

**REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET  
POPULAIRE**

**MINISTERE DE L'ENSEIGNEMENT SUPERIEUR  
ET DE LA RECHERCHE SCIENTIFIQUE**



المدرسة الوطنية المتعددة التقنيات  
Ecole Nationale Polytechnique

**ECOLE NATIONALE POLYTECHNIQUE**

**Département de Génie Electrique**

**Mémoire de fin d'études**

**En vue de l'obtention du diplôme d'ingénieur d'Etat en  
Automatique**

**MODELISATION, IDENTIFICATION ET  
COMMANDE EN TEMPS REEL DU  
ROBOT ROB3.**

Proposé et dirigé par :  
***M<sup>r</sup> M.TADJINE***  
***M<sup>r</sup> J.NASSERI***

Etudié par :  
***M<sup>r</sup> BENAICHOUCHE Abed***

*Promotion : Juin 2008*

**Laboratoire de Commande des Processus  
E.N.P.10, Avenue Hassen-Badi, EL-HARRACH, ALGER**

**ملخص:** العمل المقدم في المذكرة هو عبارة عن دراسة إنشاء جهاز تحكم ماسخ خصيصا بميكرومراقب (microcontrôleur) بعين التحكم في ذراع الية من نوع ROB3 ، لذلك قمت بتقيدم دراسة هندسية هيكلية لهذا الذراع الآلي و كل العلاقات الرياضية المستنتجة ، أيضا قمت بدراسة المحركات الفعالة لهذا الهيكل ، خورزميات التحكم قدمت في الحالة المستمرة .

دراسة الميكرومراقب و اشغاله الداخلى منحتنى إمكانية إدماج هذه الخورزميات التحكمية ، فى اخر جزء من العمل أقدم الإدارة التحكمية و كيفية اشغالها ، برمجتها و تكيفها حتى تقوم بالتحكم في محرك من نوع BUHLER ("الفعاليات") .

**كلمات مفتاحية** ذراع الآلي ,متحكمات الذراع الآلي ,الفعاليات ,بميكرومراقب

**Résumé:** Le travail présenté dans ce mémoire est une conception et réalisation d'une carte de commande à base de microcontrôleur afin de pouvoir commander un bras de robot manipulateur de série ROB3. Pour ce faire, nous avons présenté une étude sur l'architecture structurelle du robot et les différentes modélisations résultantes, ainsi qu'une description de ces actionneurs, des algorithmes de réglages ont été, une étude du microcontrôleur et son fonctionnement interne nous a permis l'implémentation des ces algorithmes de commande. Dans la dernière section de ce travail, nous précisons le schéma de principe de fonctionnement de l'interface, la configuration et la programmation nécessaire pour sa mise en fonction pour la commande des moteurs BUHLER ("actionneurs").

**Mots clés:** bras manipulateur, commande des robots, actionneurs, microcontrôleur.

**Abstract:** The work presented in this memory is a conception and realization of command interface basis microcontroller in ordering to control a manipulator arm series ROB3. With this intention, we had presented a study of the structural architecture of the robot and the different resulting models, also a description of the actuators. Algorithms of regulation had been established in the continuous domain, a study of the microcontroller permitted the implementation of these algorithms. In the last section of this work, we set the principle diagram of the interface which is necessary to configure and program to put the according functions for the setting of the motors BUHLER ("actuators")

**Key words:** manipulator arm, Robot's control, actuators, microcontroller.

# REMERCIEMENTS

*Le travail présenté dans ce mémoire a été mené au laboratoire d'électronique appliquée de l'université Ibn Khaldoune de Tiaret.*

*Je tiens d'abord à remercier mon promoteur, Monsieur M. Tadjine pour avoir accepté mon encadrement et pour l'ensemble de ses conseils et les efforts Fournis durant toute l'année.*

*De mêmes, j'adresse mes remerciements à Messieurs : M. Meslem et J. Nasserri pour m'avoir offert l'hospitalité et qui m'ont aidé à travailler dans de bonnes conditions.*

*Je remercie aussi Messieurs H. Chekireb, R.Iloul et O.Stihi pour leurs précieux conseils et orientations lesquels m'ont été d'une très grande utilité.*

*Mes remerciements vont également à ceux qui ont participé à ma formation.*

*Abed*

# *DEDICACES*

*Je dédie ce travail avant tout, à mes très chers parents; qui m'ont tout offerts, et j'espère que j'aurais un jour l'occasion de les remercier d'avantage.*

*A mes frères Nacer-eddine, Yacine, Abdelatif, Mohamed et ma sœur Ratiba, qui compte le plus pour moi.*

*A tout les membres de ma famille ; Mon oncle Nacer-eddine, Tonton Hamid, Mes tantes Assia, Fahima, Soad, Khadija, et a ma grand-mère, qui m'ont apporté leur soutien durant mes études.*

*A la famille : BENAICHOUCHE, AIT OUDIA, SAWANE, CHAKOR, OTMANI, ...*

*A mes amis : Fahim, Djilali, Mohamed, Khaled, Zaki, Hamani, Walid, Jack, Cherif, Kadi, Houcine, Oussama et a toute la promotion 2008.*

*Abed BENAICHOUCHE*

\*

# TABLES DES FIGURES ET TABLEAUX

# *Liste des figures et tableaux*

## **CHAPITRE I**

**Figure 1.1 :** *Représentation schématique d'une chaîne cinématique.*

**Figure 1.2 :** *Table des symboles.*

**Figure 1.3 :** *L'espace du travail du bras robotique Rob3.*

**Figure 1.4 :** *Les axes du robot ROB3.*

**Figure 1.5 :** *Les dimensions du robot ROB3 (vue de haut).*

**Figure 1.6 :** *Les dimensions du robot ROB3 (vue de face).*

**Figure 1.7 :** *la pince du robot ROB3.*

**Figure 1.8 :** *unité de commande Teachbox.*

**Figure 1.9 :** *Constitution d'un moteur à courant continue à aimant permanent.*

**Figure 1.10 :** *Schéma équivalent.*

**Figure 1.11 :** *Fonction de transfert d'un moteur à courant continu.*

**Figure 1.12 :** *Le capteur du robot ROB3.*

**Figure 1.13 :** *Caractéristique du capteur potentiométrique.*

**Tableau 1.1 :** *Les angles de rotation du robot ROB3.*

**Tableau 1.2 :** *Caractéristique des capteurs du robot ROB3.*

## **CHAPITRE II**

**Figure 2.1 :** *Changement de base des coordonnées d'un point.*

**Figure 2.2 :** *Rotation d'un solide autour d'un point fixe.*

**Figure 2.3 :** *Exemple de rotation plane de la rotation de centre  $O$  et d'angle  $\theta$ , autour de  $z$ .*

**Figure 2.5 :** *Rotations successives dans le paramétrage par les angles de roulis, tangage et lacet.*

**Figure 2.6 :** *Transformation des repères.*

**Figure 2.7 :** *Transformation d'une translation pure.*

**Figure 2.8 :** Transformation de rotation autour de l'axe  $x$ .

**Figure 2.9 :** Composition des transformations: multiplication à droite.

**Figure 2.10 :** Composition des transformations: multiplication à gauche.

**Figure 2.11:** Robot à structure ouverte simple.

**Figure 2.12 :** Paramètres géométrique dans le cas d'une structure ouverte simple.

**Figure 2.13 :** Les axes du robot ROB3.

**Figure 2.14 :** Placement des repères.

**Figure 2.15 :** Représentation des membres des robots dans un plan.

**Figure 2.16 :** Représentation au niveau des robots.

**Figure 2.17 :** Calcul des angles  $Q1$ ,  $Q2$  et  $Q3$ .

**Figure 2.18 :** réponse indicielle du bras pour les trois axes.

**Figure 2.19 :** réponse indicielle du bras pour le modèle identifié.

**Figure 2.20.a :** erreurs de modélisation axe 1.

**Figure 2.20.b :** erreurs de modélisation axe 2.

**Figure 2.20.c :** erreurs de modélisation axe 3.

**Tableau 2.1 :** Paramètres géométriques du ROB3.

**Tableau 2.2:** Les paramètres des modèles de simulation.

## **CHAPITRE III**

**Figure 3.1 :** Schéma classique d'une commande PID.

**Figure 3.2. :** Schéma classique d'une commande PD.

**Figure 3.3.1.a :** Suivi en position.

**Figure 3.3.1.b :** Erreur en position.

**Figure 3.3.1.c :** suivie en vitesse.

**Figure 3.3.2.a :** suivie en position.

**Figure 3.3.2.b :** erreur en position.

**Figure 3.3.2.c :** suivie en vitesse.

**Figure 3.3.3.a :** suivie en position.

**Figure 3.3.3.b :** erreur en position.

**Figure 3.3.3.c** : *suivie en vitesse.*

**Figure 3.3.4.a** : *suivie en position.*

**Figure 3.3.4.b** : *erreur en position.*

**Figure 3.3.4.c** : *suivie en position.*

**Figure 3.4** : *Schéma de la simulation sur Matlab de la commande par mode de glissement.*

**Figure 3.5.1.a** : *suivie en position.*

**Figure 3.5.1.b** : *suivie en position.*

**Figure 3.5.1.c** : *suivie en vitesse.*

**Figure 3.5.2.a** : *suivie en position.*

**Figure 3.5.2.b** : *suivie en position.*

**Figure 3.5.2.c** : *suivie en vitesse.*

**Figure 3.6.1.a** : *suivie en position.*

**Figure 3.6.1.b** : *erreur en position.*

**Figure 3.6.1.c** : *suivie en vitesse.*

**Tableau 3.1** : *Les paramètres des modèles de simulation.*

## **CHAPITRE IV**

**Figure 4.1** : *Bloc diagramme du système de commande.*

**Figure 4.2** : *Organigramme de l'algorithme de régulation PID.*

**Figure 4.3** : *Organigramme de génération du signal de commande.*

**Figure 4.4** : *Définition des pins et boîtier externe.*

**Figure 4.5** : *Circuit imprimé de la carte de commande.*

**Figure 4.6** : *La partie puissance.*

**Figure 4.7** : *Circuit imprimé de la carte de commande.*

**Figure 4.8** : *Implémentation des composants sur la carte de commande.*

**Figure 4.9.a** : *suivie en position (axe 1).*

**Figure 4.9.b** : *erreur en position (axe 1).*

**Figure 4.9.c** : *suivie en position (axe 2).*

**Figure 4.9.d** : *erreur en position (axe 2).*

# Sommaire

<b>Introduction générale</b> .....	1
<b>Chapitre 1 : Description du manipulateur ROB3</b>	
1.1. Notations .....	3
1.2. L'espace du travail du manipulateur Rob3.....	4
1.3. Propriétés .....	4
1.3.1. Propriétés mécaniques .....	4
1.3.2. Propriété électronique .....	6
1.3.3. L'unité Teachbox .....	7
1.4. Les modes opératoires de ROB3.....	8
1.5. Limite en commande .....	9
1.6. Description des actionneurs.....	10
1.6.1. Avantages des actionneurs électriques .....	10
1.6.2. Moteur électrique à courant continu .....	11
1.6.2.1. Constitution d'un moteur à courant continu .....	11
1.6.2.2. Etude mécanique .....	12
1.6.2.3 Etude électrique .....	13
1.6.2.4. Fonction de transfert du moteur à courant continu .....	14
1.7. Description des capteurs .....	14
1.7.1. Caractéristique du capteur .....	15
1.8. Conclusion.....	16
<b>Chapitre 2 : Modélisation et identification</b>	
2.1 Modélisation .....	17
2.1.1. Notations et définitions .....	17
2.1.1.1. Points .....	17
2.1.1.2. Solides .....	18
2.1.1.3 Transformations rigides.....	18
2.1.1.4. Rotations .....	18
2.1.1.4.1. Matrice de rotation .....	18
2.1.1.4.2. Rotation d'un point appartenant à un solide .....	19
2.1.1.4.3. Exemple .....	20
2.1.1.4.4. Rotation d'un vecteur .....	21
2.1.1.4.5. Propriétés des rotations .....	21
2.1.1.4.6. Représentation de l'orientation d'un solide dans l'espace ....	21
2.1.1.4.6.1. Matrice de rotation et cosinus directeurs .....	22
2.1.1.4.7. Représentation angle/axe .....	22
2.1.1.4.7.1. Vecteur rotation .....	22
2.1.1.4.7.2. Coordonnées exponentielles .....	23
2.1.1.4.7.3. Angles d'Euler classiques .....	23
2.1.1.4.7.4. Angles roulis, tangage et lacet .....	24
2.1.2. Transformation Homogène .....	25
2.1.2.1. Matrice de transformation d'une translation pure .....	26

2.1.2.2. Matrice de transformation de rotation autour des axes principaux .....	27
2.1.2.2.1. Transformation correspondante à une rotation $\theta$ autour de l'axe x .....	27
2.1.2.2.2. Matrice de transformation correspondante à une rotation $\theta$ autour de l'axe y..	27
2.1.2.2.3. Matrice de transformation correspondante à une rotation $\theta$ autour de l'axe z...	28
2.1.2.3. Propriétés des matrices de transformation homogène.....	28
2.1.3. Modélisation des robots manipulateurs .....	30
2.1.3.1. Méthodes et notations .....	30
2.1.3.2. Description de la géométrie des robots à structure ouverte simple.....	30
2.1.3.3. Application au robot ROB3 .....	32
2.1.3.4. Modélisation géométrique inverse du robot ROB3.....	35
2.1.4. Modèle cinématique .....	39
2.1.4.1. Modèle cinématique direct .....	39
2.1.4.2. Calcul du Jacobi de base.....	40
2.1.5. Modèle cinématique inverse .....	41
2.1.5.1. Forme générale du modèle cinématique inverse .....	42
2.1.5.2. Modèle cinématique inverse dans le cas régulier .....	42
2.1.5.2.1. Première méthode. ....	42
2.1.5.2.2. Deuxième méthode. ....	43
2.1.6. Modèle dynamique .....	43
2.1.6.1 Formalisme de Lagrange .....	44
2.1.6.1 Forme générale de l'équation dynamique .....	44
2.1.6.1.1. Calcul de l'énergie cinétique .....	45
2.1.6.1.2. Calcul de l'énergie potentielle.....	45
2.2. Identification .....	46
2.2.1 Description de la méthode utilisée pour l'identification .....	47
2.3. Test de validation .....	48
2.4. Conclusion .....	50

### Chapitre 3 : Stratégies de commande et simulations

3.1. Commande classique .....	51
3.1.2. Simulation .....	54
3.1.2.1. Commande Proportionnelle Dérivée .....	54
3.1.2.2. Commande Proportionnelle intégrale et Dérivée .....	57
3.1.3. Conclusion .....	59
3.2. Commande par mode glissement .....	59
3.2.1. Surface de glissement .....	60
3.2.2 Commande équivalente .....	60
3.2.3. Résultats de la simulation .....	61
3.2.4 Test de robustesse .....	63
3.2.5 Conclusion .....	64
4. Conclusion .....	64

### Chapitre 4 : Mise en œuvre pratique

4.1. Présentation et principes .....	65
4.2. Mesure de la position du moteur .....	66
4.3. Implémentation de l'algorithme de régulation .....	66
4.4. Génération du signal de commande du moteur (PWM) .....	68

4.5. Modules utilisés dans la mise en œuvre pratique .....	69
4.5.1. Présentation du L298.....	69
4.5.2. Circuit imprimé.....	72
4.6. Réalisation .....	73
4.7 Résultat pratique .....	74
4.8. Résultats d'application .....	75
4.9. Conclusion .....	75
<b>Conclusion générale.....</b>	<b>76</b>
<b>Références Bibliographiques.....</b>	<b>77</b>

INTRODUCTION  
*GENERALE*

## Introduction générale

Depuis l'avènement des technologies, l'homme a voulu se décharger des tâches répétitives et inintelligibles en cherchant à automatiser tout processus de production et cela en développant des systèmes de réglages automatiques et de gestion des tâches. L'avancement de telles idées a donné naissance à la robotique, cette branche a permis le regroupement de plusieurs domaines de connaissances tels que l'automatique, l'électronique, l'électrotechnique et la mécanique. Les objectifs principaux de la conception des robots industriels sont de réaliser des tâches avec rapidité et une grande précision. De nombreux travaux de recherche, basés essentiellement sur les stratégies de commande et de leur implémentation, ont vu le jour.

La commande de mouvement est d'une très grande importance dans le domaine de la robotique. Elle a fait l'objet d'une évolution des outils informatiques et numériques, permettant l'amélioration de la commande des actionneurs grâce à l'exploitation de leur grande vitesse de traitement de l'information. Ceci a permis d'appliquer les différentes techniques auparavant impossible à utiliser avec les anciens circuits de commande. Cela a abouti à l'élaboration de nouveaux matériels spécialisés dans la commande.

Le présent travail a pour objet l'étude du robot *ROB3* dont le but est de concevoir et de réaliser une carte de commande pour l'asservissement articulaire en position.

Le traitement de robot du point de vue mécanique et structurel nous paraît important. Pour cela, nous détaillons les méthodes de modélisation des robots manipulateurs et nous les appliquons au robot *ROB3*. De même, un modèle mathématique du moteur à courant continu est donné. Dans cette partie nous allons citer les avantages des actionneurs électriques, décrire en détail le moteur à courant continu en commençant par sa constitution et ses caractéristiques mécaniques et électriques, lesquels seront utiles au dimensionnement du régulateur.

Les techniques de commande des robots manipulateurs sont nombreuses, leur utilisation dépend essentiellement de la structure du robot (complexité) et des tâches à effectuer. La synthèse des différentes commandes est basée, principalement, sur le modèle

dynamique du robot, ce dernier étant tiré à partir d'une méthode expérimentale. Nous allons donc traiter deux types de commandes qui feront l'objet de simulations.

Ces commandes sont :

- la commande classique de type PID décentralisé ;
- la commande par mode glissement ;

Les simulations ont été effectuées sous environnement *Matlab* dont nous donnons la représentation d'état du modèle du robot *ROB3* ainsi que sa commande.

Enfin, une carte de commande, réalisée pour l'asservissement en position d'un moteur à courant continu est présentée afin de commander le robot *ROB3*. Une étude approfondie du PIC16F877 est établie dans le but d'utiliser toutes les ressources disponibles. Le PIC16F877 est chargé d'effectuer la tâche du calculateur de processus et d'exécuter l'algorithme du réglage.

CHAPITRE I  
*DISCRIPTION DU  
MANIPULATEUR  
ROB3.*

# 1. Description du manipulateur ROB3

## 1.1. Notations

Les robots se distinguent par la nature et la succession des articulations. La figure 1.1 présente un exemple d'une représentation schématique d'une chaîne cinématique d'un robot à 2 axes. La figure 1.2 présente la signification des différents éléments de la figure 1.1. Pour une classification des types des robots, la cinétique des axes principaux est déterminante [1].

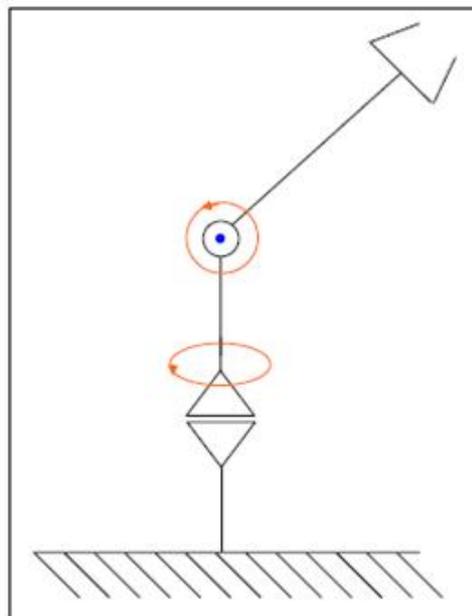


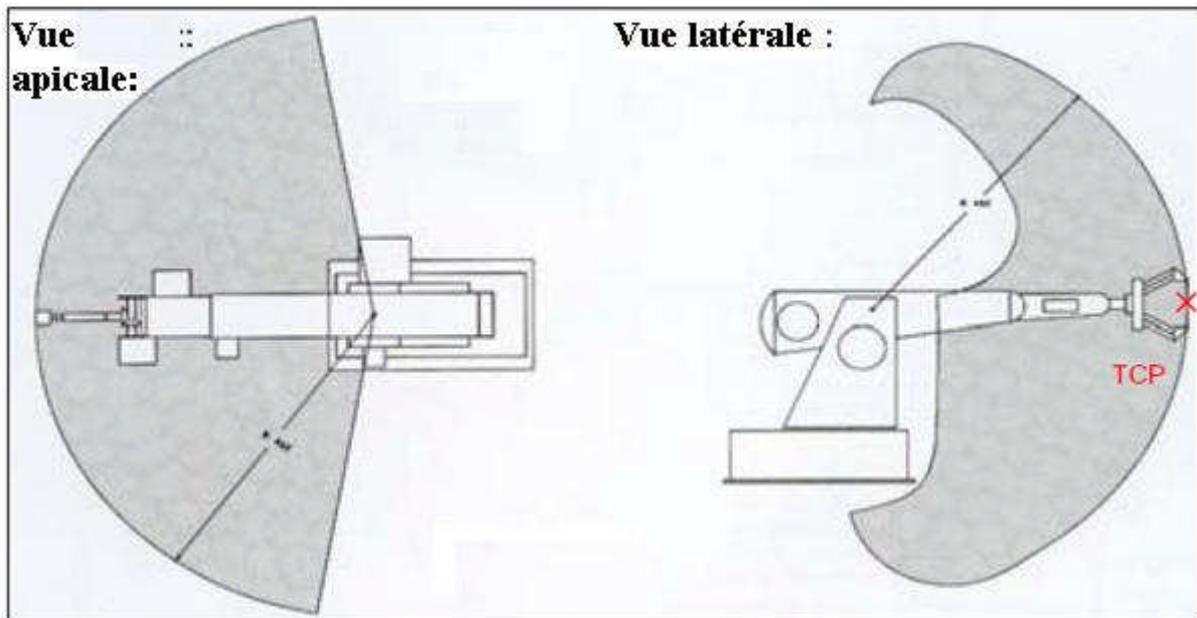
Figure 1.1 : Représentation schématique d'une chaîne cinématique.

Nom	Symbole
Axe prismatique	
Axe rotative	
Outil/actionneur	
Joint	

Figure 1.2 : Table des symboles.

## 1.2. L'espace de travail du manipulateur Rob3

La figure 1.3 décrit l'espace de travail du bras robotique ROB3. Si l'un des objets est à l'extérieur de ce secteur il ne peut pas être atteint. La partie indiquée par le bleu c'est la zone de points que le robot avec son centre d'outil peut les atteindre [1].



**Figure 1.3 :** *L'espace de travail du bras robotique Rob3.*

## 1.3. Propriétés

### 1.3.1. Propriétés mécaniques

La structure ROB3 est composée de pièces en aluminium vissés entre elles. Chacun des cinq axes indépendants est lié par un servomoteur à courant continu. La puissance de transmission est assurée par des engrenages. Pour la transmission des axes 3 et 4 des chaînes dentées sont utilisées en plus. La position absolue de tous les axes sont déterminés par des capteurs potentiométriques. Cette caractéristique la plus marquante du ROB3 garantie que, même après le passage à une autre position, le robot repère la position exacte de chaque axe.

Comme il est représenté dans la figure 1.4, le bras manipulateur ROB3 a en total 5 axes de rotations. Le tableau 1.1 permet de représentés les différentes limites des angles du bras manipulateur [2].

Les figures 1.5 et 1.6 montrent les différentes longueurs des tiges selon 2 angles différents.

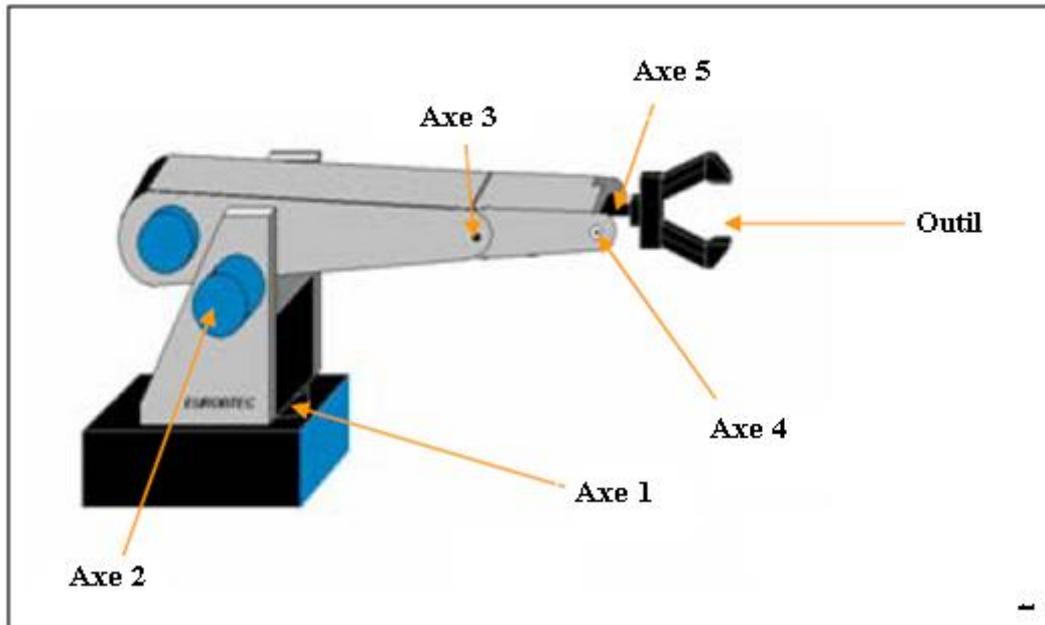


Figure 1.4 : Les axes du robot ROB3.

Les Articulations	Axe	Limite articulaire	Résolution
Axe 1 : Base	Q0	+80° ... 0° ... -80°	0 ... 255
Axe 2 : Epaule	Q1	+50° ... 0° ... -50°	0 ... 255
Axe 3 : Coude	Q2	0 ... +100°	0 ... 255
Axe 4 : Poignet	Q3	+100° ... 0° ... -100°	0 ... 255
Axe 5 : Orientation	Q4	+100° ... 0° ... -100°	0 ... 255
Outil (Pince)		0 ... 60 mm	0 ... 255

Tableau 1.1 : Les angles de rotation du robot ROB3.

