

3/10
REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE
ECOLE NATIONALE POLYTECHNIQUE



PROJET DE FIN D'ETUDES POUR L'OBTENTION
DU DIPLOME
D'INGENIEUR D'ETAT

SPECIALITE : *AUTOMATIQUE*

THEME :

المدرسة الوطنية المتعددة التخصصات
BIBLIOTHEQUE — المكتبة
Ecole Nationale Polytechnique

Commande

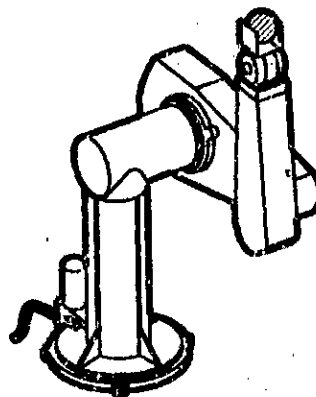
En Position-Orientation
d'Un Robot Manipulateur
Dans
l'Espace Opérationnel

Réalisé Par :

- BENAÏSSA Ismain
- HAMACHE Amar

Encadré Par :

- Mr. H.CHEKIREB
- Mr. M.TADJINE



Année Universitaire 2003 - 2004

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTRE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE
ECOLE NATIONALE POLYTECHNIQUE

PROJET DE FIN D'ETUDES POUR L'OBTENTION
DU DIPLOME
D'INGENIEUR D'ETAT

SPECIALITE : *AUTOMATIQUE*

THEME :

المدرسة الوطنية المتعددة التقنيات
المكتبة —
BIBLIOTHEQUE —
Ecole Nationale Polytechnique

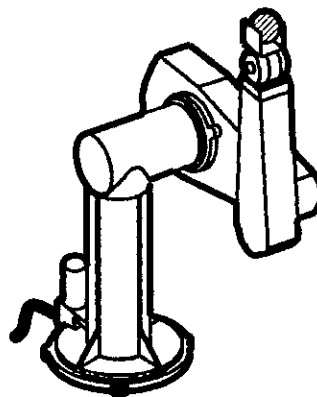
Commande
En Position-Orientation
d'Un Robot Manipulateur
Dans
l'Espace Opérationnel

Réalisé Par :

- BENAÏSSA Ismain
- HAMACHE Amar

Encadré Par :

- Mr. H.CHEKIREB
- Mr. M.TADJINE



Année Universitaire 2003 - 2004

Dédicace

Il me tient à cœur de dire que je ne pourrais jamais assez remercier mes parents qui m'ont aidé le long de mes études, tout particulièrement mon défunt père à qui je dois ce que je suis devenu aujourd'hui. Que Dieu l'accueille dans son vaste paradis.

Je dédie ce modeste travail à :

La mémoire de mon cher et regretté Père.

Celle à qui mon cœur depuis sa naissance n'a pas pu éprouver qu'amour et reconnaissance, à celle qui a donné un sens à mon existence en m'offrant une éducation digne de confiance à ma Mère chérie.

Mes sœurs et frères, nièces et neveux.

Mes amis tout particulièrement Rida et Fayçal.

*Tous ceux qui portent le nom Kamache
en particulier Da Mohamed*



Mourad

DEDICACE

Je dédie ce modeste travail :

*À Celle à qui mon cœur depuis sa naissance n'a pas pu éprouver
qu'amour et reconnaissance, à celle qui a donné un sens à mon
existence en m'offrant une éducation digne de confiance*

à ma chère Mère.

*À mon père, pour sa disponibilité et son soutien moral depuis mon
enfance. Père merci.*

*À tout les membres de ma famille ; mes sœurs, mes frères :
Kaïss, Abderrahmane, et au tétu Houari.*

À la famille TIDJANI tout Particulièrement Hamza.

À tous mes amis de l'université d'Oran.

*À mes amis de la cité universitaire BOURAOUI Amar,
et à tous qui m'ont connu et aidé de près ou de loin dans la réalisation
de ce travail.*

À tous les Kabyles qui me sont chers.



Amain

SOMMAIRE

المدرسة الوطنية المتعددة التقنيات
المكتبة — BIBLIOTHEQUE
Ecole Nationale Polytechnique

INTRODUCTION GENERALE.

