\cos(\theta_1)^2+\sin(\theta_1)^2)^2)^{1/2}+6215631590794461/27487790 \\ & 6944*\cos(\theta_1)*\omega_1 \end{aligned}$$

$$E1 = \frac{1}{2} * m1 * (t_p_1^2 * d1^2 + t_p_1^2 * a1^2) + 6215631590794461 / 274877906944 * t_p_1^2$$

Simulation 3D

Pour valider les modèles dynamiques, géométriques direct et inverse, nous avons réalisé une simulation 3D. Le simulateur a été développé en langage C, à l'aide et de la bibliothèque OPENGL de l'IDE (environnement de développement intégré) DEV C++ (car nous avons développé sous windows). Le but du simulateur est de visualiser et anticiper sur les mouvements qu'effectuent l'ensemble des robots.

Grâce à des touches du clavier, vous pourrez agir sur les articulations du robot Kuka et sur les roues du robot holonome. Nous avons pris la pointe de la dernière articulation comme effecteur.

Voici les commandes :

A	Rotation de l'articulation 1 du kuka dans le sens trigonométrique
a	Rotation de l'articulation 1 du kuka dans le sens anti-trigonométrique
E	Rotation de l'articulation 3 du kuka dans le sens trigonométrique
e	Rotation de l'articulation 3 du kuka dans le sens anti-trigonométrique
F	Translation longitudinale ou latérale du châssis mobile
f	Translation longitudinale ou latérale du châssis mobile
K	Zoom avant
k	Zoom arrière
M	Rotation du châssis mobile dans le sens trigonométrique
m	Rotation du châssis mobile dans le sens anti-trigonométrique
V	Abaissement du châssis mobile
v	Soulèvement du châssis mobile
R	Rotation de l'articulation 4 du kuka dans le sens trigonométrique
r	Rotation de l'articulation 4 du kuka dans le sens anti-trigonométrique
T	Rotation de l'articulation 5 du kuka dans le sens trigonométrique
t	Rotation de l'articulation 5 du kuka dans le sens anti-trigonométrique
Y	Rotation de l'articulation 6 du kuka dans le sens trigonométrique
y	Rotation de l'articulation 6 du kuka dans le sens anti-trigonométrique
Z	Rotation de l'articulation 2 du kuka dans le sens trigonométrique
z	Rotation de l'articulation 4 du kuka dans le sens anti-trigonométrique
w	Modélisation géométrique inverse

CONCLUSION

Ce projet s'est avéré pluridisciplinaire en faisant appel à des connaissances dans le domaine de la mécanique, de l'informatique, de la commande et de l'électronique de puissance. Ceci nous a permis de constater et d'apprécier les interconnexions qui existent entre les notions acquises en cours. Ce caractère diversifié du projet nous a permis d'acquérir des notions supplémentaires notamment lors de la modélisation du châssis et de prise en main de la librairie OpenGL lors de la modélisation 3D. Nous avons aussi vu à quel point la modélisation des systèmes peut-être complexes avec peu d'interactions avec le monde physique et nécessitant un niveau d'abstraction élevé. En perspective de ce travail il serait intéressant de poursuivre avec l'inclusion du modèle dynamique du châssis. Il serait également intéressant d'améliorer le simulateur en lui ajoutant une interface HM.

Ce projet nous a également permis de travailler en autonomie. Nous avons ainsi pu constater nos limites et nos erreurs en matière de gestion du temps et ceci nous permettra nous nous améliorer pour les projets à venir.

Nous remercions Monsieur Merzouki, pour son encadrement et pour nous avoir proposé ce sujet.