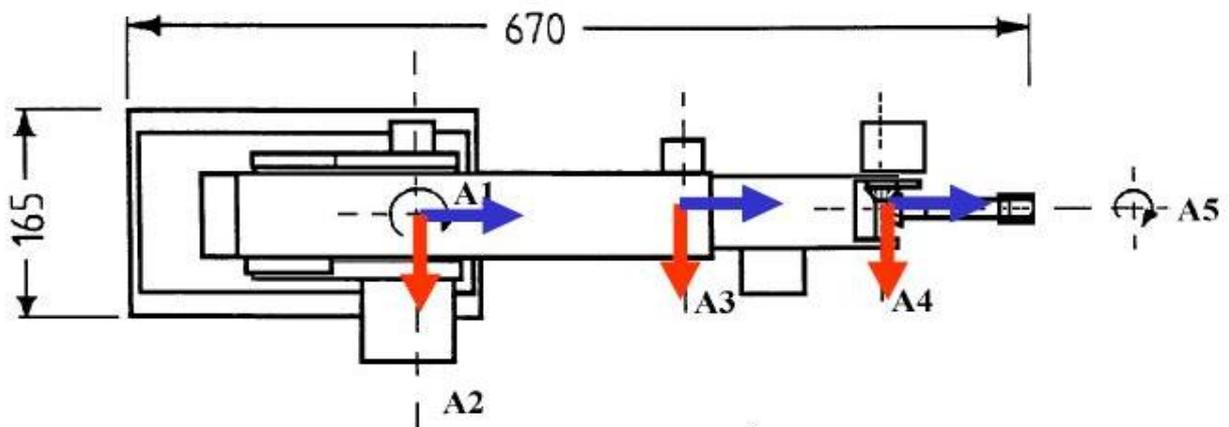


Figure 1.5 : Les dimensions du robot ROB3 (vue de haut).

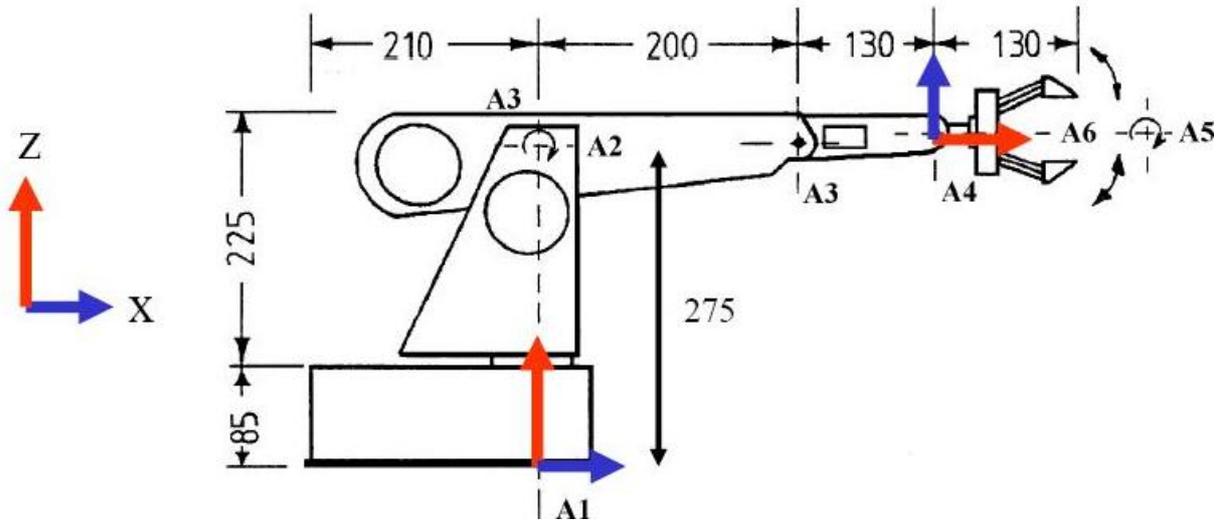


Figure 1.6 : Les dimensions du robot ROB3 (vue de face).

### 1.3.2. Propriétés électroniques

Le robot ROB3 à un système de contrôle intégré qui peut être programmé directement à travers un clavier « l'unité Teachbox », le robot est configuré en introduisant une séquence d'instructions pour faire bouger le bras vers des positions désirées, ces instructions sont mémorisées dans une mémoire comme un programme de commande [2].

Pour les gens qui préfèrent travailler avec un ordinateur, le ROB3 peut aussi être commandé à travers une interface série RS-232, il est possible de contrôler ou programmer le robot à travers une unité IBM compatible. Pour les personnes qui veulent développer leurs propres programmes de commande, des interfaces sont disponibles comme des fichiers introduits sur les langages Basic, Pascal et C. ces programmes contiennent des sous-routines pour être prêts à communiquer avec le robot à travers l'interface série [2].

Le contrôleur est intégré dans la base du robot. Un processeur 8031 est le cœur du contrôle. Les programmes utilisés peuvent être stockés dans une mémoire auxiliaire. La capacité de mémoire est de 8 ko, ce qui correspond à environ 1000 instructions programme. En outre, le robot peut être connecté avec un pc via l'interface série RS 232. 8 entrées/sorties digitales, compatible TTL sont à la disposition de l'utilisateur par l'intermédiaire d'un connecteur port parallèle [2].

La pince est alimentée par un servomoteur à courant continu par le biais d'un câble comme le montre la figure 1.7, de très minces pièces peuvent être manipulées par l'outil. Le manipulateur ROB3 fait partie des robots de haute qualité qui a démontré sa fiabilité durant plusieurs années, dans tous les domaines de l'éducation et de la formation. Bien que l'objet

d'un processus continu d'amélioration de la fiabilité du modèle, 5 axes rotatifs et une pince parallèle, plusieurs possibilités d'applications peuvent être réalisées. Comme le contrôleur est intégré dans la base du robot, un accès facile et rapide d'installation et la mise en œuvre peut être assurée avec une facilité d'utilisation.



**Figure 1.7** : la pince du robot ROB3.

### 1.3.3 L'unité Teachbox

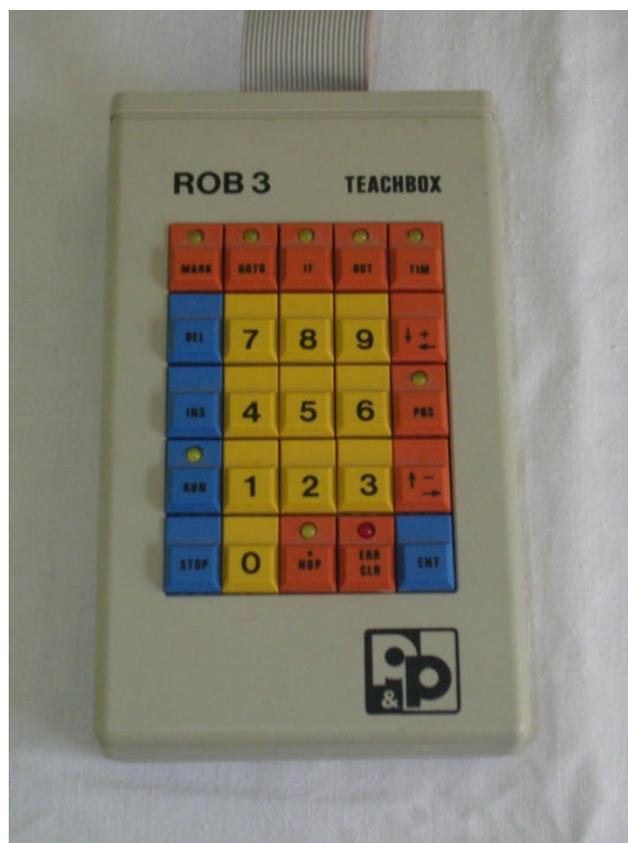
Le Teachbox (figure 1.8) permet de contrôler et programmer le ROB3 sans utiliser un ordinateur, cela en utilisant un clavier digital avec 25 boutons, répartis en trois groupes de boutons :

- boutons numériques : pour donner une valeur décimale ;
- boutons de commandes : pour indiquer la commande à exécuter directement ;
- boutons d'instructions : pour entrer le programme d'instruction de commande.

Pour programmer le robot il suffit de connecter le ROB3 avec le Teachbox via le câble série RS232; l'unité peut être facilement débranchée une fois le programme lancé. Dès que le ROB3 est en mode opératoire il reste connecté avec Teachbox seules trois sorties numériques (DO 1-3) et 5 entrées numériques (DI 1-5) sont disponibles. Chaque bouton d'instruction est joint à une LED qui s'allume pour confirmer que l'entrée est correcte ou non, si une fausse

instruction est entrée la LED de ERR s'allume, elle s'éteint une fois que l'erreur est corrigée, en appuyant sur CLR. Toutes les instructions doivent être confirmées par le bouton ENT [2].

- Boutons numériques : utilisés pour entrer le numéro de l'axe désigné et la valeur de la position.
- Boutons de commandes : (RUN, STOP, INS, DEL, POS, OUT) ces commandes sont exécutées immédiatement.
- Boutons d'instructions : (POS, TIM, OUT, MARK, IF, GOTO, NOP) : une fois l'instruction entrée elle sera sauvegardée dans la mémoire du robot par ordre d'entrée (FIFO), elles sont exécutées en mode RUN ou STEP.



**Figure 1.8 :** unité de commande Teachbox.

#### 1.4 Les modes opératoires de ROB3

Le robot contient six modes de système de commande [2].

1. INPUT mode : pour la création d'un programme de contrôle par l'entrée des séquences d'instructions. C'est le mode par défaut, il est activé quand le système est mis sous tension.

2. POSITION mode : ce mode est sélectionné en appuyant sur un bouton numérique 1-6, alors on peut tourner dans un sens ou l'autre avec les boutons + et -. Avec ce mode on peut aussi positionner les axes directement avec la séquence POS (axe.1). (position.n) ENT. en appuyant sur le CLR en retournant au premier mode.
3. RUN mode : quand la séquence instructions : RUN (étiquette m) ENT est introduite, le ROB3 commence à les exécuter dans sa mémoire en commençons a partir de l'étiquette m.
4. BREAK mode : quand le robot est dans le mode RUN, tous les boutons sauf STOP sont inactifs. Appuyant sur STOP CLR active le mode BREAK. On peut retourner au mode RUN avec RUN ENT (=CONTINUE).appuyant sur STOP ENT permet la sélection de chaque modes.
5. STEP mode : introduisant RUN (étiquette m) ENT sélectionne le mode STEP, débutant a l'étiquette m. On peut après exécuter le programme par pas en appuyant sur le bouton +. Pour quitter se mode il suffit d'appuyer sur CLR.
6. DISPLAY mode : introduisant STOP (étiquette m) ENT sélectionne le mode DISPLAY, débutant a l'étiquette m. En appuyant sur le bouton +, la prochaine instruction sera sauvegardée dans la mémoire et affichée (dans la LED correspondante), mais l'instruction n'est pas exécuté sur le ROB3. Pour quitter ce mode il suffit d'appuyer sur le bouton CLR).

### 1.5. Limite en commande

Le robot ROB3 a un système de contrôle intégré à base d'un microprocesseur Intel 8031, ce dernier permet de commander le manipulateur uniquement en boucle ouverte. Dans le cadre de notre travail, nous nous sommes rendu compte que la carte de commande originale était défectueuse, outre l'absence des références des différents circuits électroniques composants, cette situations nous a rendue la compréhension du fonctionnement de la carte originale impossible.

Ces facteurs nous a obligé à concevoir une nouvelle carte de commande qui permet de contrôler le système en introduisant le calcul dynamiques de l'erreur c'est-à-dire la commande en boucle fermé.

## 1.6. Description des actionneurs

Un robot réalise des fonctions mécaniques telles qu'un déplacement ou un positionnement. Pour cela, il a besoin d'un actionneur qui va réaliser à la fois une transmission d'information et une conversion d'énergie [3]. Plusieurs technologies existent : on rencontre des actionneurs électriques, pneumatiques et hydrauliques. Les premiers sont le plus souvent des moteurs en rotation, un mécanisme permettant éventuellement de modifier la nature du mouvement. Ce sont ces machines qui nous intéressent ici.

Les moteurs employés font appel aux mêmes principes que ceux qu'on utilise en électrotechnique classique, mais leurs caractéristiques et leurs technologies sont différentes. Tout d'abord, la puissance en général modeste des machines fait qu'on fait souvent appel à des aimants permanents plutôt qu'à des inducteurs bobinés, ce qui simplifie la réalisation et élimine les pertes par effet joule correspondantes. Ensuite, on ne cherche pas à optimiser les mêmes paramètres. Les grandeurs liées à la conversion d'énergie (puissance, rendement) sont certes importantes, mais les grandeurs liées à l'information (précision, rapidité de réponse) sont primordiales. Ainsi, on minimise l'inertie des parties tournantes en adoptant des structures particulières ou une géométrie adaptée pour réaliser des moteurs à réponse rapide.

Les performances d'un actionneur électrique sont intimement liées à celles de son environnement : le convertisseur d'énergie et sa commande électronique, l'éventuel asservissement avec ses capteurs et les mécanismes associés à la charge. Il est important d'en tenir compte lors d'une comparaison de coût entre plusieurs solutions

Dans cette partie je vais citer les avantages des actionneurs électriques, décrire en détail le moteur à courant continu en commençant par sa constitution ainsi que ses caractéristiques mécaniques et électriques [4].

### 1.6.1. Avantages des actionneurs électriques

Par rapport à leurs concurrents hydrauliques et pneumatiques, les actionneurs électriques présentent un certain nombre d'avantages parmi lesquels [3]:

- une énergie facilement disponible, soit à partir du secteur, soit à partir de batteries pour les engins autonomes ;
- une adaptation aisée de l'actionneur et de sa commande du fait de la nature électrique de l'ensemble des grandeurs.

Les progrès récents ont permis un élargissement du domaine d'emploi des actionneurs électriques. On peut citer :

- l'amélioration des performances des moteurs grâce en particulier à la disponibilité d'aimants plus efficaces ;
- la simplification de l'électronique associée grâce aux avancées dans le domaine des composants (circuits intégrés, transistors de puissance ...).

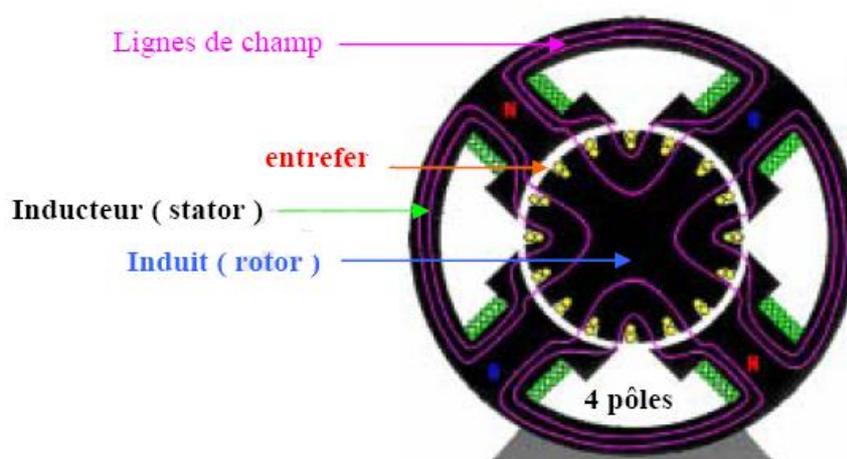
### 1.6.2. Moteur électrique à courant continu

Le moteur à courant continu est l'actionneur électrique le plus classique. Utilisé depuis longtemps, il est toujours présent dans de nombreuses applications. Beaucoup de structures différentes existent, mais le principe de base est le même [4].

#### 1.6.2.1. Constitution d'un moteur à courant continu

On s'intéresse ici aux moteurs classiques, c'est-à-dire ceux qui sont conçus à peu près de la même façon que les machines de forte puissance.

Comme toute machine tournante, le moteur à courant continu comporte une partie fixe, le stator et une partie mobile, le rotor, séparées par un entrefer figure 1.9 Le stator porte des aimants qui sont chargés de créer le champ magnétique dans l'entrefer [3].



**Figure 1.9 :** Constitution d'un moteur à courant continu à aimant permanent.

### 1.6.2.2. Etude mécanique

Le phénomène essentiel dans un moteur à courant continue est la création d'un couple qui tend à faire tourner le rotor, nous avons l'expression du couple électromagnétique :

$$\Gamma_m = \frac{1}{2\pi} \frac{p}{a} N \Phi i \quad (1.1)$$

Le moment s'exprime en newtons-mètres (N·M).  $\Phi$  est le flux sous un pôle, en weber(Wb),  $i$  l'intensité du courant dans l'induit exprimé en ampère(A),  $p$  est le nombre de paire de pôle,  $a$  le nombre de paires de voies de la machine et  $N$ , le nombre de conducteur actif. Pour un moteur donnée  $a$ ,  $p$  et  $N$  sont fixes, on considère que le flux est pratiquement constant ; On pose donc :

$$\Gamma_m = K_t i \quad (1.2)$$

$K_t$  est la constante du moteur, avec cette définition, on a :

$$K_t = \frac{1}{2\pi} \frac{p}{a} N \Phi \quad (1.3)$$

L'unité de  $K_t$  est le newton-mètre par ampère ( $N \cdot M \cdot A^{-1}$ ), la valeur de ce constant est généralement précisé dans la notice technique du moteur.

L'équation mécanique d'un moteur à courant continue découle du principe fondamental de la dynamique, pour un système en rotation qui est le cas du moteur nous avons :

$$\Gamma_m = J_T \frac{d\Omega}{dt} + B_T \Omega + \Gamma_s + \frac{\Gamma_p}{n} \quad (1.4)$$

Où :

- $J_T$  représente l'inertie totale vue par le moteur est égale à :

$$J_T = J_m + \frac{J_c}{n^2} \quad (1.5)$$

$J_m$  et  $J_c$  : représente respectivement l'inertie du rotor du moteur généralement considéré comme un cylindre et l'inertie de la charge externe.

-  $n$  : est le rapport de réduction.

- $B_T$ : représente le coefficient de frottement visqueux totale vue par le moteur, il est donné par :

$$B_T = B_m + \frac{B_c}{n^2} \quad (1.6)$$

- $B_m$  et  $B_C$ : représente respectivement le coefficient de frottement visqueux au niveau du rotor du moteur et le coefficient de frottement visqueux au niveau de la charge externe.

Le coefficient  $B_m$  a deux valeurs selon que le moteur est commandé en tension ou en courant.

-  $\Gamma_s$  est le couple de frottement sec donné par :

$$\Gamma_s = \Gamma_0 \text{sign}(\Omega) \quad (1.7)$$

La fonction  $\text{sign}$  étant le signe de la vitesse  $\Omega$  et a pour valeur  $\pm 1$ .

-  $\Gamma_P$  est le couple de perturbation que subit la charge externe.

### 1.6.2.3 Etude électrique

Sur le plan électrique, le moteur à courant continu à aimant est un dipôle, l'enroulement de l'induit a une résistance  $R$  et une inductance  $L$ . la loi des mailles donne :

$$u = Ri + L \frac{di}{dt} + E \quad (1.8)$$

$E$  est définie comme étant la force contre électromotrice, elle est donnée par :

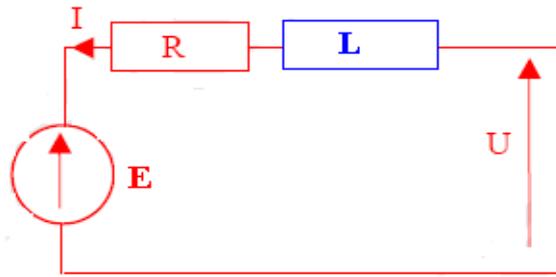
$$E = \frac{1}{2\pi} \frac{p}{a} N \Phi \Omega \quad (1.9)$$

$$E = K_E \Omega \quad (1.10)$$

On peut mettre cette relation sous la forme :

Avec  $K_E = K_t$

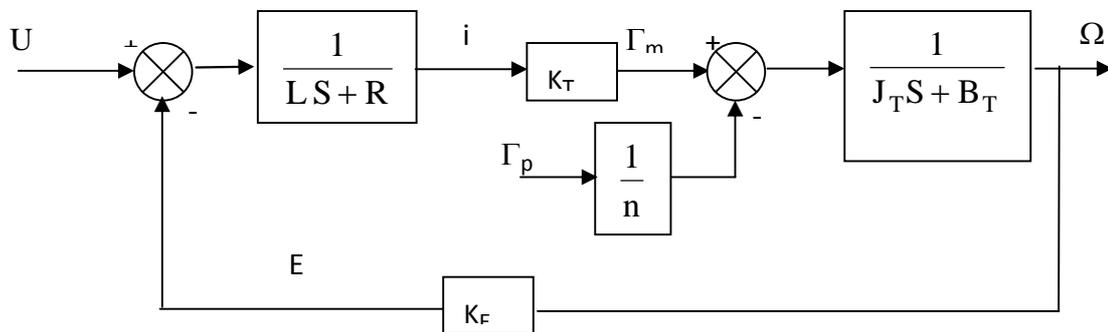
On peut représenter le moteur par un schéma électrique équivalent (Figure 1.10).



**Figure 1.10 :** Schéma équivalent.

#### 1.6.2.4 Fonction de transfert du moteur à courant continu

Afin d'étudier le comportement dynamique d'un moteur à courant continu, il est essentiel de déterminer sa fonction de transfert ou sa transmittance, elle définit le transfert entre la vitesse de rotation  $\Omega$  et la tension de l'induit  $u$ , d'après les relations (1.3), (1.4), (1.7) et (1.9) obtenus précédemment nous avons le schéma suivant :



**Figure 1.11 :** Fonction de transfert d'un moteur à courant continu.

On peut remarquer que le terme de  $\Gamma_s n$  n'est pas représenté car il sera considéré comme une perturbation incluse dans le couple de perturbation externe. Les paramètres du moteur utilisés sont identifiés dans le chapitre 2.

## 1.7. Description des capteurs

Pour la réalisation des fonctions mécaniques telles qu'un déplacement ou un positionnement, il est nécessaire de disposer d'un capteur qui permet de localiser la position absolue de chaque axe. Plusieurs technologies existent : on rencontre des capteurs

incrémentaux et potentiométrique. Pour notre plateforme, des capteurs de position potentiométriques (Figure 1.12) sont utilisés.



**Figure 1.12 :** *Le capteur du robot ROB3.*

Le tableau 1.2 donne la relation entre les angles et les tensions correspondante données par le potentiomètre.

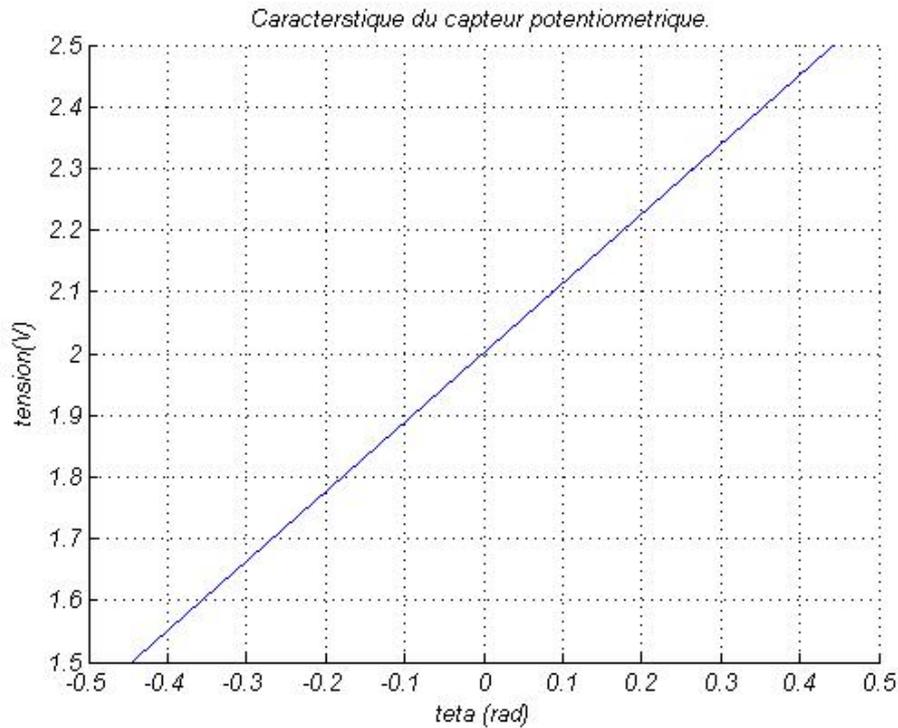
Les Articulations	Axe	Limite articulaire	Résolution (Volts)
Axe 1 : Base	Q0	+80° ... 0° ... -80°	1.5 ... 2.5
Axe 2 : EpauLe	Q1	+50° ... 0° ... -50°	1.5 ... 2.5
Axe 3 : Coude	Q2	0 ... +100°	1.5 ... 2.5
Axe 4 : Poignet	Q3	+100° ... 0° ... -100°	1.5 ... 2.5
Axe 5 : Orientation	Q4	+100° ... 0° ... -100°	1.5 ... 2.5
Outil (Pince)		0 ... 60 mm	1.5 ... 2.5

**Tableau 1.2 :** *Caractéristique des capteurs du robot ROB3.*

### 1.7.1. Caractéristique du capteur

Après des prélèvements expérimentales sur le capteur potentiométrique on trace la courbe qui donne la relation entre la tension au borne du potentiomètre et l'angle correspondant, on remarque bien que cette est linéaire, cette relation est donnée par la formule cité ci-dessous. Illustrée par la courbe de la figure 1.13.

$$V(\text{volts}) = 1.125\theta(\text{rad}) + 2 \quad (1.11)$$



**Figure 1.13** : Caractéristique du capteur potentiométrique.

## 1.8. Conclusion

Nous avons présenté dans cette partie les différentes caractéristiques de la plateforme étudiée en donnant le modèle qui représente un moteur à courant continu, ainsi que les caractéristiques des capteurs utilisés, ceci sera pris en compte pour le dimensionnement du régulateur.

CHAPITRE 2  
*MODELISATION ET  
IDENTIFICATION.*

## 2. Modélisation et identification

### 2.1 Modélisation

En robotique, on associe à tout élément du poste de travail un ou plusieurs repères. Ces repères sont généralement définis de telle sorte que leurs origines correspondent à des directions et à des points privilégiés ayant un rôle fonctionnel lors de l'exécution d'une tâche, telle que la direction d'insertion, le centre de gravité, l'orientation de l'articulation ou extrémité de l'outil [5].

La notion de transformation de repère est fondamentale, elle permet :

- D'exprimer la situation des différents corps du robot les uns par rapport aux autres ;
- De spécifier les situations que doit prendre le repère associé à l'organe terminal du robot pour réaliser une tâche donnée ainsi que les vitesses correspondantes.

Nous présentons dans ce chapitre une notation qui permet de décrire de façon homogène les différents systèmes de coordonnées.

#### Représentation des transformations et des mouvements rigides

Ce chapitre contient des rappels mathématiques sur les transformations rigides et l'orientation des corps dans l'espace. [6], [7] et [8]

##### 2.1.1. Notations et définitions

###### 2.1.1.1. Points

Soit un repère orthonormé cartésien noté  $\mathcal{R} = (O, \underline{x}, \underline{y}, \underline{z})$  selon la convention de Gibbs. La position d'un point M est donnée par un triplet de coordonnées. On choisit par commodité de représenter ces coordonnées par un vecteur  $m$  de  $\mathbb{R}^3$ , sous la forme d'une matrice colonne :

$$m = \begin{pmatrix} m_x \\ m_y \\ m_z \end{pmatrix} \quad (2.1)$$

Le mouvement du point est la courbe paramétrée  $m(t)$  de  $\mathbb{R}^3$  donnant sa position au cours du temps. Le support du mouvement (la courbe non paramétrée) est la trajectoire du point.

### 2.1.1.2. Solides

Un solide  $S$  est dit indéformable si, pour toute paire de points de ce solide de coordonnées  $m$  et  $n$ ,  $\|m(t) - n(t)\| = \|m(0) - n(0)\| = \text{constant}$  au cours du temps. Par la suite, les solides seront tous considérés comme indéformables. Le mouvement rigide d'un solide  $S$ , considéré comme un sous-ensemble de  $\mathbb{R}^3$  est le mouvement de chacun de ces points. Il est ainsi représenté par une famille d'applications  $g(t) : S \longrightarrow \mathbb{R}^3$ . La situation d'un solide  $S$  est donnée par la position et l'orientation dans d'un repère  $\mathbb{R}$  lié à ce solide.

### 2.1.1.3. Transformations rigides

On appelle transformation rigide le résultat d'un mouvement rigide amenant le solide d'une situation initiale à une situation finale. Une transformation rigide est représentée par une application unique  $g : S \longrightarrow \mathbb{R}^3$  qui transforme les coordonnées des points du solide  $S$  de leur position initiale vers leur position finale.

Une application  $g$  représentera une transformation rigide si elle conserve à la fois les distances et l'orientation. Soit le vecteur  $v = m - n$  ou  $m$  et  $n$  représentent les coordonnées de deux points et :

$$g_* : \mathbb{R}^3 \longrightarrow \mathbb{R}^3 \text{ telle que } g_*(v) = g(m) - g(n).$$

Conformément à la définition précédente, une application  $g$  représente une transformation rigide si et seulement si :

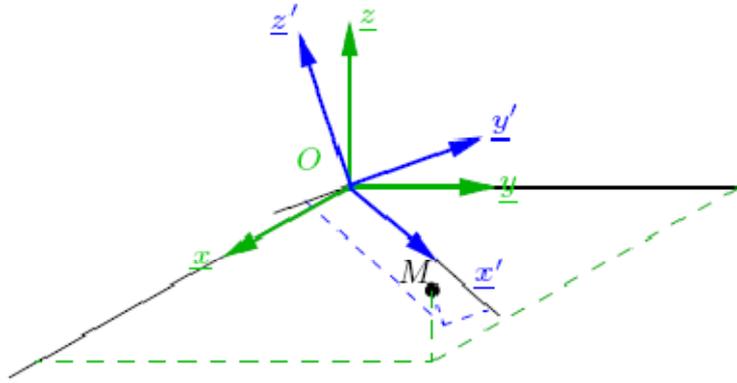
1.  $\|g(m) - g(n)\| = \|m - n\|, \forall m, n \in \mathbb{R}^3$
2.  $g_*(v \times w) = g_*(v) * g_*(w), \forall v, w \in \mathbb{R}^3$

Cela signifie notamment qu'un repère orthonormé direct reste orthonormé direct par application d'une transformation rigide.

### 2.1.1.4. Rotations

#### 2.1.1.4.1. Matrice de rotation

Soit deux repères orthonormés directs  $\mathcal{R} = (O, \underline{x}, \underline{y}, \underline{z})$  et  $\mathcal{R}' = (O, \underline{x}', \underline{y}', \underline{z}')$  partageant le même origine  $O$ , conformément à la figure 2.1.



**Figure 2.1 :** *Changement de base des coordonnées d'un point.*

Soit  $x', y', z$  les coordonnées des vecteurs de la base  $\mathcal{R}'$  exprimées dans  $\mathcal{R}$ . Alors :

$$x' = \begin{pmatrix} \underline{x} \cdot \underline{x}' \\ \underline{x} \cdot \underline{y}' \\ \underline{x} \cdot \underline{z}' \end{pmatrix}, y' = \begin{pmatrix} \underline{y} \cdot \underline{x}' \\ \underline{y} \cdot \underline{y}' \\ \underline{y} \cdot \underline{z}' \end{pmatrix}, z' = \begin{pmatrix} \underline{z} \cdot \underline{x}' \\ \underline{z} \cdot \underline{y}' \\ \underline{z} \cdot \underline{z}' \end{pmatrix}. \quad (2.2)$$

La matrice  $\mathcal{R} = (x, y, z)$  de dimension  $3 \times 3$  est appelée matrice de rotation ou matrice de passage ou encore matrice de changement de base du repère  $\mathcal{R}$  vers le repère

$\mathcal{R}'$ . En effet, selon le but recherché, on pourra l'utiliser pour rendre compte :

- du changement de base des coordonnées d'un point ou d'un vecteur entre les repères  $\mathcal{R}$  et  $\mathcal{R}'$ .
- de la rotation d'un solide liée à un repère orthonormé, initialement en  $\mathcal{R}$ , et déplacé en  $\mathcal{R}'$  par la rotation autour de O.

Nous allons illustrer ceci dans les paragraphes suivants.

#### 2.1.1.4.2. Rotation d'un point appartenant à un solide

Le repère  $\mathcal{R}$  est un repère fixe cartésien orthonormé. Soient  $\underline{m} = (m_x, m_y, m_z)^T$  et  $\underline{m}' = (m'_x, m'_y, m'_z)^T$ . Les coordonnées d'un point M respectivement dans  $\mathcal{R}$  et  $\mathcal{R}'$  (voir figure 1.1). Les coordonnées des vecteurs de la base  $\mathcal{R}'$  exprimées dans  $\mathcal{R}$  étant (toujours) notées  $x', y'$  et  $z'$ , les coordonnées de M dans  $\mathcal{R}$  sont données par :

$$\underline{m} = m'_x \cdot \underline{x}' + m'_y \cdot \underline{y}' + m'_z \cdot \underline{z}' \quad (2.3)$$

soit encore sous forme matricielle :

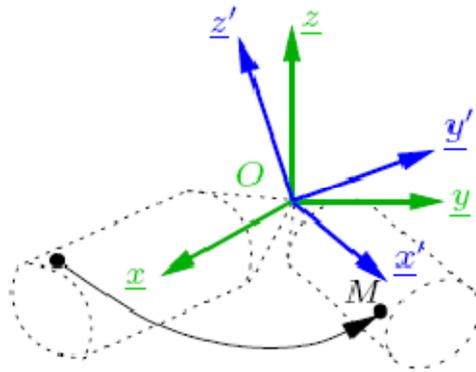
$$\underline{m} = (x', y', z') \begin{pmatrix} m'_x \\ m'_y \\ m'_z \end{pmatrix}. \quad (2.4)$$

C'est-à-dire :

$$m = R.m'. \quad (2.5)$$

La relation (2.4) rend compte du changement de base des coordonnées d'un point.

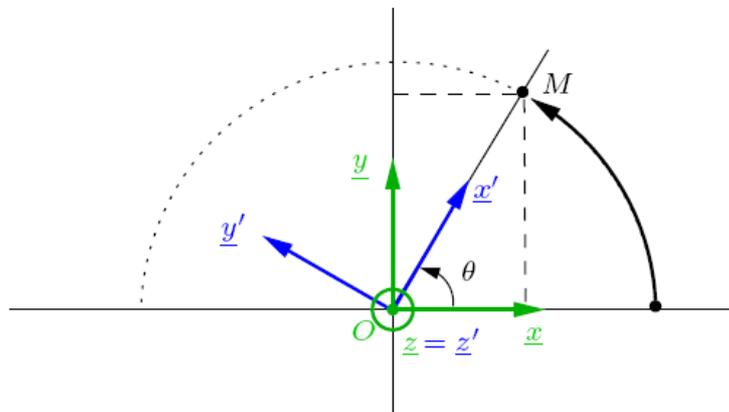
On peut aussi en faire une analyse en terme de rotation si l'on considère que  $M$  est un point d'un solide  $S$  (en pointillés sur la figure 2.2) ayant effectué, autour de  $O$ , une rotation caractérisée par la matrice  $R$ . Alors  $m_0$  représentent les coordonnées initiales de  $M$  dans  $R$  et  $m$  ses coordonnées finales, toujours dans  $R$ , une fois la rotation effectuée.



**Figure 2.2 :** Rotation d'un solide autour d'un point fixe.

#### 2.1.1.4.3. Exemple

On considère la rotation d'un point  $M$  de coordonnées initiales  $(0 \ 0 \ 0)^T$  dans le repère  $R$ , comme cela est représenté à la figure 2.3. les coordonnées du point résultant dans le repère  $R$  sont  $(\sqrt{3} \ 0 \ 1)^T$ .



**Figure 2.3 :** Exemple de rotation plane de la rotation de centre  $O$  et d'angle  $\theta$ , autour de  $z$ .

#### 2.1.1.4.4. Rotation d'un vecteur

Bien évidemment, la rotation s'applique aussi aux vecteurs. Les coordonnées d'un vecteur sont en effet la différence des coordonnées de deux points de  $\mathbb{R}^3$ . On peut ainsi appliquer la rotation à un vecteur de coordonnées  $v = m - n$  dans  $\mathbb{R}^3$ .

$$m - n = Rm' - Rn' = R(m' - n'). \quad (2.6)$$

Soit, en posant  $v = m' - n'$  :

$$v = Rv'. \quad (2.7)$$

#### 2.1.1.4.5. Propriétés des rotations

La matrice de rotation  $R$ , par définition, est constituée de colonnes orthonormales et donc :

$$R^T R = I. \quad (2.8)$$

Par ailleurs, on montre que :

$$\det R = (x')^T (y' \times z') = (x')^T x' = 1. \quad (2.9)$$

La matrice de rotation  $R$  est donc un élément de l'ensemble :

$$SO(3) = \{R \in \mathbb{R}^{3 \times 3} / R R^T = I, \det R = 1.\} \quad (2.10)$$

Appelé groupe spécial orthogonal. Cet ensemble, muni de l'opération multiplication matricielle, est un groupe sur  $\mathbb{R}^{3 \times 3}$ . Ceci a des implications pratiques :

1. La combinaison de deux rotations  $R_1$  et  $R_2$  est la rotation  $R_1 R_2$ .
2. Il existe un unique élément neutre, qui est la matrice identité<sup>1</sup> d'ordre 3.
3. Pour une matrice de rotation  $R$  donnée, il existe un unique inverse  $R^{-1} = R^T$ .

Par ailleurs, une rotation est une transformation rigide car :

1.  $\|Rm - Rn\| = \|m - n\|, \forall m, n \in \mathbb{R}^3$ .
2.  $R(v \times w) = Rv \times Rw, \forall v, w \in \mathbb{R}^3$ .

#### 2.1.1.4.6. Représentation de l'orientation d'un solide dans l'espace

La donnée d'une base attachée à un solide  $S$  en rotation détermine de manière unique son orientation dans l'espace. Différentes représentations peuvent être utilisées pour cela, avec pour chacune avantages et inconvénients.

#### 2.1.1.4.6.1. Matrice de rotation et cosinus directeurs

On considère la rotation d'un repère  $R$  vers un repère  $R'$ . La matrice de rotation  $R$  est une matrice de dimension  $3 \times 3$  à valeurs dans  $R$  :

$$R = \begin{pmatrix} x_x & y_x & z_x \\ x_y & y_y & z_y \\ x_z & y_z & z_z \end{pmatrix}. \quad (2.11)$$

Les éléments de cette matrice sont appelés cosinus directeurs car ils représentent les coordonnées des trois vecteurs de la base  $R$  exprimés dans  $R'$ . Par définition, les colonnes de  $R$  sont orthogonales entre elles et par conséquent la connaissance de deux colonnes suffit. En robotique, on omet ainsi souvent la seconde colonne :

$$R = \begin{pmatrix} x_x & * & z_x \\ x_y & * & z_y \\ x_z & * & z_z \end{pmatrix}. \quad (2.12)$$

Les six paramètres restant sont appelés cosinus directeurs incomplets. Par ailleurs les colonnes de  $R$  formant une base orthonormée :

– les deux colonnes restantes sont orthogonales et donc :

$$x_x z_x + x_y z_y + x_z z_z = 0. \quad (2.13)$$

– les deux colonnes restantes sont de norme unité et donc :

$$x_x^2 + x_y^2 + x_z^2 = 1, \quad (2.14)$$

$$z_x^2 + z_y^2 + z_z^2 = 1. \quad (2.15)$$

Les six cosinus directeurs incomplets sont ainsi liés entre eux par trois relations. Malheureusement, ces relations non linéaires ne permettent pas de fournir trois paramètres de manière unique, mais une collection de solutions : il faut donc conserver les six paramètres ou utiliser un autre mode de représentation de l'orientation.

#### 2.1.1.4.7. Représentation angle/axe

##### 2.1.1.4.7.1. Vecteur rotation

Pour toute rotation, on peut trouver une unique paire  $(\underline{e}, \varepsilon)$  où  $\underline{e}$  représente de vecteur directeur de l'axe de la rotation, de norme unité, et  $\varepsilon$  l'angle de rotation en radians. Pour que la représentation soit unique, il faut choisir  $-\pi < \varepsilon \leq \pi$  et  $e$  de coordonnées  $e = [e_x \ e_y \ e_z]^T$  tel que  $e_x + e_y + e_z > 0$ . On appelle vecteur rotation le vecteur  $\underline{\varepsilon} = \varepsilon \underline{e}$ . On peut bien évidemment établir la correspondance entre cette représentation à quatre paramètres ( $\varepsilon$  et les trois composantes de  $\underline{e}$ ) et les cosinus directeurs. On montre que :

$$\varepsilon = \arccos\left(\frac{x_x + y_y + z_z - 1}{2}\right). \quad (2.16)$$

$$\text{et } e = \frac{1}{2\sin\varepsilon} \begin{pmatrix} y_z - z_y \\ z_x - x_z \\ x_y - y_x \end{pmatrix}. \quad (2.17)$$

Il existe de très nombreux cas particuliers [7].

#### 2.1.1.4.7.2. Coordonnées exponentielles

La vitesse d'un point  $M$  de coordonnées  $m$  en rotation autour d'un axe  $\underline{e}$  à vitesse constante unité s'écrit :

$$\frac{dm(t)}{dt} = e \times m(t) = \tilde{e}m(t). \quad (2.18)$$

$$\text{avec } \tilde{e} = \begin{pmatrix} 0 & -e_z & e_y \\ e_z & 0 & -e_x \\ -e_y & e_x & 0 \end{pmatrix}. \quad (2.19)$$

La matrice du tenseur antisymétrique associé au vecteur de coordonnées  $e$ . Alors :

$$m(t) = \exp(\tilde{e}t) m(0). \quad (2.20)$$

Avec  $m(0)$  les coordonnées de  $M$  à  $t = 0$  et  $\exp(\tilde{e}t)$  la matrice exponentielle :

$$\exp(\tilde{e}t) = I + \tilde{e}t + \frac{(\tilde{e}t)^2}{2} + \dots \quad (2.21)$$

Cette matrice représente la rotation d'axe  $\underline{e}$ . Si on l'évalue en  $t = \varepsilon$ , on obtient une représentation de la rotation d'axe  $\underline{e}$  et d'angle  $\varepsilon$  :

$$R(\underline{e}, \varepsilon) = \exp(\tilde{e}\varepsilon). \quad (2.22)$$

La matrice exponentielle  $\exp(\tilde{e}\varepsilon)$  se calcule en appliquant la formule de Rodrigues :

$$\exp(\tilde{e}\varepsilon) = I + \tilde{e} \sin(\varepsilon) + \tilde{e}^2 (1 - \cos(\varepsilon)). \quad (2.23)$$

#### 2.1.1.4.7.3. Angles d'Euler classiques

Les angles d'Euler classiques permettent de décrire l'orientation d'un solide par trois rotations successives. Conformément à la figure 2.4, il s'agit des rotations  $R(\underline{z}, \psi)$ ,  $R(\underline{x}_\psi, \theta)$  et enfin  $R(\underline{z}_\theta, \varphi)$  ou  $\underline{x}_\psi$  et  $\underline{z}_\theta$  sont déduits des vecteurs du repère de base conformément à la figure 2.4. Les angles  $\psi$ ,  $\theta$  et  $\varphi$  sont connus respectivement sous les termes de précession, nutation et rotation propre. Chaque nouvelle rotation étant effectuée par rapport à un repère ayant tourné :

$$R = R(\underline{z}, \psi)R(\underline{x}_\psi, \theta)R(\underline{z}_\theta, \varphi). \quad (2.24)$$

Sous forme développée :

$$R = \begin{pmatrix} \cos\psi & -\sin\psi & 0 \\ \sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta \\ 0 & \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} \cos\varphi & -\sin\varphi & 0 \\ \sin\varphi & \cos\varphi & 0 \\ 0 & 0 & 1 \end{pmatrix}. \quad (2.25)$$

Soit finalement :

$$R = \begin{pmatrix} \cos\psi\cos\varphi - \sin\psi\cos\theta\sin\varphi & -\cos\psi\sin\varphi - \sin\psi\cos\theta\cos\varphi & \sin\psi\sin\theta \\ \sin\psi\cos\varphi + \cos\psi\cos\theta\sin\varphi & -\sin\psi\sin\varphi + \cos\psi\cos\theta\cos\varphi & -\cos\psi\sin\theta \\ \sin\theta\sin\varphi & \sin\theta\cos\varphi & \cos\theta \end{pmatrix} \quad (2.26)$$

La transformation inverse permet d'établir les angles d'Euler à partir des cosinus directeurs<sup>2</sup> :

- Si  $z_z \neq \mp 1$

$$\psi = \text{atan}(z_x, -z_y). \quad (2.27)$$

$$\theta = \text{acos}z_z \quad (2.28)$$

$$\varphi = \text{atan}(x_x, -y_z) \quad (2.29)$$

- Si  $z_z = \mp 1$

$$\theta = \pi(1 - z_z)/2. \quad (2.30)$$

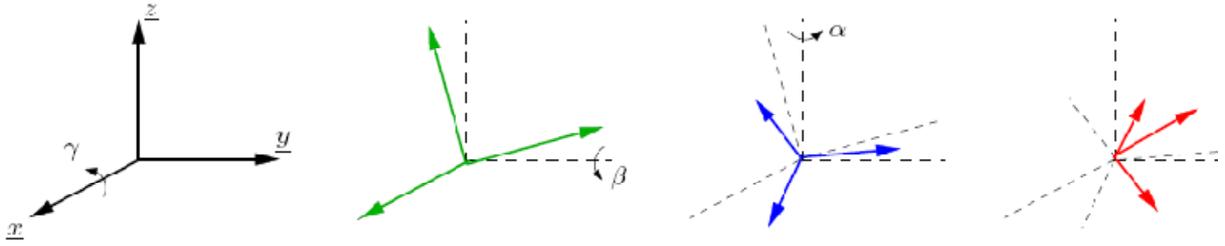
$$\psi + z_z\varphi = \text{atan}(y_x, x_x) \quad (2.31)$$

Et donc  $\psi$  et  $\varphi$  sont indéterminés.

#### 2.1.1.4.7.4. Angles roulis, tangage et lacet

Pour cette représentation, la littérature donne de nombreuses définitions [1], [7], [8] et [9]. Ces angles, très utilisées par les anglo-saxons et donc par les industriels, portent les noms de roll, pitch et yaw en anglais. Il s'agit en fait d'angles d'Euler non classiques en cela que les rotations s'effectuent autour d'axes fixes.

Nous adoptons la définition de [8] : les rotations successives, conformément à la figure 2.5, sont  $R(\underline{x}, \gamma)$ ,  $R(\underline{y}, \beta)$  puis  $R(\underline{z}, \alpha)$  Les angles  $\gamma$ ,  $\beta$  et  $\alpha$  sont respectivement désignés sous les noms d'angles de roulis, tangage et lacet. Chaque nouvelle



**Figure 2.5** : Rotations successives dans le paramétrage par les angles de roulis, tangage et lacet.

Rotation étant effectuée par rapport à un axe du repère fixe  $R$  :

$$R = R(\underline{z}, \alpha)R(\underline{y}, \beta)R(\underline{x}, \gamma).$$

Sous forme développée :

$$R = \begin{pmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\gamma & -\sin\gamma \\ 0 & \sin\gamma & \cos\gamma \end{pmatrix}, \quad (2.32)$$

Soit finalement :

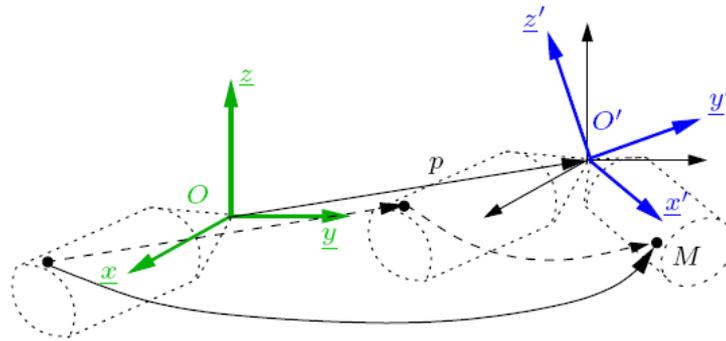
$$R = \begin{pmatrix} \cos\alpha\cos\beta & -\sin\alpha\sin\gamma - \cos\alpha\sin\beta\cos\gamma & \sin\alpha\sin\gamma + \cos\alpha\sin\beta\cos\gamma \\ \sin\alpha\cos\beta & \cos\alpha\sin\gamma + \sin\alpha\sin\beta\cos\gamma & -\cos\alpha\sin\gamma + \sin\alpha\sin\beta\cos\gamma \\ -\sin\beta & \cos\beta\sin\gamma & \cos\beta\cos\gamma \end{pmatrix}. \quad (2.33)$$

### 2.1.2. Transformation Homogène

Faisons subir une transformation quelconque au repère  $\mathbf{R}_i$ , cette transformation qui l'amène sur le repère  $\mathbf{R}_j$ , (figure 2.6). [5] Cette transformation est définie par la matrice  ${}^jT_i$  appelée matrice de transformation homogène, de dimension (4x4), telle que :

$${}^i T_j = \begin{bmatrix} {}^j s_i & {}^j n_i & {}^j a_i & {}^j P_i \\ s_x & n_x & a_x & P_x \\ s_y & n_y & a_y & P_y \\ s_z & n_z & a_z & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.34)$$

Où  ${}^j s_i$ ,  ${}^j n_i$  et  ${}^j a_i$  désignent respectivement les vecteurs unitaires suivant les axes  $\mathbf{x}_i$ ,  $\mathbf{y}_i$  et  $\mathbf{z}_i$  du repère  $\mathbf{R}_j$  exprimés dans le repère  $\mathbf{R}_i$  et où  ${}^j P_i$  est le vecteur exprimant l'origine du repère  $\mathbf{R}_j$  dans le repère  $\mathbf{R}_i$ .



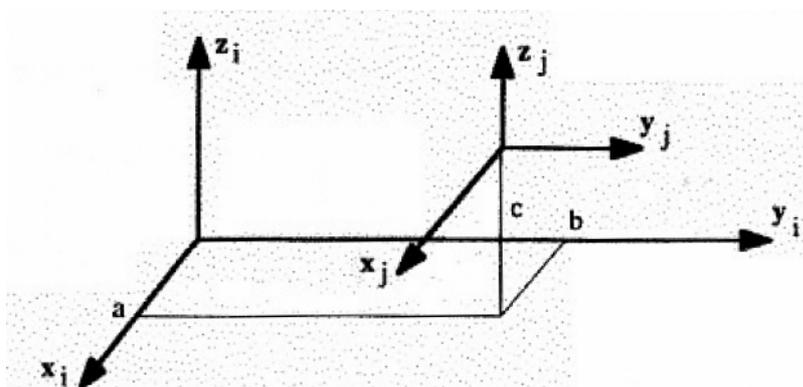
**Figure 2.6 :** Transformation des repères.

Par abus de notation la matrice de transformation homogène peut être exprimée de la manière suivante :

**2.1.2.1. Matrice de transformation d'une translation pure**

Soit cette transformation  $\text{Trans}(a,b,c)$ , où  $a$ ,  $b$  et  $c$  désigne les composantes de la translation le long de  $x$ ,  $y$  et  $z$  respectivement. [5]

L'orientation étant conservée dans cette transformation,  $\text{Trans}(a,b,c)$  a pour expression (figure 2.7) :



**Figure 2.7 :** Transformation d'une translation pure.

### 2.1.2.2. Matrice de transformation de rotation autour des axes principaux

#### 2.1.2.2.1. Transformation correspondante à une rotation $\theta$ autour de l'axe x

Soit la rotation  $\mathbf{Rot}(x,\theta)$  (figure 2.8). On déduit les composantes des vecteurs unitaires  ${}^i\mathbf{s}_j$ ,  ${}^i\mathbf{n}_j$  et  ${}^i\mathbf{a}_j$  portées respectivement par les axes  $\mathbf{x}_i$ ,  $\mathbf{y}_i$  et  $\mathbf{z}_i$  du repère  $\mathbf{R}_j$  exprimée dans le repère  $\mathbf{R}_i$ .

Si on note  $S\theta$  et  $C\theta$  les sinus et cosinus de  $\theta$  respectivement nous obtenons :

$\mathbf{rot}(x,\theta)$  désigne la matrice d'orientation de dimension (3x3).

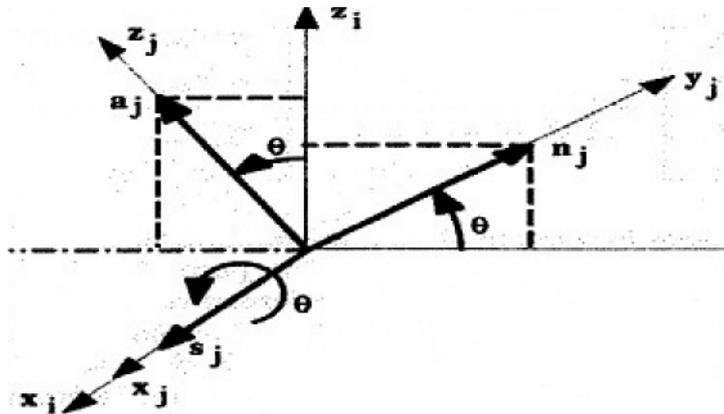


Figure 2.8 : Transformation de rotation autour de l'axe x.

#### 2.1.2.2.2. Matrice de transformation correspondante à une rotation $\theta$ autour de l'axe y :

Avec un raisonnement analogue, nous obtenons

### 2.1.2.2.3. Matrice de transformation correspondante à une rotation °autour de l'axe z

Avec un raisonnement analogue, nous obtenons

$${}^i T_j = Rot(z, \theta) = \begin{bmatrix} C\theta & -S\theta & 0 & 0 \\ S\theta & C\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} rot(z, \theta) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.39)$$

### 2.1.2.3. Propriétés des matrices de transformation homogène

Les matrices de transformation homogène possèdent certaines propriétés qui sont les suivantes :

Une matrice de transformation peut se mettre, d'après la relation (2.36), sous la forme :

$$T = \begin{bmatrix} s_x & n_x & a_x & P_x \\ s_y & n_y & a_y & P_y \\ s_z & n_z & a_z & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} A & P \\ 0 & 1 \end{bmatrix}. \quad (2.40)$$

La matrice  $\mathbf{A}$  représente la rotation alors que la matrice colonne  $\mathbf{P}$  représente la translation. Pour une translation pure,  $\mathbf{A}=\mathbf{I}_3$  ( $\mathbf{I}_3$  représente la matrice identité d'ordre 3), tandis que pour une rotation pure,  $\mathbf{P}=0$ .

- b) La matrice  $\mathbf{A}$  est orthogonale, c à d  $\mathbf{A}^{-1} = \mathbf{A}^T$
- c) L'inverse de la matrice de transformation représenté par la relation (2.41) peut être calculé par:

$$T^{-1} = \begin{bmatrix} A^T & -s^T \cdot P \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} A^T & -A^T P \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.41)$$

L'inverse d'une transformation  ${}^j T_i$  est  $({}^i T_j)^{-1} = {}^i T_j$

- d) Si un repère  $\mathbf{R}_0$  a subi k transformation consécutives (figure 2.9) et si chaque transformation  $i$ , ( $i=1, \dots, k$ ), est définie par rapport au repère courant  $\mathbf{R}_{i-1}$ , alors la transformation  ${}^0 T_k$  peut être déduite de la composition des multiplications à droite de ces transformations :

$${}^0 T_k = {}^0 T_1 {}^1 T_2 {}^2 T_3 \dots {}^{k-1} T_k \quad (2.42)$$

e) Si un repère  $R_j$ , définie dans le repère  $R_i$  par la transformation  ${}^i T_j$ , subit une transformation  $T$  exprimée dans le repère  $R_i$ , le repère  $R_j$  se transforme en  $R_{j'}$ , avec  $A$  partir des propriétés e et d, on déduit que :

- une multiplication à droite de la transformation  ${}^i T_j$  signifie que la transformation est faite par rapport au repère courant  $R_j$  ;

$${}^i T_{j'} = T {}^i T_j \text{ (figure 2.10).}$$

- une multiplication à gauche signifie que la transformation est faite par rapport au repère de référence  $R_i$ .

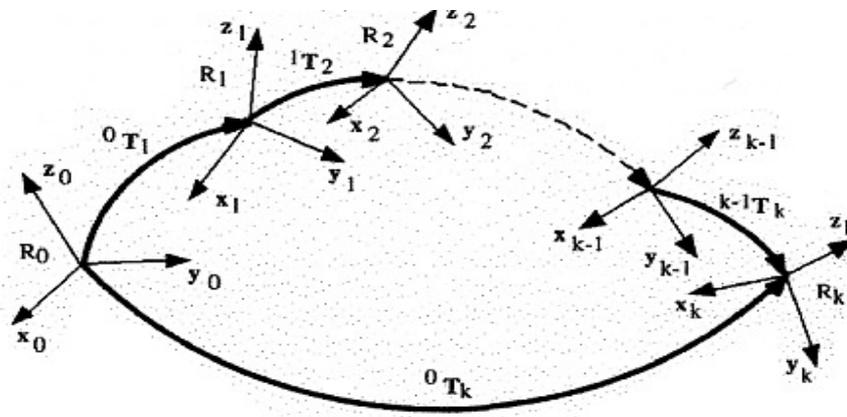


Figure 2.9 : Composition des transformations: multiplication à droite.

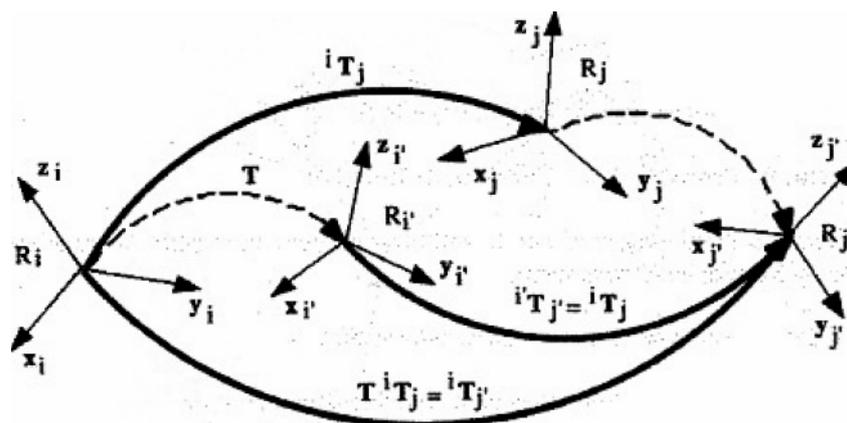


Figure 2.10 : Composition des transformations: multiplication à gauche.

### 2.1.3. Modélisation des robots manipulateurs

#### 2.1.3.1. Méthodes et notations

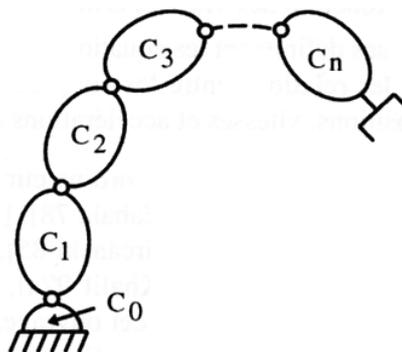
La modélisation des robots de façon systématique exige une méthode adéquate pour la description de leur morphologie. Plusieurs méthodes et notation ont été proposées ; la plus répandue est celle de Denavit-Hartenberg qui permet la description homogène, et avec un nombre minimum de paramètres. [5]

#### 2.1.3.2 Description de la géométrie des robots à structure ouverte simple

Ce paragraphe présente la méthodologie à suivre pour décrire les robots à structure ouverte simple.

Une structure ouverte simple est composée de  $n+1$  corps notés  $C_0, \dots, C_n$ , et de  $n$  articulation. Le corps  $C_0$  désigne la base du robot et le corps  $C_n$  le corps qui porte l'organe terminal. L'articulation  $j$  connecte le corps  $C_j$  au corps  $C_{j-1}$  (figure 2.11) .la méthode de description est fondée sur les règles et convention suivantes :

- les corps sont supposés parfaitement rigides. Ils sont connectés par des articulations considérées comme idéales (pas de jeu mécanique, pas d'élasticité), soit rotoïdes, soit prismatiques ;
- le repère  $R_j$  est lié au corps  $C_j$  ;
- la variable de l'articulation  $j$  est notée  $q_j$ .



**Figure 2.11:** Robot à structure ouverte simple.

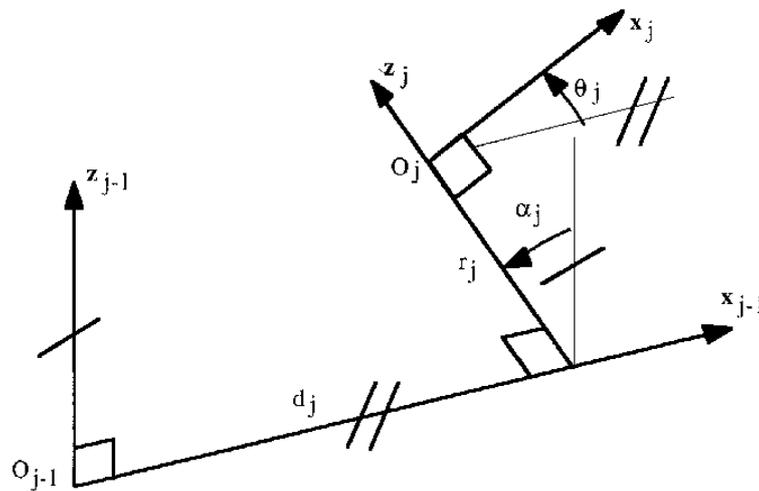
Le corps  $R_j$ , fixé au corps  $C_j$  est défini de sorte que :

- l'axe  $z_j$  est porté par l'axe de l'articulation  $j$  ;

- l'axe  $x_j$  est porté par la perpendiculaire commune aux axes  $z_j$  et  $z_{j+1}$ .

Le passage du repère  $R_{j-1}$  au repère  $R_j$  s'exprime en fonction des quatre paramètres géométriques suivants (figure 2.12) :

- $\alpha_j$  : angle entre les axes  $z_{j-1}$  et  $z_j$  correspondant à une rotation de  $x_j$  ;
- $d_j$  : distance entre  $x_j$  et  $z_{j-1}$  le long de  $z_{j-1}$  ;
- $\theta_j$  : angle entre les axes  $x_{j-1}$  et  $x_j$  correspondant à une rotation de  $z_{j-1}$  ;
- $a_j$  ( $r_j$ ): distance entre  $z_{j-1}$  et  $z_j$  le long de  $x_j$ .



**Figure 2.12 :** Paramètres géométrique dans le cas d'une structure ouverte simple.

La variable articulaire  $q_j$  associée à la  $j^{\text{ème}}$  articulation est soit  $\theta_j$ , soit  $r_j$ , selon que cette articulation est de type rotoïde ou prismatique, ce qui se traduit par la relation.

$$q_j = \bar{\sigma}_j \theta_j + \sigma_j r_j. \quad \text{avec}$$

$$\sigma_j = 0 \text{ si l'articulation } j \text{ est prismatique}$$

$$\sigma_j = 1 \text{ si l'articulation } j \text{ est rotoïde}$$

La matrice de transformation définissant le repère  $R_j$  dans le repère  $R_{j-1}$  est donnée par :

$${}^{j-1}T_j = \begin{bmatrix} \cos(\theta_j) & -\sin(\theta_j) & 0 & d_j \\ \sin(\theta_j)\cos(\alpha_j) & \cos(\alpha_j)\cos(\theta_j) & -\sin(\alpha_j) & -a_j\sin(\alpha_j) \\ \sin(\theta_j)\sin(\alpha_j) & \sin(\alpha_j)\cos(\theta_j) & \cos(\alpha_j) & a_j\cos(\alpha_j) \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (2.43)$$

Le modèle géométrique direct (MGD) est l'ensemble des relations qui permettent d'exprimer la situation de l'organe terminal, c'est à dire les coordonnées opérationnelles du robot, en fonction de ses coordonnées articulaires. Il peut être représenté par la matrice de passage  ${}^0T_n$  :

$${}^0T_n = {}^0T_1(q_1) {}^1T_2(q_2) \dots {}^{n-1}T_n(q_n). \quad (2.44)$$

Le modèle géométrique direct du robot peut aussi être représenté par la relation :

$$X = f(q).$$

$q$  étant le vecteur des variables articulaires tel que :

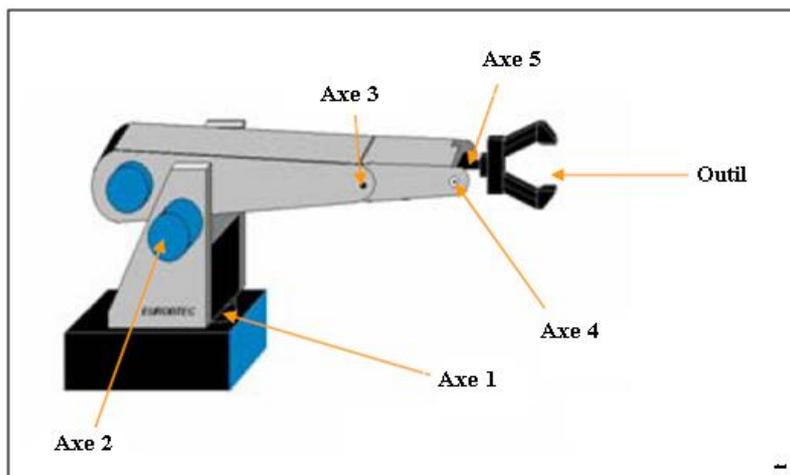
$$q = [q_1 \ q_2 \ \dots \ q_n]^T.$$

Les coordonnées opérationnelles sont définies par :

$$X = [X_1 \ X_2 \ \dots \ X_m]^T.$$

### 2.1.3.3. Application au robot ROB3 :

Le robot ROB3 est un robot possédant 6 degrés de liberté dont 5 sont rotoïdes et la sixième et prismatique, (figure 2.13) il appartient à la catégorie des robots à structure ouverte simple.



**Figure 2.13 :** Les axes du robot ROB3.

En utilisant la méthode citée ci-dessus, les repères sont placés comme il est montré sur la figure 1.2.

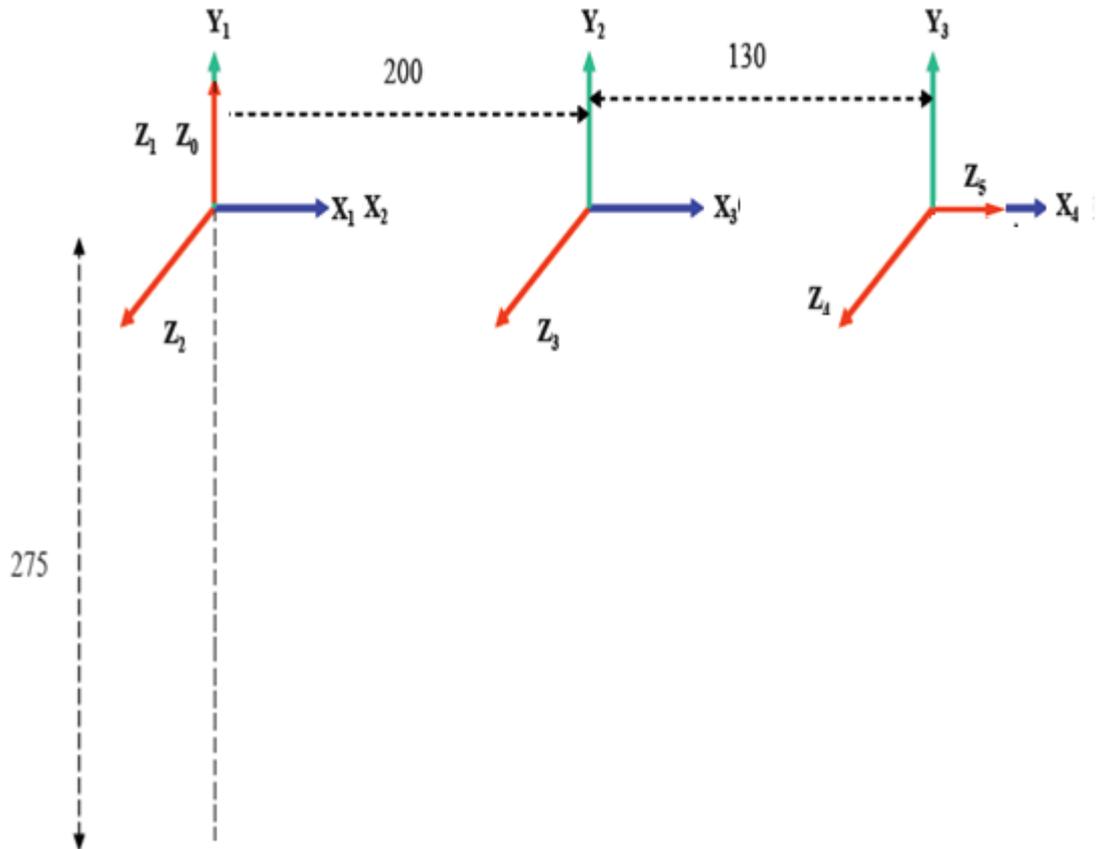


Figure 2.14 : Placement des repères.

	$\theta_i$	$d_i$	$a_i$	$\alpha_i$
Segment 1	$\Theta_1(q_0)$	<b>0</b>	<b>0</b>	<b>0</b>
Segment 2	$\Theta_2(q_1)$	<b>0</b>	<b>0</b>	<b>90</b>
Segment 3	$\Theta_3(q_2)$	<b>0</b>	<b>200</b>	<b>0</b>
Segment 4	$\Theta_4(q_3)$	<b>0</b>	<b>130</b>	<b>0</b>
Segment 1	$\Theta_5(q_4)$	-	-	-

Tableau 2.1 : Paramètres géométriques du ROB3.

Concernant le  $\Theta_5$  on a une rotation suivant l'axe  $y_5$

A partir de ce tableau et compte tenu de la relation (2.43), on déduit les matrices de transformation élémentaire  ${}^{j-1}T_j$  suivante :

$${}^0A_1 = \begin{bmatrix} \cos(\theta_1) & -\sin(\theta_1) & 0 & 0 \\ \sin(\theta_1) & \cos(\theta_1) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (2.45)$$

$${}^1A_2 = \begin{bmatrix} \cos(\theta_2) & -\sin(\theta_2) & 0 & 0 \\ 0 & 0 & -1 & 0 \\ \sin(\theta_2) & \cos(\theta_2) & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (2.46)$$

$${}^2A_3 = \begin{bmatrix} \cos(\theta_3) & -\sin(\theta_3) & 0 & 200 \\ \sin(\theta_3) & \cos(\theta_3) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (2.47)$$

$${}^3A_4 = \begin{bmatrix} \cos(\theta_4) & -\sin(\theta_4) & 0 & 130 \\ \sin(\theta_4) & \cos(\theta_4) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (2.48)$$

$${}^4A_5 = \begin{bmatrix} \cos(\theta_5) & 0 & \sin(\theta_5) & 0 \\ 0 & 1 & 0 & 0 \\ -\sin(\theta_5) & 0 & \cos(\theta_5) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (2.49)$$

Le calcul de  ${}^0T_5$  donne le résultat suivant :

$${}^0A_5 = {}^0A_1 {}^1A_2 {}^2A_3 {}^3A_4 {}^4A_5. \quad (2.50)$$

$${}^0A_1 = \begin{bmatrix} \cos(\theta_1) & -\sin(\theta_1) & 0 & 0 \\ \sin(\theta_1) & \cos(\theta_1) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^0A_2 = \begin{bmatrix} C\theta_1 C\theta_2 & -C\theta_1 S\theta_2 & S\theta_1 & 0 \\ S\theta_1 C\theta_2 & -S\theta_1 S\theta_2 & C\theta_1 & 0 \\ S(\theta_2) & C\theta_2 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^0A_3 = \begin{bmatrix} C\theta_1(C(\theta_2 + \theta_3)) & -C\theta_1(S(\theta_2 + \theta_3)) & S\theta_1 & 200C\theta_1 C\theta_2 \\ S\theta_1(C(\theta_2 + \theta_3)) & -S\theta_1(S(\theta_2 + \theta_3)) & C\theta_1 & 200S\theta_1 C\theta_2 \\ S(\theta_2 + \theta_3) & C(\theta_2 + \theta_3) & 0 & 200S\theta_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^0A_4 = \begin{bmatrix} C\theta_1(C(\theta_2 + \theta_3 + \theta_4)) & -C\theta_1(S(\theta_2 + \theta_3 + \theta_4)) & S\theta_1 & 200C\theta_1 C\theta_2 + 130C\theta_1(C(\theta_2 + \theta_3)) \\ S\theta_1(C(\theta_2 + \theta_3 + \theta_4)) & -S\theta_1(S(\theta_2 + \theta_3 + \theta_4)) & C\theta_1 & 200C\theta_1 C\theta_2 + 130S\theta_1(C(\theta_2 + \theta_3)) \\ S(\theta_2 + \theta_3 + \theta_4) & C(\theta_2 + \theta_3 + \theta_4) & 0 & 200S\theta_2 - 130(\theta_2 + \theta_3) \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^0A_5 = \begin{bmatrix} C\theta_1 C(\theta_2 + \theta_3 + \theta_4) C\theta_5 - S\theta_1 S\theta_5 & -C\theta_1 S(\theta_2 + \theta_3 + \theta_4) & C\theta_1 C(\theta_2 + \theta_3 + \theta_4) S\theta_5 - S\theta_1 C\theta_5 & 200C\theta_1 C\theta_2 + 130C\theta_1(C(\theta_2 + \theta_3)) \\ S\theta_1 C(\theta_2 + \theta_3 + \theta_4) C\theta_5 - C\theta_1 S\theta_5 & -S\theta_1 S(\theta_2 + \theta_3 + \theta_4) & S\theta_1 C(\theta_2 + \theta_3 + \theta_4) S\theta_5 - C\theta_1 C\theta_5 & 200C\theta_1 C\theta_2 + 130S\theta_1(C(\theta_2 + \theta_3)) \\ -S(\theta_2 + \theta_3 + \theta_4) C\theta_5 & -C(\theta_2 + \theta_3 + \theta_4) & S(\theta_2 + \theta_3 + \theta_4) S\theta_5 & 200S\theta_2 - 130(\theta_2 + \theta_3) \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

La relation précédente est la matrice de transformation définissant le repère R5 dans le repère R0.

### 2.1.3.4 Modélisation géométrique inverse du robot ROB3

On cherche à résoudre le système d'équation  $X = f(q)$ ; Le modèle géométrique direct d'un robot permet de calculer les coordonnées opérationnelles donnant la situation de l'organe terminal en fonction des coordonnées articulaires. Le problème inverse consiste à calculer les coordonnées articulaires correspondant à une situation donnée de l'organe terminal.

Lorsqu'elle existe, la forme explicite qui donne toutes les solutions possibles (il y a rarement unicité de solution) constitue ce que l'on appelle le modèle géométrique inverse (MGI). Les méthodes sont nombreuses, pour cela nous nous intéressons à une seule méthode pour cela nous avons basé sur la géométrie pour le calcul.

Remarque :

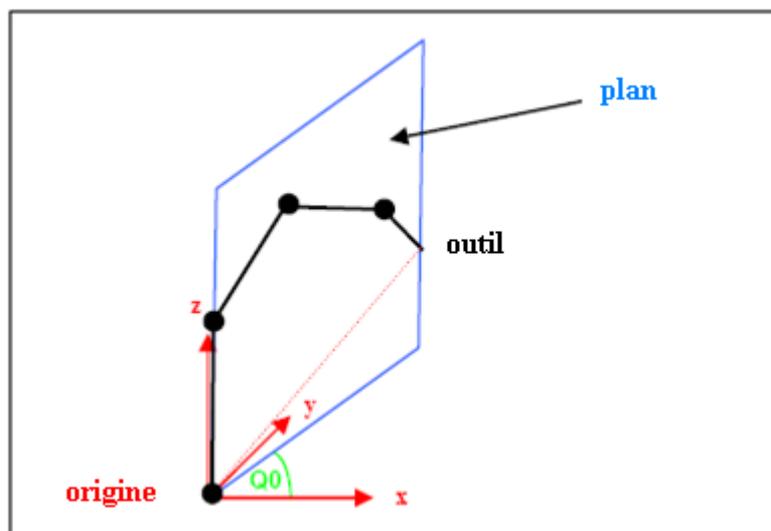


Figure 2.15 : Représentation des membres des robots dans un plan.

D'après la figure 2.15 nous remarquons bien que les articulations 1, 2, 3 sont dans le même plan ainsi que l'articulation Q0 est la seule qui tourne suivant l'axe Z nous remarquons aussi que l'articulation 4 c'est elle qui permet de changer l'orientation de l'outil dans

Ainsi donc Q0 peut être calculé de la façon suivante :

$$Q0 = \arcsin(y/x). \quad (2.51)$$

Aussi nous remarquons que la valeur de x ne peut jamais être à zéro. Ce qui correspond une position d'angle de  $90^\circ$  ou  $-90^\circ$ . Cette limitation n'a pas en effet une signification pratique parce que le domaine de travail pour le robot pour l'articulation Q0 est entre  $-80^\circ$  et  $+80^\circ$ .

Calcul d'angle :

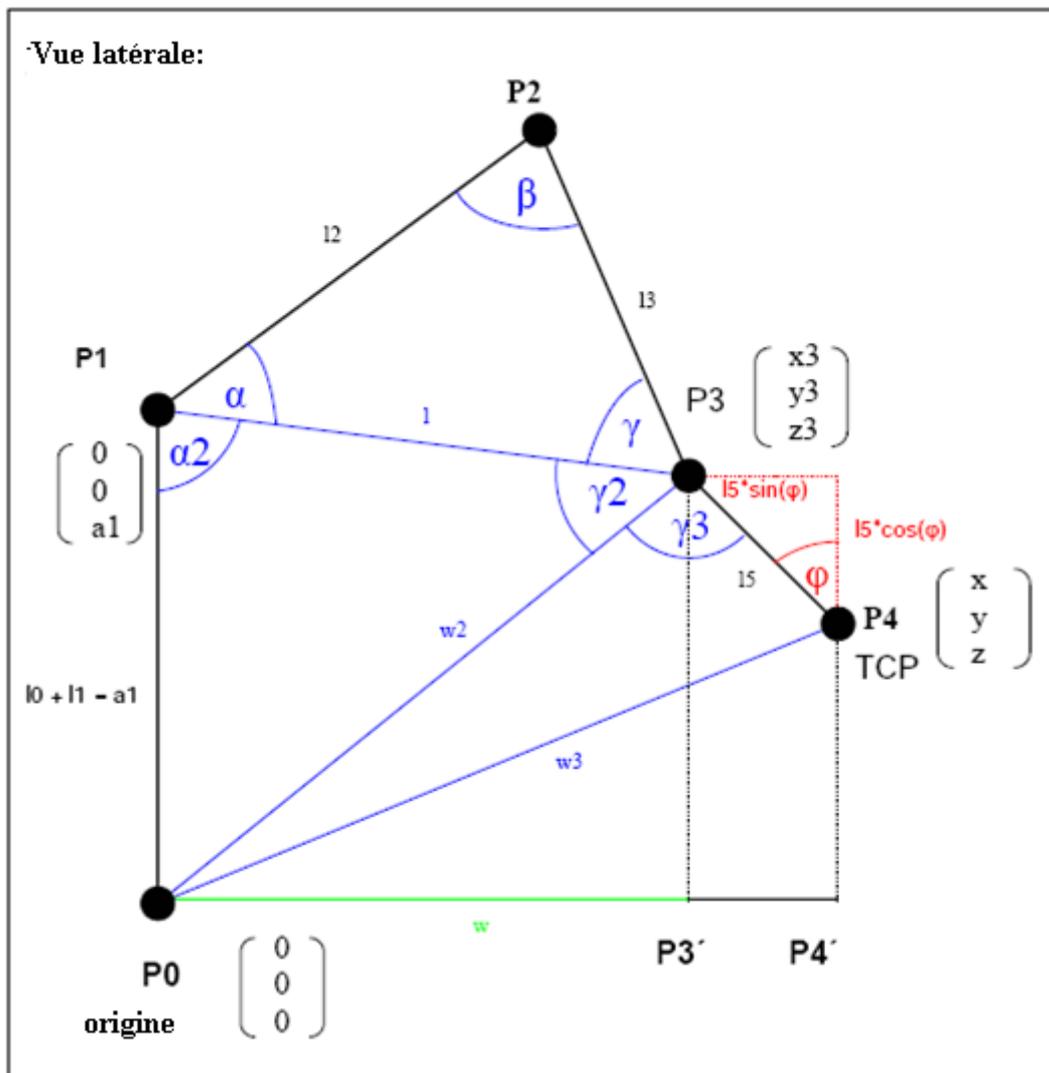


Figure 2.16 : Représentation au niveau des robots.

Calcul de P3 :

$$Q_0 = \arcsin(y/x). \quad (2.52)$$

$$w = \sqrt{x^2 + y^2} - l_5 \sin(\varphi). \quad (2.53)$$

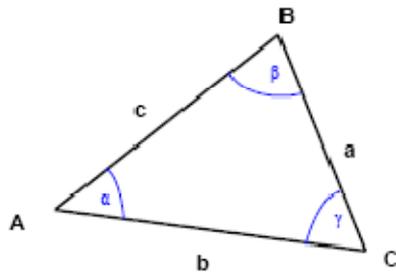
La disposition des coordonnées de P3, P2, P1 :

$$P3 = \begin{pmatrix} x3 \\ y3 \\ z3 \end{pmatrix} = \begin{pmatrix} \sin(Q0) * w \\ \cos(Q0) * w \\ z + l5 * \cos(\varphi) \end{pmatrix}. \quad (2.54)$$

$$P1 = \begin{pmatrix} x1 \\ y1 \\ z1 \end{pmatrix}; \quad P2 = \begin{pmatrix} x2 \\ y2 \\ z2 \end{pmatrix}. \quad (2.55)$$

$$a = \sqrt{(x2 - x1)^2 + (y2 - y1)^2 + (z2 - z1)^2}. \quad (2.56)$$

$$l = \sqrt{(x3 - 0)^2 + (y3 - 0)^2 + (z3 - a1)^2} \quad (2.57)$$



$$a^2 = b^2 + c^2 - 2 * b * c * \cos(\alpha)$$

$$\alpha = \arccos\left(l2^2 + l^2 - \frac{l3^2}{2 * l2 * l}\right). \quad (2.58)$$

$$\beta = \arccos\left(l^2 + l3^2 - \frac{l2^2}{2 * l3 * l}\right). \quad (2.59)$$

$$\gamma = 180^\circ - \alpha - \beta. \quad (2.60)$$

Le calcul la longueur w2

$$w_3 = \sqrt{x_3^2 + y_3^2 + z_3^2} \quad (2.61)$$

Le calcul la longueur w3

$$w_2 = \sqrt{x^2 + y^2 + z^2} \quad (2.62)$$

$$\alpha_2 = \arccos\left(a_1^2 + l^2 - \frac{w_2^2}{2 * a_1 * l}\right). \quad (2.63)$$

$$\gamma_2 = \arccos\left(w_2^2 + l^2 - \frac{a_1^2}{2 * w_2 * l}\right). \quad (2.64)$$

$$\gamma_3 = \arccos\left(l_5^2 + w_2^2 - \frac{w_3^2}{2 * l_5 * w_2}\right). \quad (2.65)$$

Calcul de position des angles :

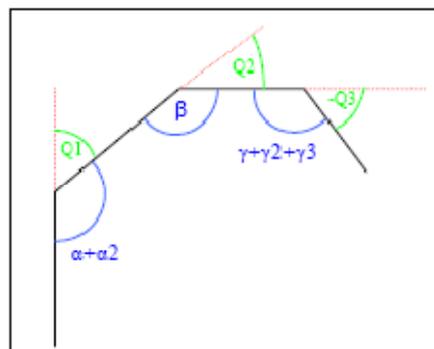
$$Q_0 = \arcsin(y/x).$$

$$Q_1 = 180^\circ - (\alpha + \alpha_2).$$

$$Q_2 = 180^\circ - \beta.$$

$$Q_3 = (\gamma + \gamma_2 + \gamma_3) - 180^\circ.$$

Le calcul des angles  $Q_0$  jusque  $Q_3$  dans les coordonnées de référence liée aux articulations respectives



**Figure 2.17 :** Calcul des angles  $Q_1$ ,  $Q_2$  et  $Q_3$ .

## 2.1.4. Modèle cinématique

### 2.1.4.1. Modèle cinématique direct :

Le modèle cinématique direct d'un robot-manipulateur décrit les vitesses des coordonnées opérationnelles en fonction des vitesses articulaires [5]. Il est noté :

$$\dot{X} = J(q)\dot{q}. \quad (2.66)$$

Où  $J(q)$  désigne la matrice jacobienne du mécanisme. Pour le cas des robots série, elle est de dimension  $n \times n$ .

$V_n$  est la dérivée par rapport au temps du vecteur  $P_n$ , tandis que  $\omega_n$  ne représente pas la dérivée d'une représentation quelconque de l'orientation [5].

Ainsi, les vecteurs des vitesses de translation et de rotation  $V_n$  et  $\omega_n$  du repère  $\{R_n\}$  sont donnés par :

$$\begin{bmatrix} V_n \\ \omega_n \end{bmatrix} = J_n \dot{q}. \quad (2.67)$$

La jacobienne définissant les vitesses opérationnelles de l'organe terminal projetées dans un repère  $\{R_s\}$  peut être déduite de celle les définissant dans le repère  $\{R_p\}$ , par la relation suivante [5] :

$${}^5J_n = \begin{bmatrix} {}^5R_n & O_3 \\ O_3 & {}^5R_n \end{bmatrix} {}^iJ_n. \quad (2.68)$$

Ainsi, la jacobienne correspondant à une représentation quelconque des vitesses opérationnelles peut être déduite du Jacobien  $J_n$ , appelé jacobien de base.

#### 2.1.4.2. Calcul du Jacobien de base

On choisit comme Jacobien de base, la jacobienne  ${}^0J_5$  définissant les vitesses opérationnelles du repère  $\{R_5\}$  dans le repère de la base  $\{R_0\}$ , tel que :

$$\dot{X}_5 = {}^0J_5(q)\dot{q}. \quad (2.69)$$

Où  $\dot{X}_5$  représente les vitesses opérationnelles réelles de l'organe terminal projetées dans le repère de la base définie par :

$$\dot{X}_5 = \begin{bmatrix} V_5 \\ \omega_5 \end{bmatrix}. \quad (2.70)$$

Où  $V_5$  et  $\omega_5$  représente les vitesses de translation et de rotation de l'organe terminal dans le repère de la base  $\{R_0\}$ .

Le Jacobien de base d'un mécanisme à chaîne ouverte simple à  $n$  articulations définissant les vitesses opérationnelles du repère  $\{R_n\}$  par rapport au repère de la base  $\{R_0\}$  est noté  ${}^0J_5$ .

La  $k^{\text{ième}}$  colonne de la jacobienne  ${}^0J_n$  notée  ${}^0J_{n,k}$  s'écrit [5] :

$${}^0J_{n,k} = \begin{bmatrix} \sigma_k {}^0a_k + \bar{\sigma}_k {}^0\hat{a}_k ({}^0P_5 - {}^0P_k) \\ \bar{\sigma}_k {}^0a_k \end{bmatrix}. \quad (2.71)$$

$$\sigma_k = \begin{cases} 0 & \text{si la } k^{\text{ième}} \text{ articulation est rotoïde.} \\ 1 & \text{si la } k^{\text{ième}} \text{ articulation est prismatique.} \end{cases}$$

Où l'on rappelle que  ${}^0a_k$  est le vecteur unitaire porté par l'axe  $z_k$  de l'articulation K.

D'où, pour un robot de type série de 5 articulations rotoïdes, en occurrence le robot ROB3, la  $k^{\text{ième}}$  colonne de  ${}^0J_5$  s'écrit :

$${}^0J_{5,k} = \begin{bmatrix} {}^0\hat{a}_k ({}^0P_5 - {}^0P_k) \\ {}^0a_k \end{bmatrix}. \quad (2.72)$$

Tel que :

$${}^0a_1 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}; \quad {}^0a_2 = \begin{bmatrix} S\theta_1 \\ C\theta_1 \\ 1 \end{bmatrix}; \quad {}^0a_3 = \begin{bmatrix} S\theta_1 \\ C\theta_1 \\ 1 \end{bmatrix}; \quad {}^0a_4 = \begin{bmatrix} S\theta_1 \\ C\theta_1 \\ 1 \end{bmatrix};$$

$${}^0a_5 = \begin{bmatrix} C\theta_1 C(\theta_2 + \theta_3 + \theta_4) S\theta_5 - S\theta_1 C\theta_5 \\ S\theta_1 C(\theta_2 + \theta_3 + \theta_4) S\theta_5 - C\theta_1 C\theta_5 \\ S(\theta_2 + \theta_3 + \theta_4) S\theta_5 \end{bmatrix}.$$

En appliquant la formule du jacobien, et en utilisant le modèle géométrique direct, on trouve le Jacobien de base  ${}^0J_5$  du robot ROB3 comme suit :

$$\begin{bmatrix} \frac{dx}{dt} \\ \frac{dy}{dt} \\ \frac{dz}{dt} \\ \frac{d\phi_x}{dt} \\ \frac{d\phi_y}{dt} \\ \frac{d\phi_z}{dt} \end{bmatrix} = \begin{bmatrix} J_{11} & J_{12} & J_{13} & J_{14} & J_{15} \\ J_{21} & J_{22} & J_{23} & J_{24} & J_{25} \\ J_{31} & J_{32} & J_{33} & J_{34} & J_{35} \\ J_{41} & J_{42} & J_{43} & J_{44} & J_{45} \\ J_{51} & J_{52} & J_{53} & J_{54} & J_{55} \\ J_{61} & J_{62} & J_{63} & J_{64} & J_{65} \end{bmatrix} \times \begin{bmatrix} \frac{dq_0}{dt} \\ \frac{dq_0}{dt} \\ \frac{dq_0}{dt} \\ \frac{dq_0}{dt} \\ \frac{dq_0}{dt} \\ \frac{dq_0}{dt} \end{bmatrix}. \quad (2.73)$$

Les éléments de la matrice du Jacobi peuvent se calculer à l'aide de l'équation (2.72).

### 2.1.5. Le modèle cinématique inverse

L'objectif du modèle cinématique inverse est de calculer à partir d'une configuration  $q$  données, les vitesses articulaires  $\dot{q}$  qui assurent au repère terminal une vitesse opérationnelle  $\dot{X}$  imposée. Pour obtenir le modèle cinématique inverse, on inverse le modèle cinématique direct en résolvant un système d'équations linéaires. La mise en œuvre peut être faite de façon analytique ou numérique :

- la solution analytique a pour avantage de diminuer considérablement le nombre d'opérations, mais on doit traiter séparément tous les cas singuliers.
- les méthodes numériques sont plus générales. Elles nécessitent un temps de calcul relativement important.

Nous présentons dans cette section les techniques et méthodologies à mettre en œuvre pour établir un modèle cinématique inverse dans les cas réguliers et singuliers. [5]

#### 2.1.5.1. Forme générale du modèle cinématique

Quelle que soit la méthode utilisée pour décrire les coordonnées opérationnelles, le modèle cinématique direct peut être mise sous la forme :

$$\dot{X} = {}^0J_x \dot{q}. \quad (2.74)$$

Si on utilise les coordonnées cartésiennes pour décrire la position. Pour alléger l'écriture, nous adopterons la forme suivante de l'équation :

$$\dot{X} = J \dot{q}. \quad (2.75)$$

#### 2.1.5.2. Modèle cinématique inverse dans le cas régulier

Dans ce cas, la matrice jacobienne  $J$  est carrée d'ordre  $n$  et son déterminant est non nul. Deux méthodes de calcul peuvent être mises en œuvre.

##### 2.1.5.2.1. Première méthode

On calcule  $J^{-1}$ , la matrice inverse de  $J$ , qui permet de déterminer les vitesses articulaires  $\dot{q}$  grâce à la relation :

$$\dot{q} = J^{-1}\dot{X}. \tag{2.76}$$

Lorsque la matrice  $J$  a la forme suivante :

$$J = \begin{bmatrix} A & 0 \\ B & C \end{bmatrix}. \tag{2.77}$$

Les matrices  $A$  et  $C$  étant carrées inversibles, il est facile de montrer que l'inverse de cette matrice s'écrit :

$$J^{-1} = \begin{bmatrix} A & 0 \\ -C^{-1}BA^{-1} & C^{-1} \end{bmatrix} \tag{2.78}$$

La résolution du problème se ramène donc à l'inversion, beaucoup plus simple, de deux matrices de dimension moindre. Lorsque le robot manipulateur possède six degrés de liberté et un poignet de type rotule, la forme générale de  $J$  est celle de la relation (2.78)  $A$  et  $C$  étant de dimension (3x3), ce qui n'est pas le cas de notre robot.

### 2.1.5.2.2. Seconde méthode

Dans cette méthode, on tient compte d'une éventuelle forme particulière de la matrice  $J$  permettant de réduire le nombre d'inconnues. Cette méthode donne, dans la plupart des cas, des solutions nécessitant moins d'opérations. Prenons par exemple, le cas d'un robot manipulateur à poignet rotule dont le jacobien a, comme on l'a déjà dit, la structure de l'équation (2.77) :

$$\begin{bmatrix} \dot{X}_p \\ \dot{X}_r \end{bmatrix} = \begin{bmatrix} A & 0 \\ B & C \end{bmatrix} \begin{bmatrix} \dot{q}_a \\ \dot{q}_b \end{bmatrix}. \tag{2.79}$$

$A$  et  $C$  étant des matrices carrées de dimension (3x3), inversibles en dehors des positions. La solution  $\dot{q}$  est donnée par :

$$\begin{cases} \dot{q}_a = A^{-1}\dot{X}_a \\ \dot{q}_b = C^{-1}[\dot{X}_b - B\dot{q}_a] \end{cases} \tag{2.80}$$

Solution qui est beaucoup plus simple que celle obtenue avec la première méthode.

### 2.1.6. Modèle dynamique

Le modèle des bras manipulateurs sont décrits par un système d'équation mathématique qui décrivent sa dynamique. Les robots étant des structures mécaniques, l'ensemble des équations dynamiques peuvent être déterminé par les lois de la mécanique classique ; le modèle dynamique décrit la relation entre les couples (et/ou forces) appliqués aux actionneurs et les positions, vitesse et accélération articulaires. Je représente le modèle dynamique par une relation de la forme [5], [10]:

$$\Gamma = f(q, \dot{q}, \ddot{q}, f_e). \quad (2.81)$$

- $\Gamma$  : vecteur des couples/forces des actionneurs, selon que l'articulation est rotoïde ou prismatique. Dans la suite, on écrira tout simplement couples ;
- $q$  : vecteur des positions articulaire ;
- $\dot{q}$  : vecteur des vitesses articulaires ;
- $\ddot{q}$  : vecteur des accélérations articulaires ;
- $f_e$  : vecteur représentant l'effort extérieur (forces et moments) qu'exerce le robot sur l'environnement.

Plusieurs formalismes ont été utilisés pour obtenir le modèle dynamique des robots, les formalismes les plus utilisés sont le formalisme de Lagrange et le formalisme de Newton-Euler qui fera l'objet de ce chapitre.

#### 2.1.6.1 Formalisme de Lagrange

Le but de ce paragraphe est d'étudier la forme générale du modèle dynamique, de mettre en évidence les différents termes qui y interviennent [5].

Le formalisme de Lagrange décrit les équations du mouvement en termes de travail et énergie du système ; ce qui se traduit par l'équation suivante :

$$\Gamma_i = \frac{d}{dt} \frac{\partial L}{\partial \dot{q}} - \frac{\partial L}{\partial q} \quad i = 1, \dots, n. \quad (2.82)$$

- $L$  : lagrangien du système égale à  $L=E-U$  ;
- $E$  : énergie cinétique totale du système ;

- U : énergie potentielle totale du système.

### 2.1.6.1 Forme générale de l'équation dynamique

L'équation dynamique peut se mettre en générale sous la forme :

$$\Gamma = A(q)\ddot{q} + C(q, \dot{q})\dot{q} + Q(q) + \Gamma_f + \Gamma_e + \Gamma_{m0}. \quad (2.83)$$

Où :

$A(q)$  est la matrice (n x n) de l'énergie cinétique, appelée aussi matrice d'inertie du robot;  $C(q, \dot{q})\dot{q}$  vecteur de dimension (n x 1) représentant les couples/forces de Coriolis et des forces centrifuges ;  $Q(q)$  vecteur des forces/couples de gravité.

#### 2.1.6.1.1. Calcul de l'énergie cinétique

$$E = \frac{1}{2} \dot{q}^T A(q) \dot{q}. \quad (2.84)$$

L'énergie cinétique du système est une fonction quadratique des vitesses articulaires :

L'énergie cinétique totale du système est donnée par la relation :

$$E = \sum_{j=1}^n E_j. \quad (2.85)$$

Où  $E_j$  désigne l'énergie cinétique du corps  $C_j$ , qui s'exprime par :

$$E_j = \frac{1}{2} (\omega_j^T I_{Gj} \omega_j + M_j V_{Gj}^T V_{Gj}). \quad (2.86)$$

#### 2.1.6.1.2. Calcul de l'énergie potentielle

L'énergie potentielle s'écrit :

$$U = \sum_{k=1}^n U_j = \sum_{k=1}^n -M_j g^T (L_{0,j} + S_j). \quad (2.87)$$

$L_{0,j}$  désignant le vecteur d'origine  $O_0$  et d'extrémité  $O_j$ . En projetant les vecteurs de cette relation dans  $R_0$ , on obtient :

$$U_j = -M_j {}^0g^T ({}^0P_j + {}^0A_j {}^jS_j). \quad (2.88)$$

L'expression précédente peut se mettre sous la forme

$$U_j = -{}^0g^T (M_j {}^0P_j + {}^0A_j {}^jMS_j) = [{}^0g^T \quad 0] \begin{bmatrix} {}^jMS_j \\ M_j \end{bmatrix}. \quad (2.89)$$

les éléments de Q tel que  $Q = [Q_1 \dots Q_n]^T$  se calculent en écrivant que :

$$Q_i = \frac{\partial U}{\partial q_i}. \quad (2.90)$$

## 2.2. Identification

L'ensemble des méthodes de commande d'un processus nécessitent la connaissance d'un modèle mathématique du système. Les méthodes d'identifications paramétriques se classent généralement en deux grandes familles.

- Les méthodes applicables seulement aux systèmes linéaires par rapport aux paramètres.
- Les méthodes applicables aux systèmes linéaires et non linéaires par rapport aux paramètres.

Parmi les méthodes directes on peut distinguer les méthodes « directes » comme par l'analyse indicielle et fréquentielle d'un système, et les méthodes « numériques » basées sur la minimisation d'un critère.

Les méthodes directes permettent d'identifier des modèles à structure simple. Les méthodes les plus répondu sont :

L'une des plus connues et la méthode de Broida [11] qui permet la recherche des paramètres d'un modèle de premier ordre, et la méthode de Strejc [11] utiliser pour la recherche des paramètres d'un modèle d'ordre supérieure ou égal a deux. Ces méthodes sont souvent appeler les méthodes « graphiques » car les paramètres du modèle sont déterminés à partir de l'analyse de la réponse temporelle du procédé.

- Méthodes basées sur l'analyse fréquentiel du système

Cette méthode cherche des paramètres de la fonction de transfert d'un système. Lorsque celui-ci est soumis à des entrées sinusoïdales. La détermination des paramètres de la fonction de transfert s'effectuera grâce au diagramme de Bode.

- Méthode des moments

Cette méthode est utilisée pour estimer les paramètres d'une équation différentielle. Elle peut être mise en œuvre de deux façons différentes :

- Par intégration temporelle d'une équation différentielle, ceci permettra de calculer les moments temporels nécessaires pour la détermination des paramètres de l'équation différentielle.
- En utilisant la transformé de Laplace et le développement en série de la fonction de transfert. Cette seconde approche aura l'avantage de définir les moments temporels de façon plus générale.

### 2.2.1 Description de la méthode utilisée pour l'identification

L'identification des systèmes consiste à déterminer les caractéristiques dynamiques d'un processus (ensemble des équations décrivant le mieux le système).

Cette procédure comporte quatre étapes.

- Acquisition des données entrée/sortie du système.
- Choix de la structure du modèle.
- Estimation des paramètres.
- Validation du modèle.

L'application du principe fondamental de la dynamique conduit à l'équation de mouvement du type :

$$J\ddot{q} + h(\dot{q}) + g(q) = \Gamma. \quad (2.91)$$

$J\ddot{q}$  représente le couple dû à la force inertielle.

$h(\dot{q})$  représente le couple dû aux frottements.

$g(q)$  décrit le couple dû à la pesanteur lorsqu'elle intervient.

$\Gamma$  est le couple développé par l'actionneur pour un axe, et les forces de Coriolis dues à l'interaction avec les autres axes n'existent pas, et la force centrifuge a un couple nul par rapport à l'axe de rotation.

Par conséquent les frottements visqueux ont une prépondérance devant la pesanteur, l'équation du mouvement peut s'écrire de la forme linéaire suivante :

$$\ddot{q} + J^{-1}C(\dot{q})\dot{q} + J^{-1}G(q)q = J^{-1}\Gamma(u). \quad 2.92)$$

En posant

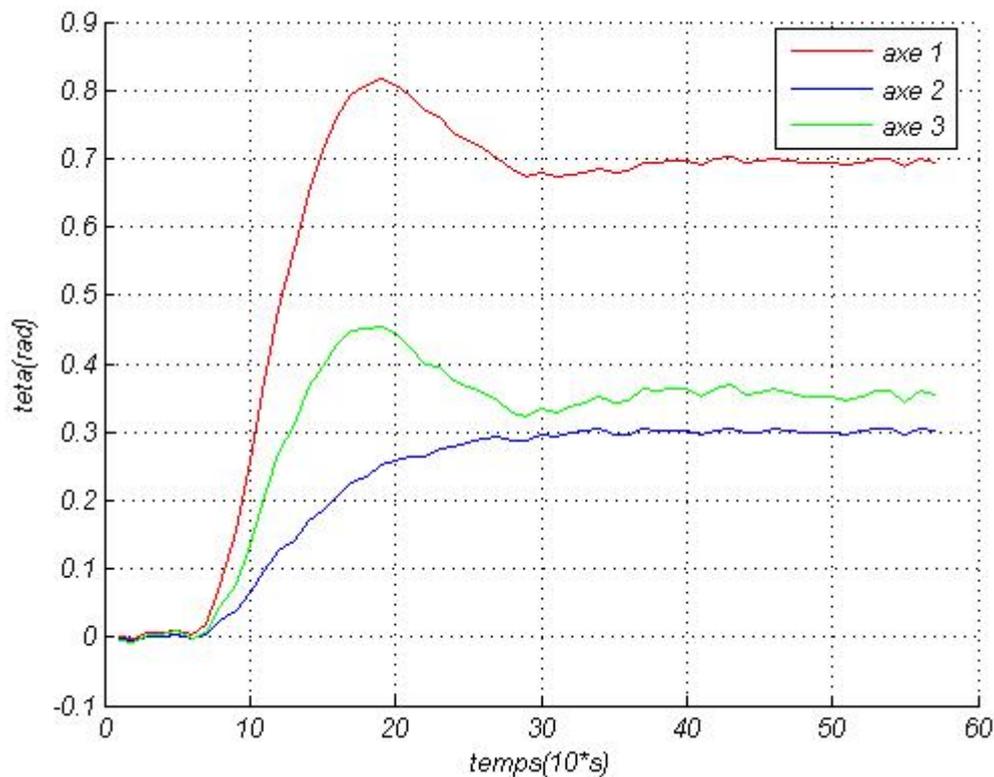
$$G(q)q = g(q) \quad \text{et} \quad C(\dot{q})\dot{q} = h(\dot{q}). \quad (2.93)$$

Où  $C(\dot{q})\dot{q}$  caractérise les frottements complexes dans l'actionneur à courant continu dans l'optimisation dans le cadre de l'approximation linéaire, l'équation du mouvement s'écrit :

$$\ddot{q} + a_1\dot{q} + a_2q = bU. \quad (2.94)$$

Où  $a_1$ ,  $a_2$  et  $b$  sont des paramètres à identifier.

Nous avons commencé par recueillir les sorties obtenues après avoir injecté des échelons avec des amplitudes assez grandes pour chaque axe.



**Figure 2.18** : réponse indicielle du bras pour les trois axes.

Le tableau (2.1) ci-dessous représente les paramètres du système identifié avec un modèle continu de deuxième d'ordre, ces paramètres seront utilisés dans la synthèse de la loi commande.

$$\frac{q_i}{U_i} = \frac{b_{0,i}}{S^2 + a_{1,i}S + a_{2,i}} \left( \frac{\text{rad}}{\text{volt}} \right). \quad (2.95)$$

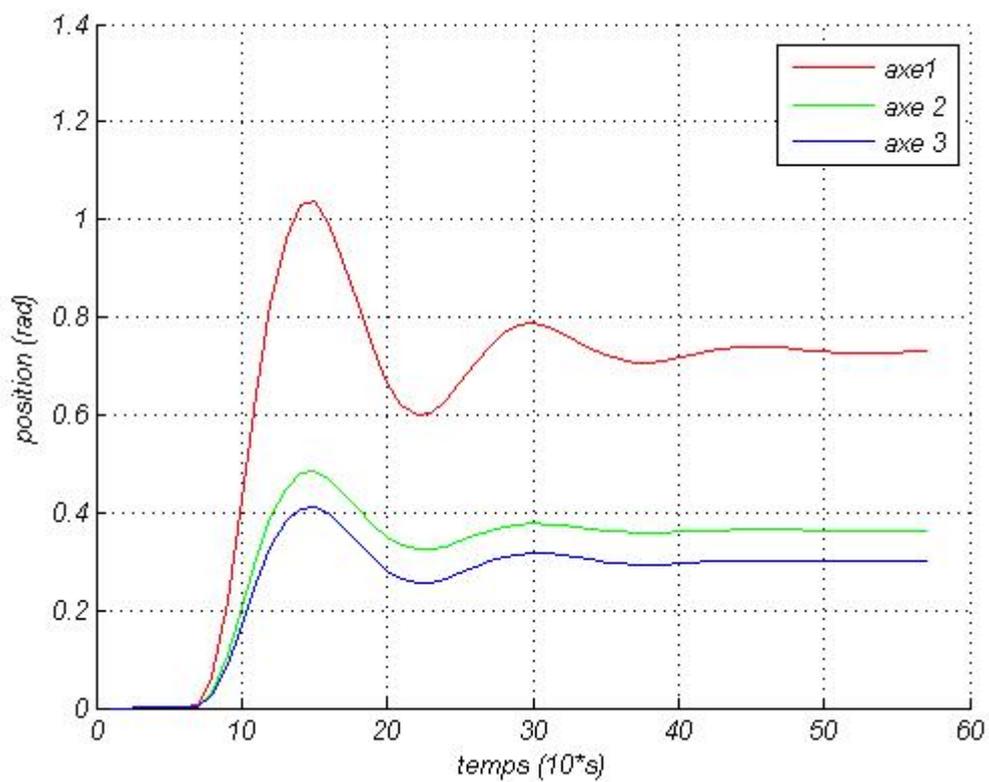
Axes	$a_1$	$a_2$	$b$
Axe 1	1.123	3.317	1.231
Axe 2	1.421	3.345	1.300
Axe3	1.250	3.173	1.360

**Tableau 2.2:** Les paramètres des modèles de simulation.

Les paramètres sont obtenus pour des échelons de 0.7 rad, 0.38 rad et 0.30 rad.

### 2.3. Test de validation

La figure 2.19 illustre les réponses indicielles du modèle identifié.



**Figure 2.19 :** réponse indicielle du bras pour le modèle identifié.

Les figures 2.20.a, 2.20.b. et 2.20.c. nous donne les erreurs de modélisation pour les trois axe étudié

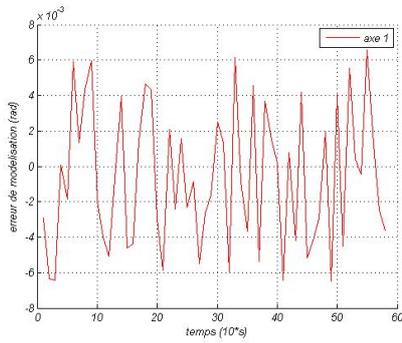


Figure 2.20.a : axe 1.

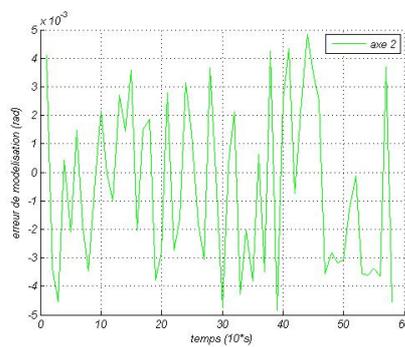


Figure 2.20.b : axe 2.

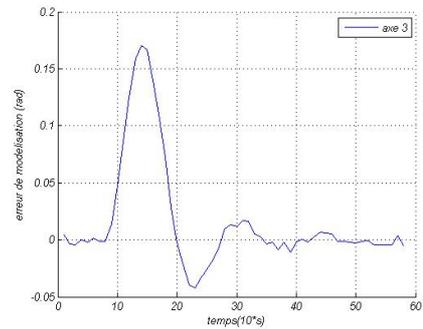


Figure 2.20.c : axe 3.

## 2.4. Conclusion

Dans ce deuxième chapitre nous avons présenté les différentes approches ou méthodes de modélisation des robots manipulateurs, et cela pour pouvoir par la suite élaborer des lois de commande.

Ayant besoin des différents modèles mathématiques du système, nous avons commencé par introduire un modèle géométrique directe, modèle géométrique inverse, modèle cinématique, modèle cinématique inverse et enfin le modèle dynamique, ce dernier étant très difficile à calculer en a opté pour une identification paramétrique en utilisant un modèle linéaire de deuxième ordre.

CHAPITRE 3  
*STRATEGIES DE  
COMMADE ET  
SIMULATIONS.*

### 3. Stratégies de commande et simulations

Les simulations ont été faites en utilisant le Toolbox Simulink de MatLab 7.2, nous avons étudié un modèle du robot et de son environnement en utilisant les résultats de l'identification qui ont été présentés dans le chapitre 2, puis nous avons introduit des régulateurs PD et PID décentralisées et par mode glissant, l'objectif étant de régler ces paramètres en simulation avant de les implémenter sur le robot réel. Enfin nous avons comparé les résultats obtenus par les différentes techniques de commande le PID et le glissement. Nous utilisons par la suite ces mêmes paramètres, obtenus en simulations pour l'implémentation des algorithmes de réglages. [4]

#### 3.1 Commande classique

Le modèle dynamique décrit un système de  $n$  fonctions de transferts du second ordre découplées,  $n$  étant le nombre d'articulations. Pourtant, dans une commande classique, qui est celle de la plupart des robots industriels actuels, le mécanisme est considéré comme un système linéaire et chacune de ses articulations est asservie par une commande décentralisée de type PID à gains constants.

Leurs avantages est la facilité d'implémentation et le faible coût en calcul par contre, je constate que la réponse temporelle varie suivant la configuration du robot et des dépassements de consigne. Dans beaucoup d'application ces inconvénients ne représentent pas un gros problème.

Une telle commande est représentée sur la figure 3.1 :

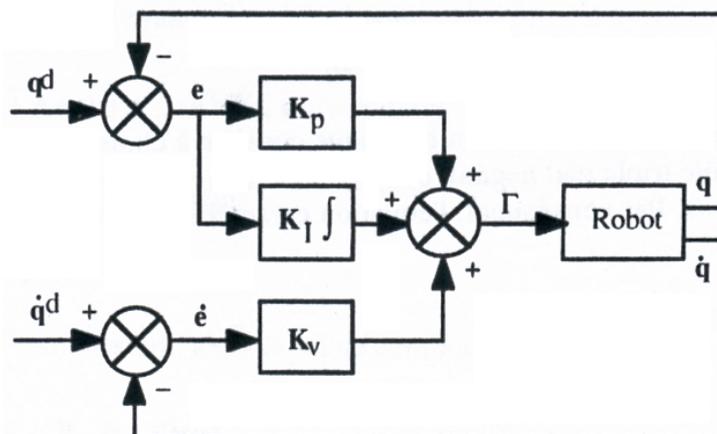


Figure 3.1 : Schéma classique d'une commande PID.

La loi de commande est donnée par :

$$\Gamma = K_p(q - q_d) + K_v(\dot{q} - \dot{q}_d) + K_i \int_{t_0}^t (q - q_d) d\tau. \quad (3.1)$$

Où  $q_d$  et  $\dot{q}_d$  représentent la vitesse et position désirées dans l'espace articulaire  $K_p$ ,  $K_v$  et  $K_i$  sont des matrices diagonales d'élément  $K_{pj}$ ,  $K_{vj}$  et  $K_{ij}$ , le calcul de ces gains est effectué en prenons le modèle de l'articulation  $j$  identifié dans le chapitre deux, elle est donnée par :

$$\frac{q_i}{U_i} = \frac{b_{0,i}}{S^2 + a_{1,i}S + a_{2,i}} \left( \frac{\text{rad}}{\text{volt}} \right). \quad (3.2)$$

Axes	$a_1$	$a_2$	$b$
Axe 1	1.123	3.317	1.231
Axe 2	1.421	3.345	1.300
Axe3	1.250	3.173	1.360

**Tableau 3.1** : Les paramètres des modèles de simulation.

La fonction de transfert en boucle fermée représente le transfert entre la position réelle  $q_d(t)$  et  $q_d(t)$  la position désirée, elle est donnée par :

$$\frac{q(s)}{q_d(s)} = \frac{b_0(K_p S + K_v S^2 + K_i)}{S^3 + (a_1 + b_0 K_v)S^2 + (a_2 + b_0 K_p)S + b_0 K_i}. \quad (3.3)$$

La solution la plus courante en robotique est de choisir les gains de manière à obtenir les pôles en boucles fermé comme étant un pôle triple réel négatif, se qui donne une réponse la plus rapide possible sans oscillation.

$$\Delta S = (S + w_j)^3. \quad (3.4)$$

Avec  $\omega_j > 0$ . Les expressions des gains des régulateurs sont donnée par :

$$\left\{ \begin{array}{l} a_1 + b_0 K_v = 3w_j \\ a_2 + b_0 K_p = 3w_j^2 \\ b_0 K_i = w_j^3 \end{array} \right. \quad (3.5)$$

Pour se qui a été dit précédemment, nous faisons les remarque suivantes :

- $\omega_j$  est choisi le plus grand possible, mais il est important de ne pas la choisir supérieure à la pulsation de résonance  $\omega_{jr}$  correspondante aux modes de vibration mécanique afin de ne pas déstabiliser le système, en général on prend  $\omega_j = \omega_{jr}/2$ .

Les performances d'une telle méthode sont d'autant plus acceptable que le rapport de réduction est important et que sont faibles.

Pour le robot ROB3, il a été montré qu'une commande Proportionnelle Dérivée suffit et cette commande est asymptotiquement stable, une telle commande est représenté par le schéma suivant :

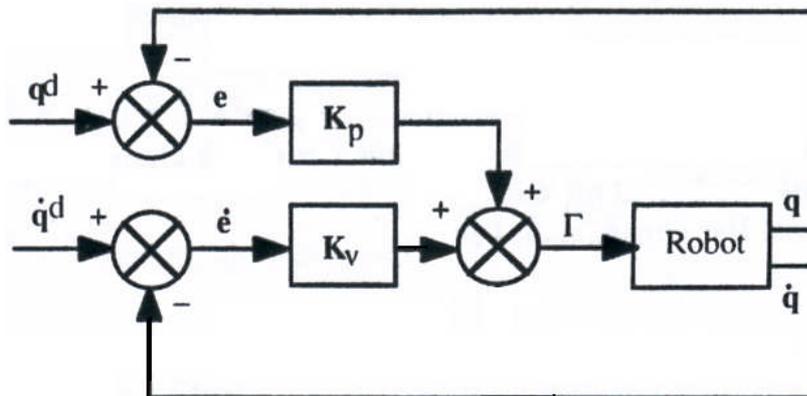


Figure 3.2. : Schéma classique d'une commande PD.

La loi de commande est donnée par :

$$\Gamma = K_p(q - q_d) + K_v(\dot{q} - \dot{q}_d). \tag{3.6}$$

Dans ce cas la fonction de transfert en boucle fermée devient :

$$\frac{q(s)}{q_d(s)} = \frac{b_0(K_p + K_v s)}{s^2 + (a_1 + b_0 K_v) s + a_2 + b_0 K_p}. \tag{3.7}$$

La solution la plus courante en robotique est de choisir les gains de manière à obtenir les pôles en boucles fermée comme étant un pôle triple réel négatif, se qui donne une réponse la plus rapide possible sans oscillation.

$$\Delta S = (S + w_j)^2. \tag{3.8}$$

Avec  $\omega_j > 0$ . Les expressions des gains des régulateurs sont donnée par :

$$\begin{cases} a_2 + b_0 K_p = \omega_j^2 \\ a_1 + b_0 K_v = 2\omega_j \end{cases} \quad (3.9)$$

### 3.1.2. Simulation

#### 3.1.2.1 Commande Proportionnelle Dérivée

Pour un robot de type ROB3 une commande proportionnelle dérivée suffit pour effectuer la poursuite de trajectoire, la commande est donnée par (3.6) ; Dans se qui suit je vais présenter les simulations utilisant se type de commande.

Je prendrai le pôle en boucle fermée  $\omega=6$  et cela pour les trois articulations ; La méthode utilisée pour le calcul des coefficients des régulateur est celle citée au dessus.

D'après les relations données par (3.9), nous obtenons les matrices  $K_p$  et  $K_v$  suivante :

Pour  $\omega=6$  nous avons :

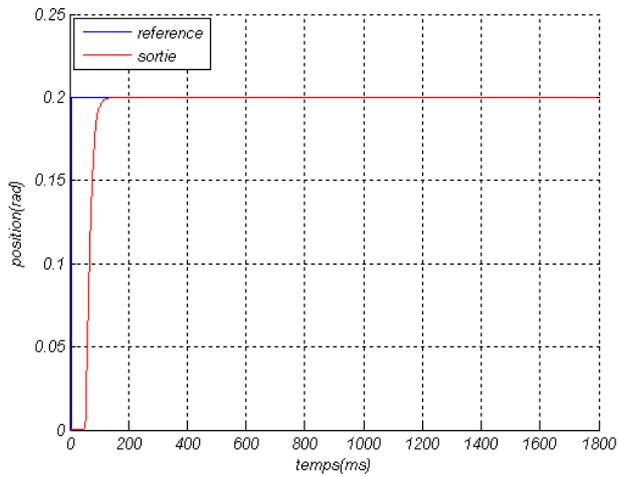
$$K_v = \begin{bmatrix} 8.8359 & 0 & 0 \\ 0 & 8.1385 & 0 \\ 0 & 0 & 7.9044 \end{bmatrix}; K_p = \begin{bmatrix} 26.9439 & 0 & 0 \\ 0 & 24.1397 & 0 \\ 0 & 0 & 25.1231 \end{bmatrix}.$$

Ainsi la commande est donné par :

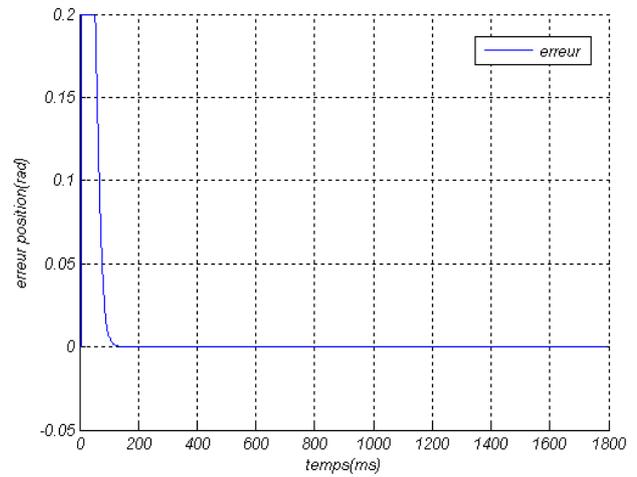
$$\Gamma = K_p e + K_v \dot{e} = K_p (q - q_d) + K_v (\dot{q} - \dot{q}_d). \quad (3.10)$$

Les simulations ont été effectués pour des faibles et grands déplacements, sans charge, nous commençons par prendre  $\omega = 6$ , nous obtenons se qui suit :

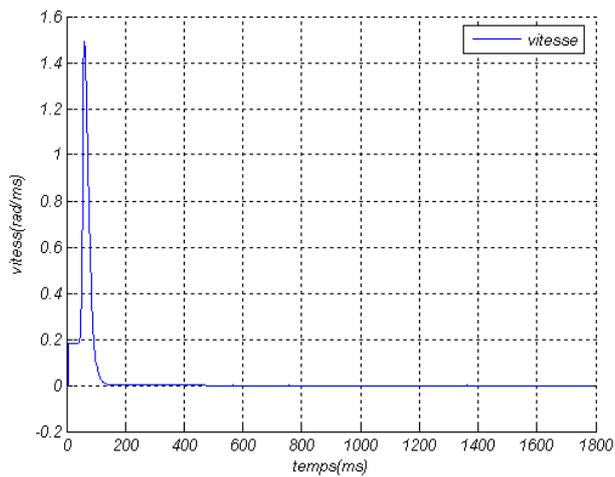
Les données sont  $q_i = 0$  et  $q_d = \pi/5.7$  nous obtenons les résultats suivant :



**Figure 3.3.1.a :** *Suivi en position.*



**Figure 3.3.1.b :** *Erreur en position.*



**Figure 3.3.1.c :** *suivi en vitesse.*

Les références  $q_d$  sont des sinusoïdes d'amplitudes  $\pi/15.7$  et de périodes 2 nous obtenons les résultats suivant :

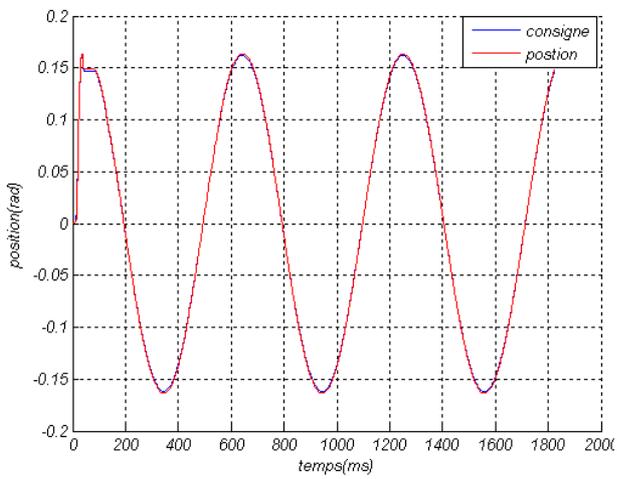


Figure 3.3.2.a : suivi en position.

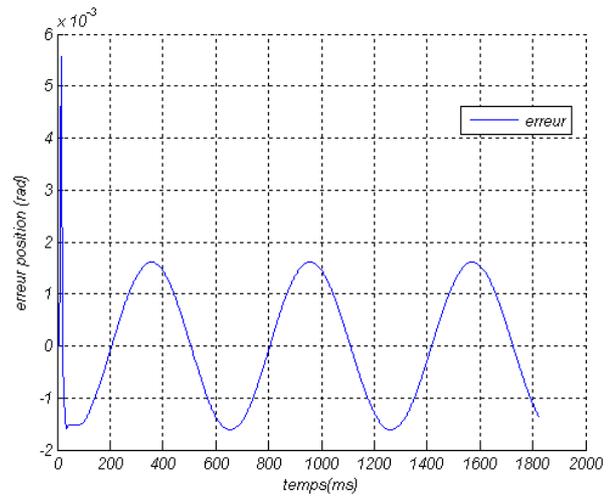


Figure 3.3.2.b : erreur en position.

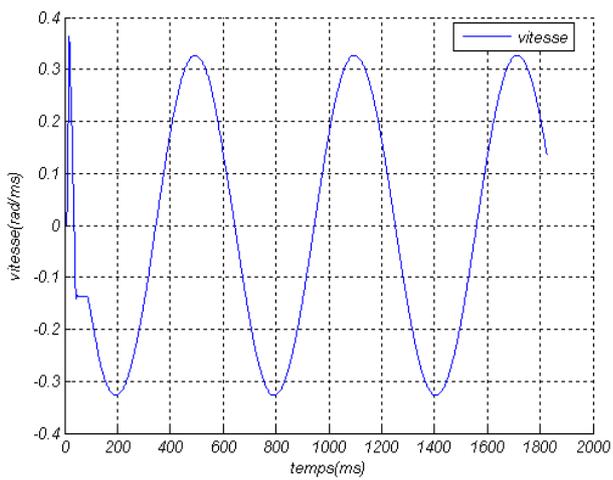
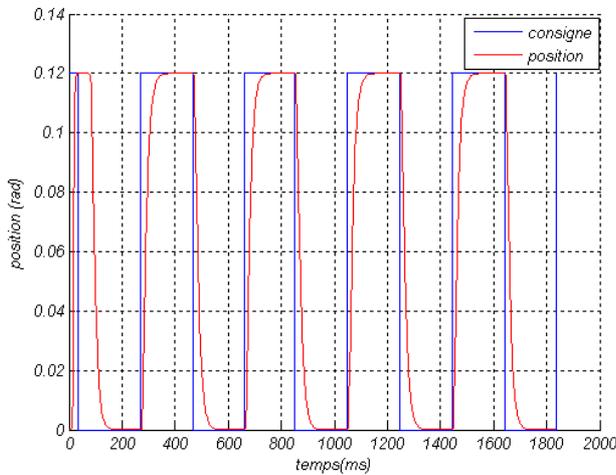
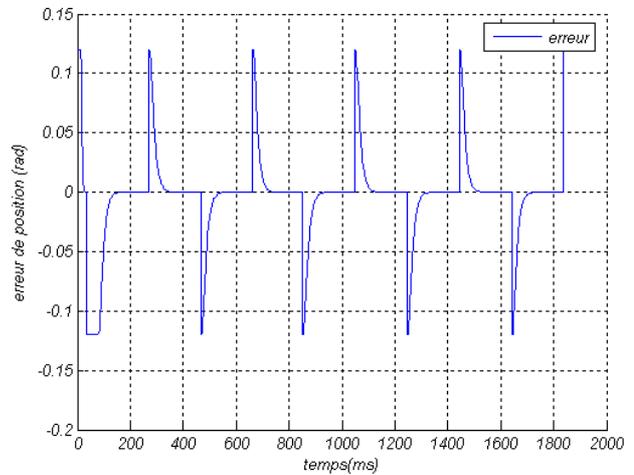


Figure 3.3.2.c : suivi en vitesse.

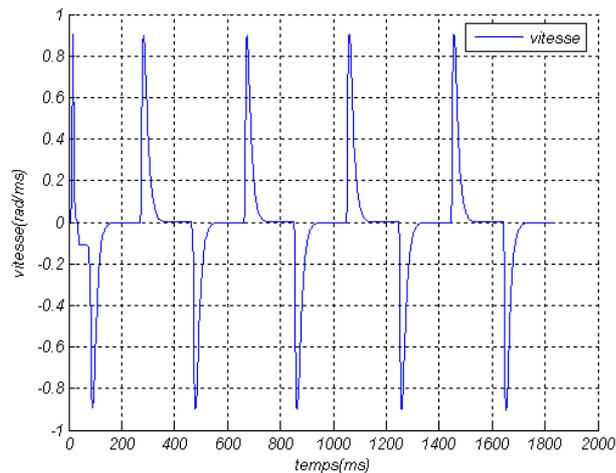
Les données  $q_d$  sont des impulsions d'amplitude  $\pi/22.5$  et fréquences de 0.5hz nous obtenons les résultats suivant :



**Figure 3.3.3.a** : suivi en position.



**Figure 3.3.3.b** : erreur en position.



**Figure 3.3.3.c** : suivi en vitesse.

### 3.1.2.2. Commande Proportionnelle intégrale et Dérivée

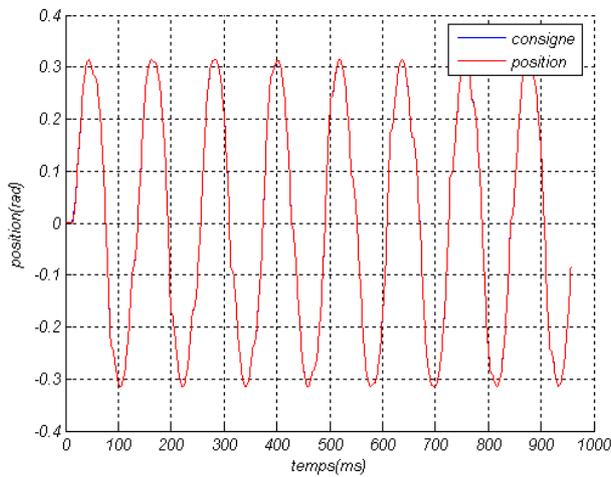
Précédemment nous avons constaté que nous avons un bon suivi de trajectoire dans cette section nous allons utiliser une commande de type PID et voir l'amélioration qui peut en résulter, le calcul des matrices  $K_p$  et  $K_v$  et  $K_I$  se fait à partir de la relation (3.10), les pôles sont choisis comme pour la commande PD, nous obtenons les matrices suivantes :

D'après les relations données par (3.5), nous obtenons les matrices  $K_p$  et  $K_v$  suivante :

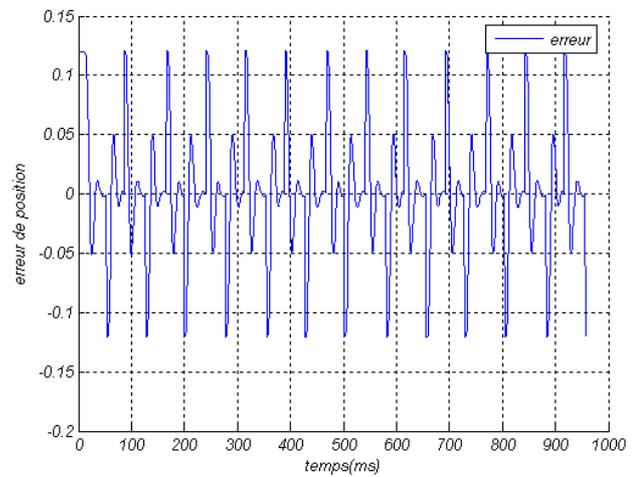
Pour  $\omega=6$  nous avons :

$$K_P = \begin{bmatrix} 86.8213 & 0 & 0 \\ 0 & 81.9846 & 0 \\ 0 & 0 & 78.4926 \end{bmatrix}; K_V = \begin{bmatrix} 13.7100 & 0 & 0 \\ 0 & 12.7538 & 0 \\ 0 & 0 & 12.3162 \end{bmatrix}; K_I = \begin{bmatrix} 175.4671 & 0 & 0 \\ 0 & 166.1538 & 0 \\ 0 & 0 & 158.8235 \end{bmatrix}.$$

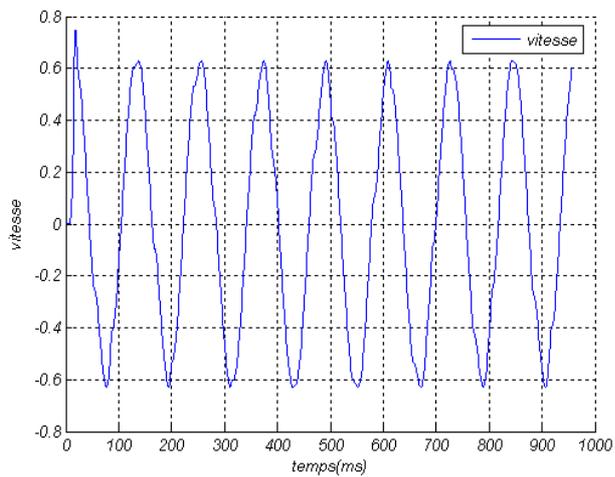
Les données sont des sinusoïdes d'amplitudes  $\pi/10$  et de fréquences 0.5hz nous obtenons les résultats suivant :



**Figure 3.3.4.a :** *suivi en position.*



**Figure 3.3.4.b :** *erreur en position.*



**Figure 3.3.4.c :** *suivi en position.*

### 3.1.3. Conclusion

La commande décentralisée du type PD et PID appliquée sur un robot de type ROB3 a donné de bons résultats pour le suivi de trajectoire et, nous pouvons constater que pour des faibles et grands déplacements, la commande à la même grandeur, cela est dû par le fait que pour une trajectoire donnée, le temps de parcours est optimal, donc nous pouvons conclure des simulations effectuées précédemment qu'une commande de type PD ou PID est une commande satisfaisante pour suivre une trajectoire de consigne.

## 3.2. Commande par mode glissement

Nous définissons pour les simulations :

- Un modèle linéaire.
- Les perturbations et le bruit des capteurs n'ont pas été pris en compte.
- Les consignes sont des échelons ou des sinusoïdes.

La dynamique des axes du robot est approchée par un modèle mathématique linéaire du second ordre fourni par l'identification de chaque axe séparément,

$$\frac{q_i}{U_i} = \frac{b_{0,i}}{S^2 + a_{1,i}S + a_{2,i}} \left( \frac{\text{rad}}{\text{volt}} \right). \quad (3.11)$$

I : représente le numéro de l'axe (1,2 ou

3) Avec

Pour nos simulations nous avons utilisé une représentation d'état du système (3.12) pour avoir un accès direct aux variables d'états,  $q_1$  et  $q_2$ .

$$\begin{cases} \dot{q}_1 = q_2 \\ \dot{q}_2 = -a_{1,i} q_2 - a_{2,i} q_1 + b_{0,i} U \end{cases} \quad (3.12)$$

Le problème est de définir  $U(t)$  pour amener le plus vite possible le vecteur d'erreur  $E = (e, \dot{e}) = (q - q_d, \dot{q} - \dot{q}_d)$  sur une variété de commutation et l'y maintenir jusqu'à l'état désiré avec la  $\ddot{q}_d = 0$ .

Dans la dynamique de régulation le manipulateur étant initialement au repos ( $q(0)$  et  $\dot{q}(0)$ ), nous voulons que  $q$  rallie une consigne  $q_d$  selon une certaine dynamique.

### 3.2.1. Surface de glissement :

Afin de concevoir une commande glissante, il faut définir une surface de glissement  $S$  dans l'espace d'état des erreurs. L'espace d'état des erreurs  $E$  pour les différents axes est

$$E = (e, \dot{e}) = (q - q_d, \dot{q} - \dot{q}_d). \quad (3.13)$$

Les valeurs désirées  $q_{di}$ , sont des échelons de position. Dans notre cas  $q_{di}$  sont des constantes donc  $\dot{q}_{di} = 0$ .

Dans ce qui suit : nous supposons la surface  $S$  comme étant linéaire et choisie sous la forme :

$$S(t) = (\lambda e + \dot{e}) \text{ Avec } \lambda > 0. \quad (3.14)$$

### 3.2.2. Commande équivalente

La commande équivalente est déterminée à partir de la solution de l'expression  $\dot{s} = 0$  on force l'expression précédente à zéro on obtient :

$$U_{eq} = b^{-1}(a_1 \dot{e} + a_2 e - \lambda \dot{e} + a_2 \dot{q}_d). \quad (3.15)$$

A cause des erreurs de modélisation la commande équivalente  $U_{eq}$  n'est qu'une estimation.

Les figures qui vont suivre illustrent les résultats obtenus en appliquant la commande à structure variable classique

Nous explicitons dans ces figures les allures des courbes qui représentent :

- La surface de glissement
- L'erreur de position
- La variation de l'erreur
- La commande

### 3.2.3. Résultats de la simulation

Plusieurs essais ont été effectués, les meilleurs résultats sont obtenus pour un coefficient de surface égal à 100. Le régime glissant est atteint en un temps de l'ordre de 0.1s ; à partir de cet instant le système évolue comme un système du premier ordre.

Il est insensible aux variations des paramètres du procédé puisque sa dynamique est fixée par la surface de glissement.

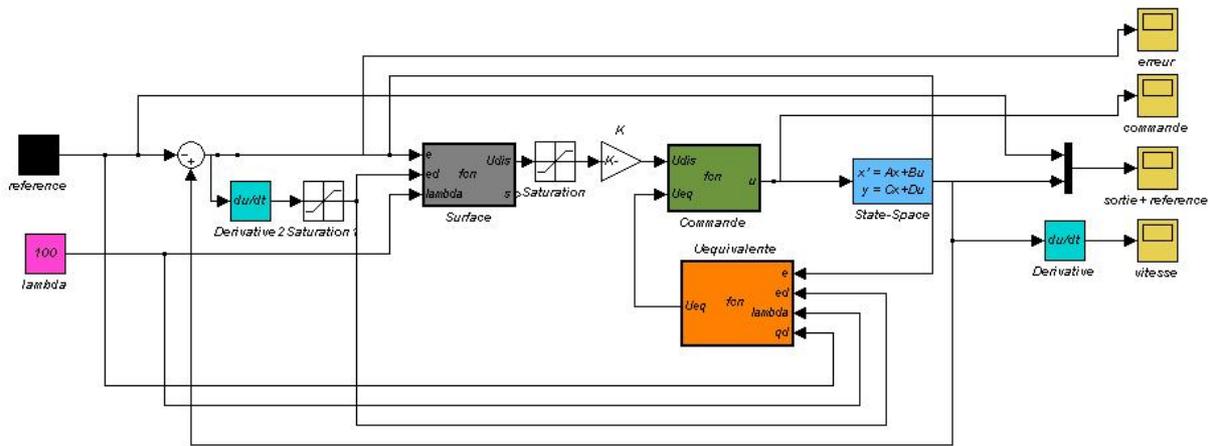


Figure 3.4 : Schéma de la simulation sur Matlab de la commande par mode de glissement.

Les données sont des impulsions d'amplitudes  $\pi/2$  et de fréquences 0.5hz nous obtenons les résultats suivant :

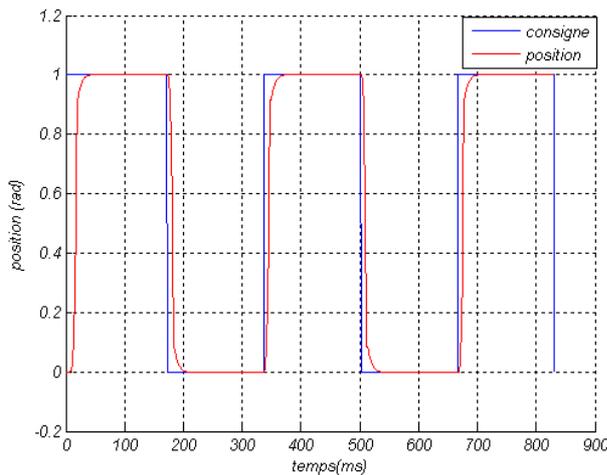


Figure 3.5.1.a : suivi en position.

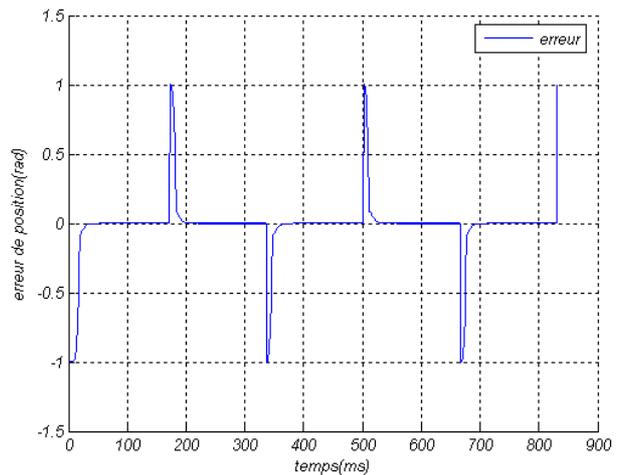
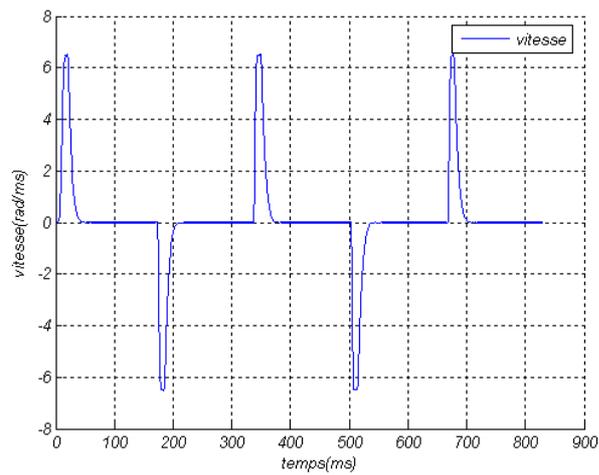
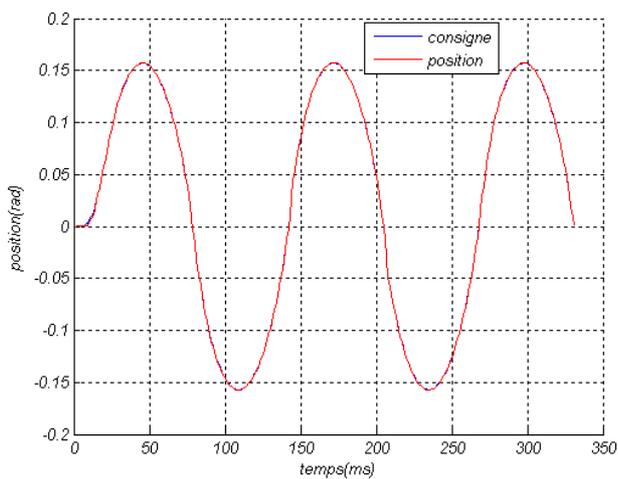


Figure 3.5.1.b : erreur en position.

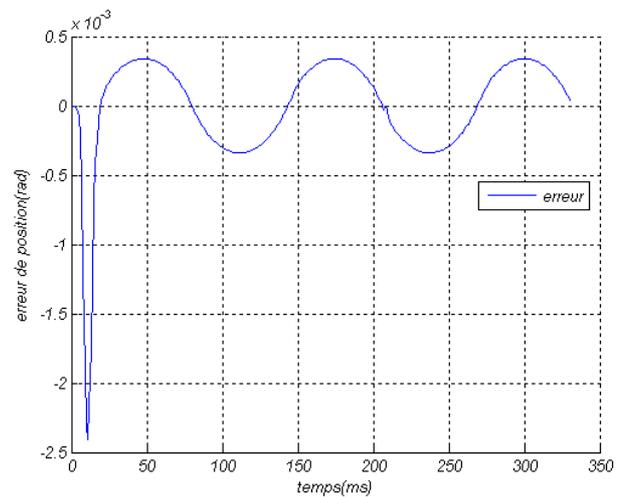


**Figure 3.5.1.c :** *suivi en vitesse.*

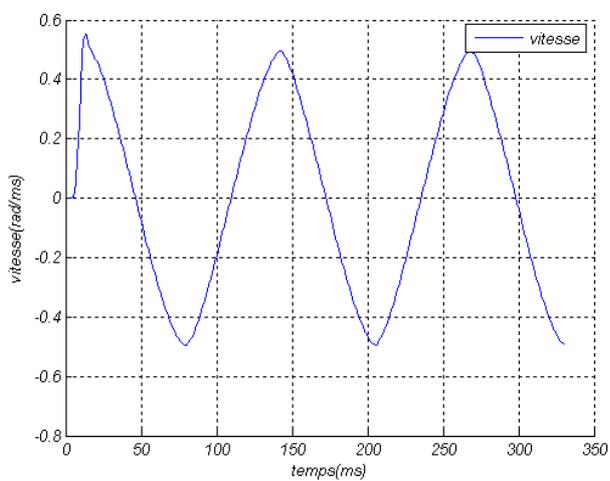
Les données sont des sinusoïdes d'amplitudes  $\pi/20$  et de fréquences 0.5hz nous obtenons les résultats suivant :



**Figure 3.5.2.a :** *suivi en position.*



**Figure 3.5.2.b :** *erreur en position.*



**Figure 3.5.2.c :** *suivi en vitesse.*

### 3.2.4. Test de robustesse

Dans cette partie je m'intéresse à la capacité de poursuite de référence en présence des erreurs de modélisation.

Les données sont des impulsions d'amplitudes  $\pi/2$  et de fréquences 0.5hz, aussi les paramètres du modèle sont perturbé a 50%, nous obtenons les résultats suivant :

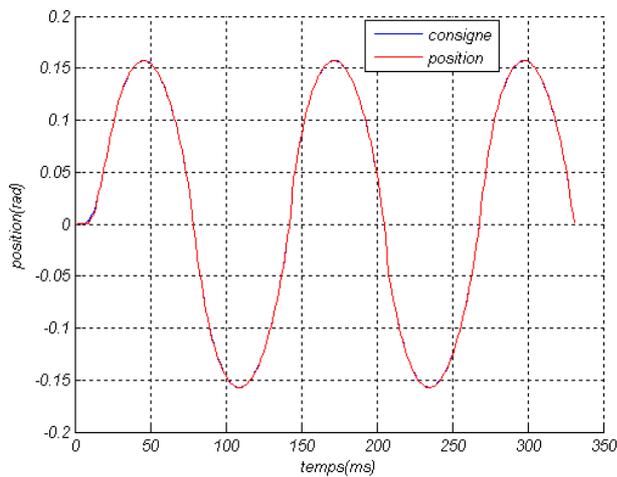


Figure 3.6.1.a : suivi en position.

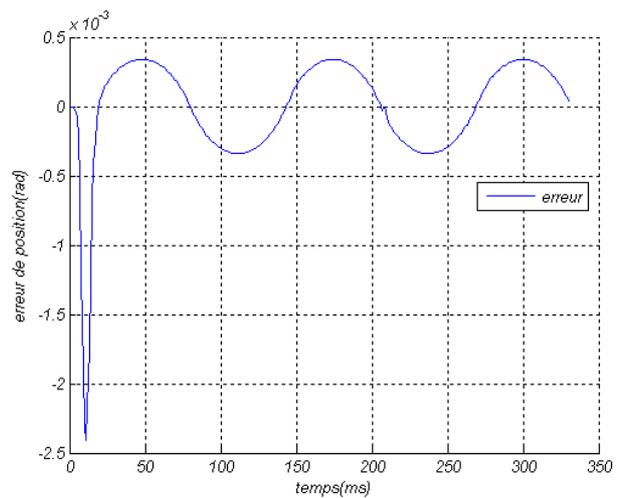


Figure 3.6.1.b : erreur en position.

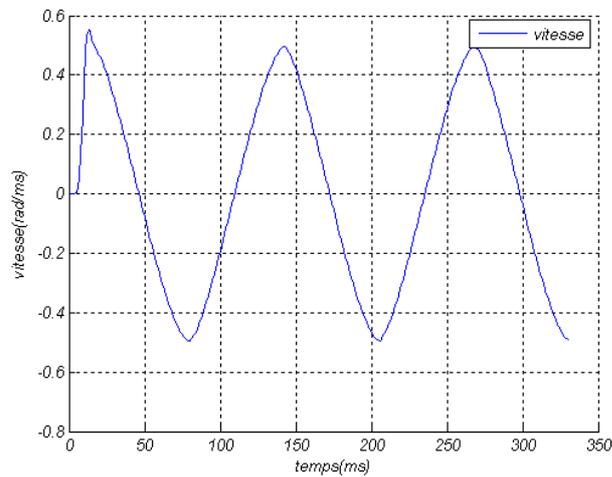


Figure 3.6.1.c : suivi en vitesse.

### **3.2.5. Conclusion**

La commande par mode glissement est très robuste vis-à-vis les erreurs de modélisation, les simulations présentes montrent l'aptitude de poursuite de référence malgré une variation paramétriques de 50%.

## **4. Conclusion**

La commande décentralisée du type PD et PID appliquée sur un robot de type ROB3 a donnée de bons résultat pour le suivi de trajectoire, elle est satisfaisante pour suivre une trajectoire de consigne. Malgré que la commande par mode glissement reste très robuste la commande classique PID reste plus facile à implémenter avec faible cout.

CHAPITRE 4  
*MISE EN OUVRE*  
*PRATIQUE.*

## 4. Mise en œuvre pratique

La commande en mouvement des robots manipulateurs est essentiellement réalisée par des circuits numériques programmables tels que les microprocesseurs, les microcontrôleurs et les DSP où les lois de commande sont implémentées sous forme d'algorithmes, leurs principaux avantages sont la rapidité de développement, leur flexibilité et leur faible espace d'occupation.

Dans ce chapitre, nous présentons la réalisation d'une carte de commande à base d'un microcontrôleur pour l'asservissement en position d'un moteur à courant continue afin de commander le robot ROB3.

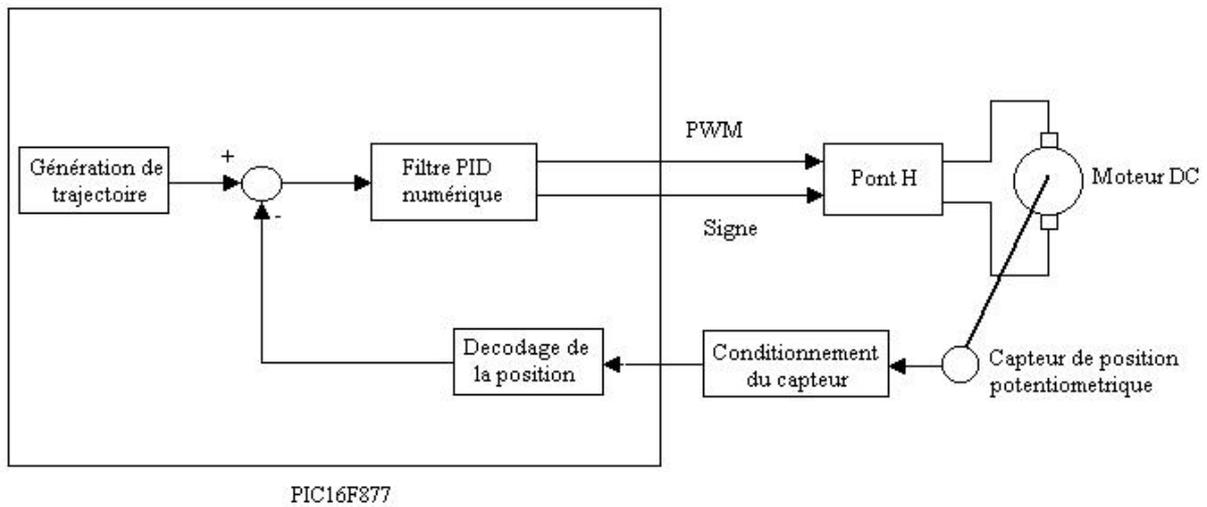
### 4.1. Présentation et principes

Le choix du microcontrôleur PIC16F877 pour réaliser la commande en position d'un moteur à courant continue n'est pas quelconque, ce circuit de la firme Microchip [12], [13] répond aux exigences d'une telle application, nous proposons un circuit d'interfaçage de ce microcontrôleur dont nous détaillerons les circuits indispensables et les différentes ressources dont il a besoin pour cette application.

Comme nous avons pu le constater, le PIC16F877 offre une multitude de fonctions pour effectuer la commande d'un moteur à courant continue telle que :

- Mesure de la position du moteur.
- Calcul du signal d'erreur et implémentation de l'algorithme de régulation (PID).
- Génération du signal de commande du moteur (PWM).

La figure 4.1 représente le synoptique de la carte de commande du moteur à courant continue.



**Figure 4.1 :** Bloc diagramme du système de commande.

## 4.2. Mesure de la position du moteur

Afin de pouvoir agir sur la commande, le calculateur doit disposer de la position actuelle du moteur, étant donnée que les capteurs fournissent des tensions, ces dernières sont connectées aux bornes RA0/AN0, RA1/AN1 et RA2/AN2/VREF-, à partir de ces valeurs obtenues, je détermine la position réelle du moteur.

## 4.3. Implémentation de l'algorithme de régulation

La position réelle et désirée des moteurs connus, l'algorithme de régulation est exécuté pour calculer la prochaine valeur du rapport cyclique. L'algorithme implémenté est un régulateur proportionnel- intégral- dérivé (PID).

Le terme proportionnel fournit une action directement liée à l'erreur, le terme intégral accumule les erreurs en position successives calculées à chaque itération de la boucle de réglage et le terme dérivé correspond à une action qui est liée à la variation de l'erreur en position.

A l'instant  $k$ , on donne [3]:

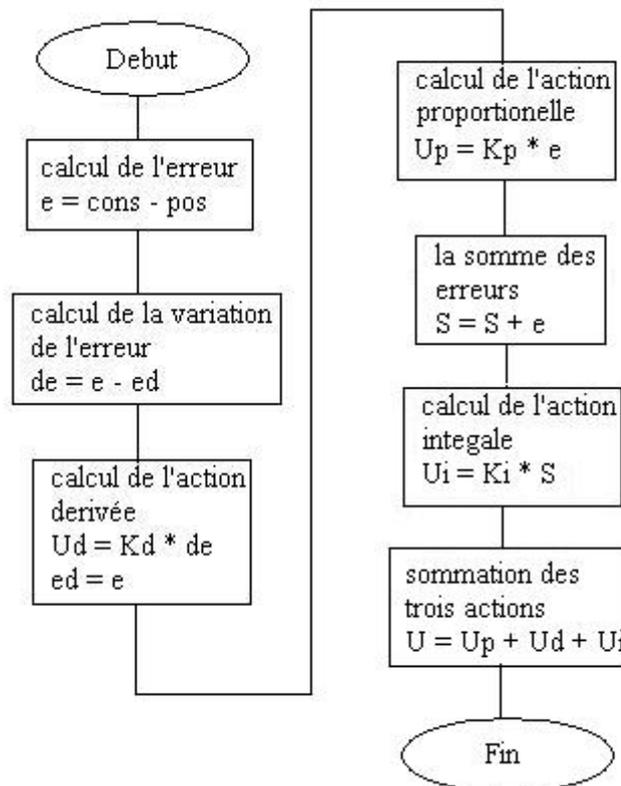
$$U = K_p e[k] + K_i \sum_{i=0}^k e[i] + K_d (e[k] - e[k - 1]) \quad (4.1)$$

Cette expression représente la loi de commande d'un régulateur classique PID, l'erreur en position est en radians et le signal de commande  $U$  est en volts, donc il est nécessaire de transformer cette loi de commande afin quelle soit adaptée à notre calculateur, nous savons que la position du moteur est fournie sous forme de tension linéaire par rapport à la position du moteur. On sait pour un hacheur quatre quadrants que :

$$U = \alpha E. \quad (4.2)$$

On sait du datasheet du microcontrôleur que la valeur du la valeur du rapport cyclique peut être directement donnée on utilise l'instruction `set_pwmX_duty( $\alpha$ )`.

La figure 4.2 donne l'organigramme de l'algorithme de réglage :



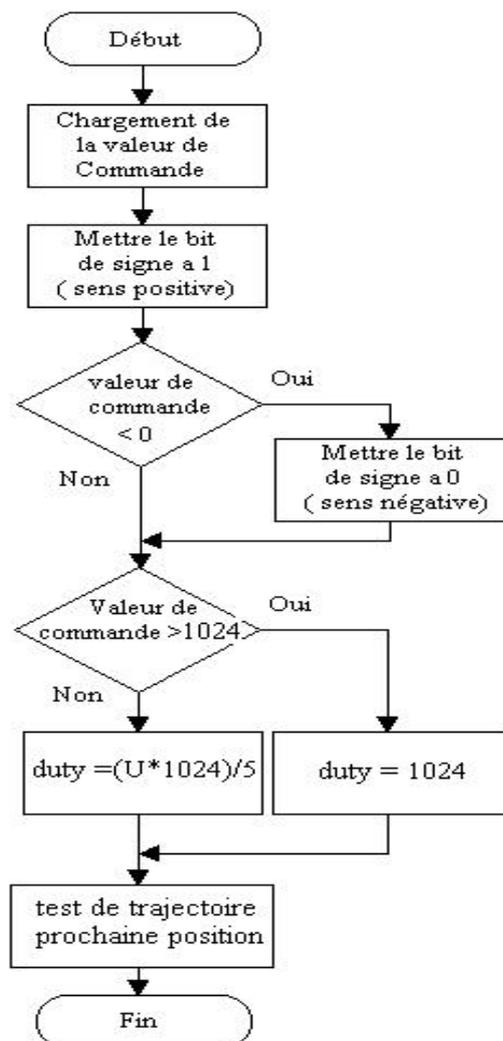
**Figure 4.2 :** Organigramme de l'algorithme de régulation PID.

L'organigramme de l'algorithme de réglage PID est linéaire, au début l'erreur de position  $e$  est calculée, ensuite la variation de l'erreur qui représente la différence entre l'erreur en position actuelle  $e$  et l'erreur précédente  $e_d$  est calculée à son tour, à partir de cette variation la valeur du terme différentiel est déterminée.

Le terme proportionnel ( $U_p$ ) est ensuite calculé, puis nous terminons par le calcul du terme intégral ( $U_i$ ) en sommant les erreurs de position dans une variable  $S$  et en le multipliant par le coefficient correspondant.

#### 4.4. Génération du signal de commande du moteur (PWM)

La commande ayant été calculée, nous obtenons une valeur correspondante au temps de fermeture du circuit de commutation, la valeur de la commande peut prendre des valeurs positives ou des valeurs négatives, mais comme nous utilisons une seule PWM, il est donc nécessaire de trouver une manière de faire fonctionner le moteur dans les deux sens, pour cela nous avons rajouté un circuit logique qui utilise un bit de signe pour inverser la tension de commande.



**Figure 4.3 :** Organigramme de génération du signal de commande.

L'organigramme (Figure 4.3) m'as permet de tester le signe de la valeur de commande, il a pour action de mettre à 1 ou à 0 le bit de signe pour inverser la tension de commande, une saturation de la valeur de commande est envoyée a la valeur maximal de la PWM dans le cas où la valeur de commande calculé est supérieur à 1023.

Après d'avoir envoyé la valeur de commande, le test de la trajectoire sera activé afin de déterminer la prochaine position désirée du moteur.

## 4.5. Module utilisés pour la mise en œuvre pratique

### 4.5.1. Présentation du L298

Le L298 est un circuit intégré monolithique à 15 pins qui existe en boîtier Multiwatt et PowerSO20 [14]. C'est un composant a haut débit de courant avec des tensions élevées, il renferme deux modules pont H et accepte des entrées de commande à niveau logique TTL, il peut piloter des charges inductives, relais, solénoïdes, moteur DC et pas à pas, il a des entrées Enable qui permettent de mettre en service ou non le pont H associer, il a deux sorties à connecter avec une résistance shunte pour relever la valeur du courant et aussi une entrée d'alimentation basse tension supplémentaire pour les portes logiques de commande.

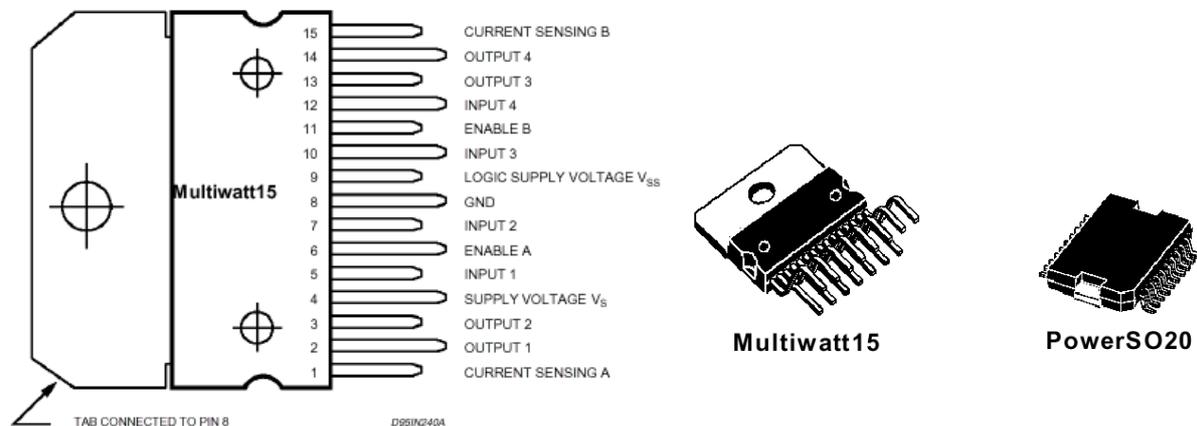


Figure 4.4 : Définition des pins et boîtier externe.

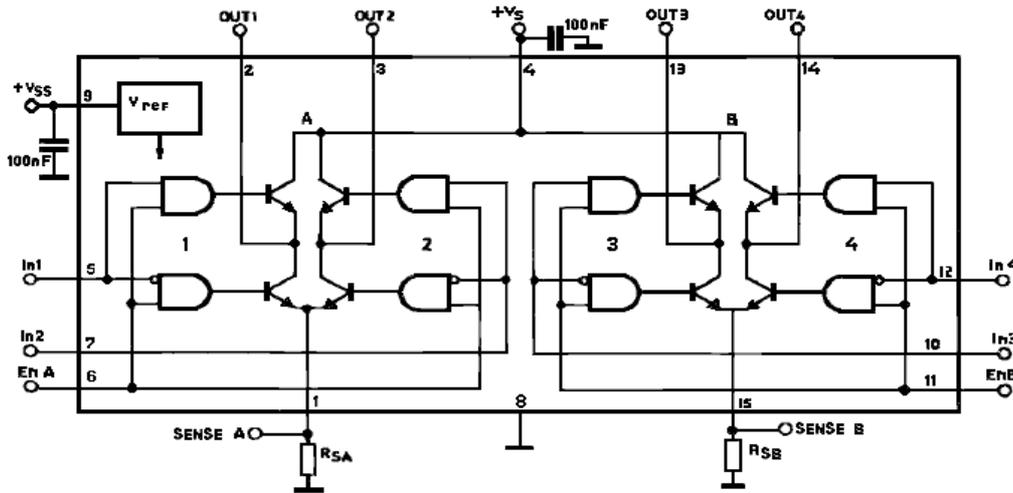


Figure 4.5. : Schéma interne et conditionnement externe des pins.

Symbole	Paramètre	Value	Unit
Vs	Tension d'application	50	V
Vss	Tension logique	7	V
VI , Ven	Tension d'entrée et de validation	-0.3 à 7	V
IO	Courant de sortie (pique pour chaque canal)		
	Non répétitive (t = 100µs)	3	A
	Répétitive (80% on – 20% off ; ton = 10ms)	2.5	A
	DC opérations	2	A
Vsens	Tension relative au courant (courent sense A, B)	-1 à 2.3	V
Ptot	Dissipation totale d'énergie (Tcase = 75°C)	25	W
TOP	Température de fonctionnement des jonctions	-25 à 130	°C
Tstg , TJ	Température de jonction et de stockage	-40 à 150	°C

Tableau 4.1. : Caractéristiques et valeurs maximales absolues.

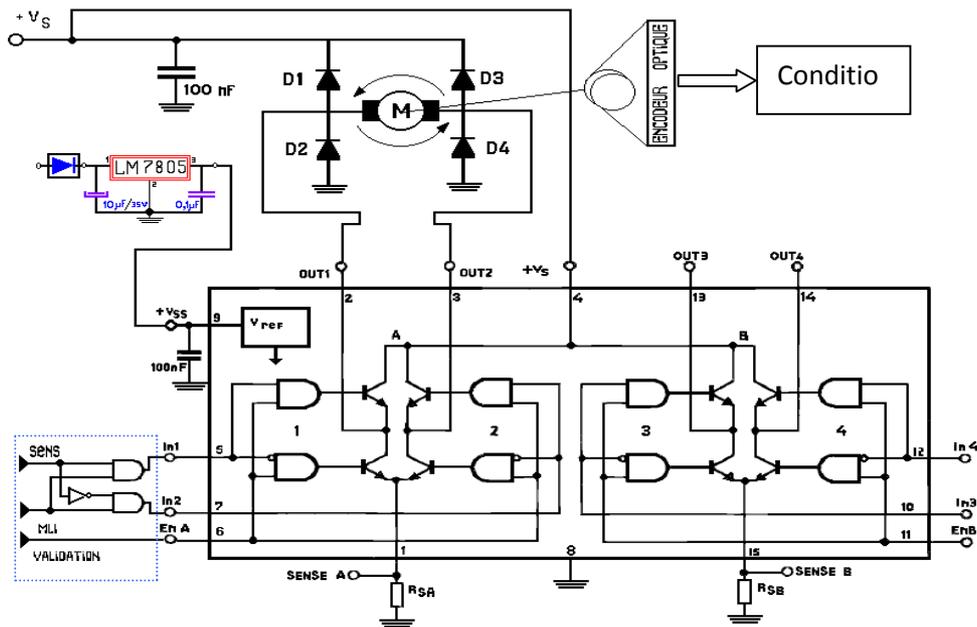


Figure 4.6. : La partie puissance.

Le circuit de puissance doit contenir tout les éléments nécessaire a la commande des ponts H permettant la gestion du sens et transmettre le signale MLI de façon a faire marcher le moteur en deux sens.

4.5.2 Circuit imprimé

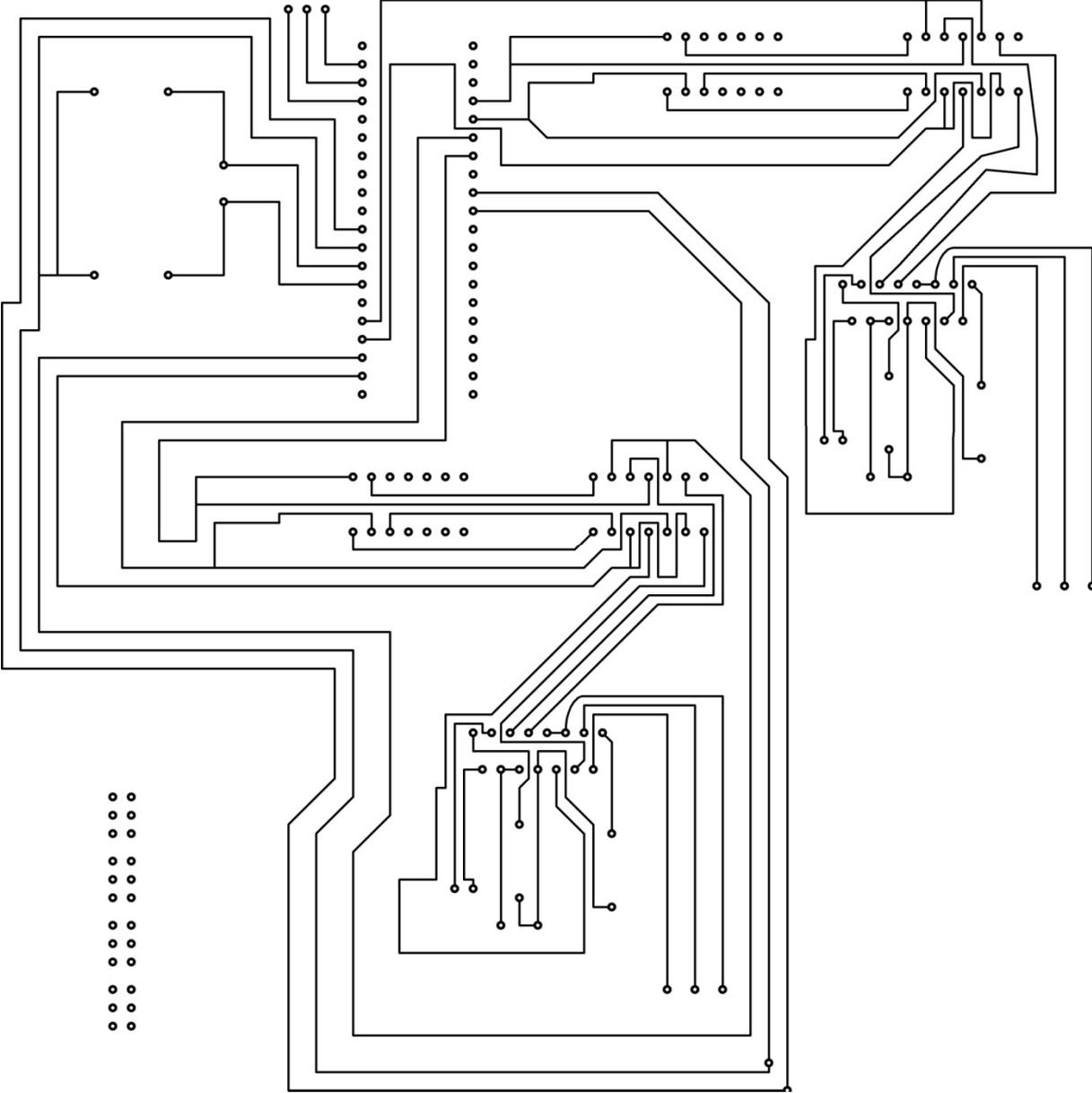


Figure 4.7. : Circuit imprimé de la carte de commande.

## 4.6. Réalisation

Nous avons réalisé le circuit de commande en position d'un moteur à courant continu, le schéma de est représenté sur la figure 4.8 principe et le circuit imprimé est donnée en annexe 2 (figure 4.7)

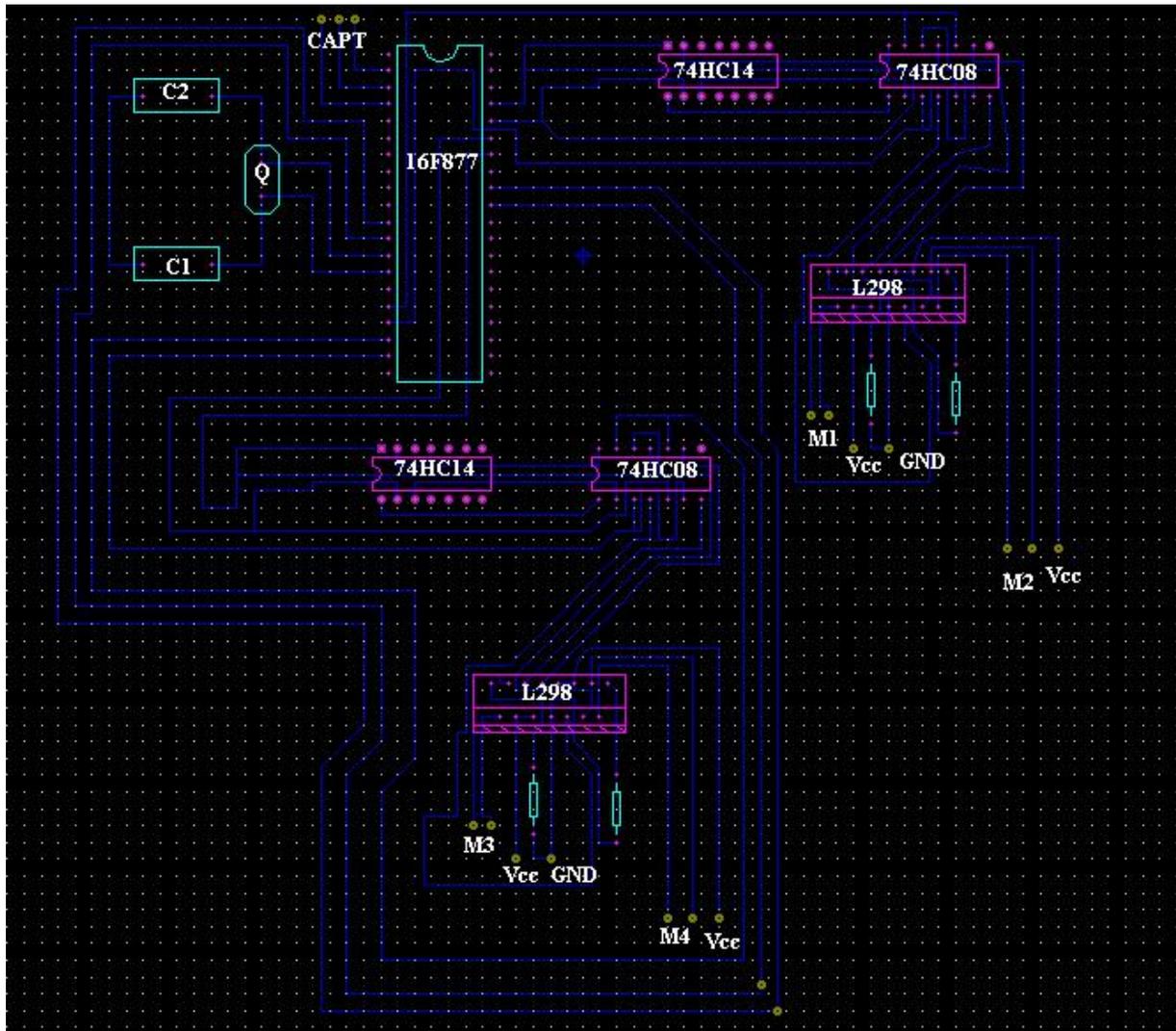
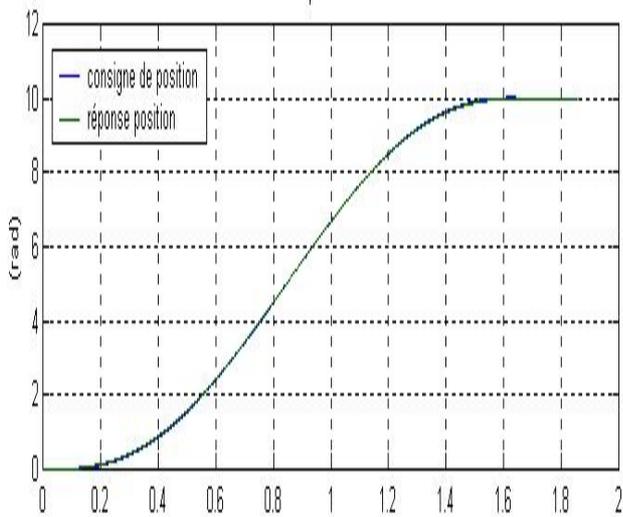


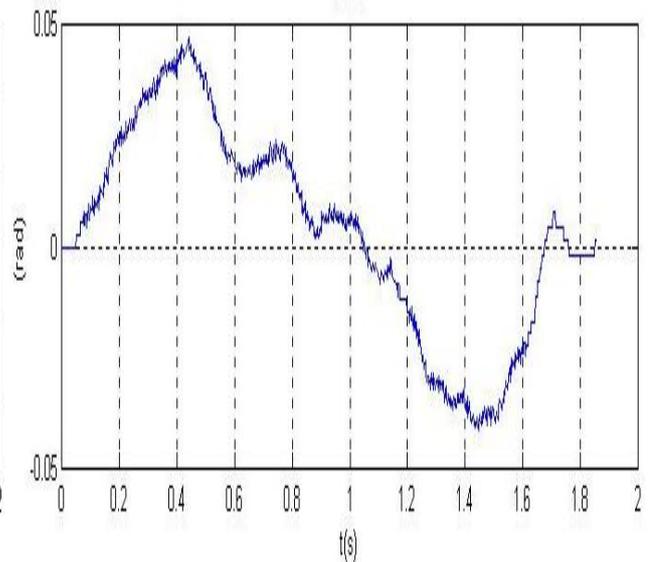
Figure 4.8. : Implémentation des composants sur la carte de commande.

### 4.7. Résultat pratique

Les figures 4.9.a et 4.9.b montrent le suivi en position de l'axe 1 (la base) du bras ROB3, les résultats sont tirés a partir de l'expérimentation.

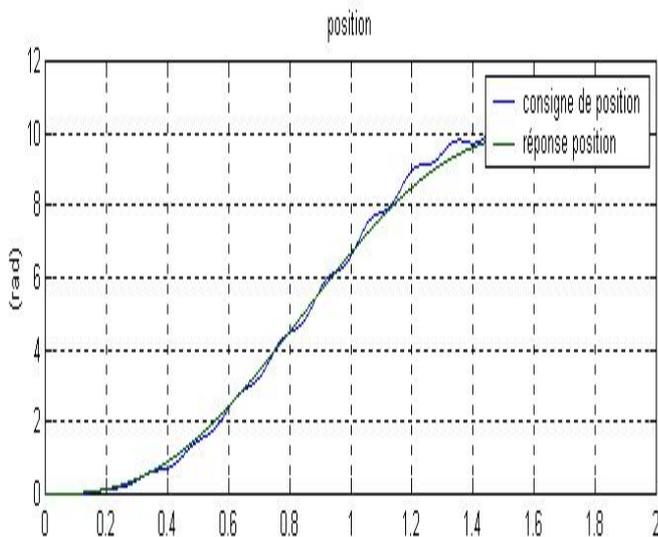


**Figure 4.9.a :** *suivie en position(axe 1).*

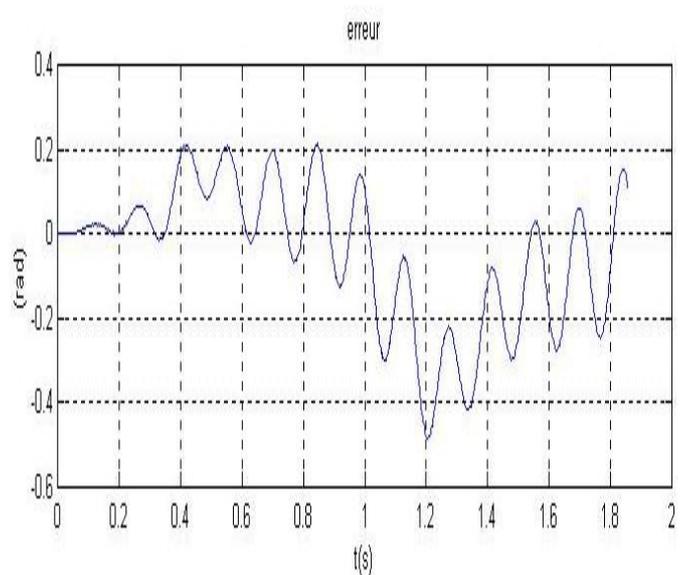


**Figure 4.9.b :** *erreur en position (axe 1).*

Les figures 4.9.c et 4.9.d montrent le suivi en position de l'axe 2 (l'épaule) du bras ROB3.



**Figure 4.9.c :** *suivie en position (axe 2).*



**Figure 4.9.d :** *erreur en position (axe 2).*

#### 4.8. Résultats d'application

Dans cette partie, nous avons calculé les coefficients du régulateur PID à implémenter dans le calculateur pour faire la poursuite en position du moteur, pour notre application nous avons choisit un régulateur PD, ainsi les coefficients obtenus sont :

$$K_v = \begin{bmatrix} 8.8359 & 0 & 0 \\ 0 & 8.1385 & 0 \\ 0 & 0 & 7.9044 \end{bmatrix}; K_p = \begin{bmatrix} 26.9439 & 0 & 0 \\ 0 & 24.1397 & 0 \\ 0 & 0 & 25.1231 \end{bmatrix}.$$

#### 4.9. Conclusion

Dans ce qui a été fait précédemment, nous avons montré comment implémenter la commande en position d'un moteur a courant continue ainsi qu'un algorithme de régulation PID tout en générant la commande sous forme d'une PWM.

CONCLUSION  
*GENERALE*

## Conclusion générale

Dans le cadre de notre travail, constituant à étudier le robot série ROB3. Comme pour tous les robots manipulateurs, nous commençons par donner une description de la plateforme en précisant les propriétés mécaniques, électroniques et l'unité de commande Teachbox, Ainsi que les caractéristiques des actionneurs et des capteurs. Après avoir effectué une modélisation du point de vue géométrique en se basant sur la notation de Denavit-Hartenberg une modélisation dynamique, qu'est fondée sur le formalisme de Lagrange, mais comme cette dernière a été très difficile à élaborer nous avons choisi comme modèle de commande un modèle linéaire d'ordre deux, ce dernier a été obtenu par une méthode d'identification basé sur l'expérimentation. Nous avons appliqué deux types de commande : la commande par mode de glissement et la commande classique (PID). Ces simulations ont été faites sur un modèle de commande linéaire.

Nous avons conçu et réalisé une carte de commande à base du microcontrôleur 16F877 qui effectue l'asservissement en position d'un moteur à courant continu. Cette réalisation, destinée à la commande du robot ROB3, m'a permis de comprendre et de résoudre les problèmes liés à la pratique, tels que la synchronisation, l'élaboration de programme et l'implémentation d'algorithme.

Eu égard à cette modeste expérience que j'ai pu acquérir dans le cadre mon travail de fin d'études, je me fixe comme perspective, de travailler sur l'implémentation des nouvelles techniques de commandes plus performantes, pouvant être utilisées dans le domaine industriels.

*REFERENCES*  
*BIBLIOGRAPHIQUES*

## REFERENCES BIBLIOGRAPHIQUES

- [1] **Christian von der Heyd, Thomas Stöber** « Untersuchungen zur geregelten Positionierung und Bahnplanung eines 5-achsigen Roboterarms über eine SPS »  
Institut für Elektronische Steuerungstechnik, Studienarbeit 2003.
- [2] « TEACHBOX MANUAL IR 50 p », © EUROBTEC GmbH, Zirndorf 1993, 1994.
- [3] **P.Mayé** ; « Moteur électrique pour la robotique »  
© Edition DUNOD, Paris, 2000
- [4] **B. Badji, A. Said** ; « Conception et réalisation d'une carte de commande a base de microcontrôleur pour robot SCARA »  
*E.N.P, Département du génie électrique, Projet de fin d'études, 2003.*
- [5] **W. Khalil, E. Dombre** ; « Modélisation, identification et commande des robot »  
© *HERMES Sciences Publications*, Paris 1988, 1999.
- [6] **R. Murray, Z. Li et S. Sastry** ; «A mathematical introduction to robotic Manipulation»; CRC Press, 1994.
- [7] **M. Renaud** ; « *Calcul analytique itératif des modèles d'un robot manipulateur* » ;  
Rapport technique 86159, LAAS–CNRS, 1986.
- [8] **J. J. Craig** ; « Introduction to robotics : Mechanics and control » ; Addison-Wesley, 1989.
- [9] **R. Paul**; «Robot manipulators: Mathematics, programming, and control». MIT press, Cambridge, Londres, Royaume-Uni, 1981.
- [10] **B.CHERKI** ; « Commande des robots manipulateurs par retour d'état »  
*Thèse doctorat, Ecole Centrale de Nantes, 1996.*
- [11] **T. Benmedakhene** ; « Commande robuste d'un robot à muscles artificiels »  
*E.M.P, Département Control et Commande, Magister, 2008.*