CHAPITRE 1

Notion Fondamentale Sur La Robotique

1.1 Introduction.	3
1.2 Constituants mécaniques des robots.	4
1.3 Classification structurelle des robots manipulateurs.	5
1.4 Les différents modèles d'un robot manipulateur.	6
1.5 Définitions générales.	7
1.5.1 Degré de mobilité d'une chaîne cinématique.	7
1.5.2 Articulations.	7
1.5.3 Articulation rotoïde.	7
1.5.4 Articulation prismatique.	7
1.5.5 Espace articulaire.	8
1.5.6 Espace opérationnel.	8
1.5.7 Redondance.	8
1.5.8 Configuration singulière.	8
1.5.9 Morphologie des robots manipulateurs.	9
1.6 Conclusion.	10

CHAPITRE 2

Modélisation Des Robots Manipulateurs

2.1 Introduction.	11
2.2 Modélisation géométrique.	12
2.2.1 Modèle géométrique direct.	12
2.2.1.1 Description de la position.	12
2.2.1.2 Description de l'orientation.	13
2.2.1.3 Paramètres de Denavit-Hartenberg.	15
2.2.2 Modèle géométrique inverse.	17
2.3 Modélisation du robot Puma 560.	19
2.3.1 Modèle géométrique.	19
2.3.2 Modèle géométrique inverse.	22
2.4 Modélisation cinématique.	24
2.4.1 Modélisation cinématique directe.	24
2.4.2 Modélisation cinématique inverse.	28
2.5 Modélisation dynamique.	29
2.5.1 Approche d'Euler-Lagrange.	29
2.5.2 Modèle dynamique.	30
2.6 Conclusions.	32

Chapitre 3

Génération De Trajectoire

3.1 Introduction.	33
3.2 Génération de la trajectoire dans l'espace articulaire.	34
3.2.1 La fonction polynomial cubique.	34
3.2.2 Polynôme de degré cinq.	36
3.2.3 Loi Bang-Bang avec palier de vitesse (loi trapèze).	37
3.3 Génération de la trajectoire dans l'espace cartésien.	40
3.4 Les problèmes géométriques dans un parcours cartésien.	40
3.5 Etude comparative et discussion du choix d'une trajectoire.	43
3.6 Génération de trajectoire dans l'espace cartésien.	44
3.6.1 Description d'une trajectoire planaire.	44
3.6.1.1 Génération de la trajectoire sur le 1 ^{er} tronçon.	45
3.6.1.2 Génération de la trajectoire sur le 2 ^{eme} tronçon.	46
3.6.2 Description d'une trajectoire spatiale.	47
3.7 Conclusion.	49

Chapitre 4

Synthèse De La Commande Par Couple Calculé

4.1 Introduction.	50
4.2 Commande dans l'espace opérationnel avec correction dans l'espace articulaire.	51
4.3 Commande dans l'espace opérationnel avec correction dans l'espace opérationnel.	55
4.4 Etude de la robustesse de la commande par découplage non linéaire.	57
4.5 simulations et interprétations.	59
4.6 Robustification de la commande.	66

Chapitre 5

Synthèse De La Commande Basée Sur Le Principe De Lyapunov

5.1 introduction.	71
5.2 notions préliminaires.	72
5.2.1 Points d'Equilibre.	72
5.2.2 Notions de Stabilité.	73
5.2.3 Stabilité Locale est Globale.	73
5.3 analyse de la stabilité par Lyapunov.	74
5.3.1 La méthode indirecte.	74
5.3.2 La méthode directe.	75
5.3.3 Le principe d'invariance de Lasalle.	76
5.4 Application a la commande des robots.	77
5.4.1 Introduction.	77
5.4.2 Synthèse d'une première loi de commande.	77
5.4.3 Synthèse d'une deuxième loi de commande.	81
5.5 Simulations et interprétations.	83
CONCLUSION GENERALE.	87
REFERENCE BIBLIOGRAPHIQUE.	89
ANNEXE	

INTRODUCTION GENERALE

Etymologiquement, le mot robot a connu plusieurs définitions selon différentes langues, le mot « robot » tire sa racine du Bulgare « robu » qui signifie « serviteur », donnant naissance au mot russe « robota » qui signifie « travail » et au mot tchèque « robota » qui se traduit par « travail forcé ».

Vers les années vingt (1920), l'écrivain tchèque Karel CAPEK a popularisé le terme « robot » à travers une pièce théâtrale intitulée « Rossum's Universal Robots », elle met en scène des petits êtres artificiels qui répondent aux ordres de leurs maîtres comme des esclaves, de cela le mot « robot » a pris sa véritable signification.

Les premières applications de la robotique étaient la saisie et le transfert d'objets, le moulage par injection et l'emboutissage où le robot est simplement équipé d'une presse pour décharger et transférer ou empiler la partie finie. Ces premiers robots étaient capables d'être programmés pour exécuter une séquence de mouvement, comme aller à un emplacement donné, fermer une pince, etc. mais n'avaient pas de capteurs externes. Certaines applications plus compliquées, comme la soudure, le meulage, l'ébavurage, et l'assemblage exigent non seulement un mouvement plus complexe mais également un certain nombre de capteurs externes comme la vision et les capteurs de force vu la forte interaction du robot avec son environnement. Aujourd'hui les robots ont pu remplacé l'homme dans son travail directement en ayant la possibilité d'effectuer la quasi-totalité des travaux industriels.

Il existe d'autres applications de la robotique là où l'utilisation de l'opérateur humain est indésirable comme l'exploration spatiale et sous-marine et le travail dans un environnement radioactif.

Les robots manipulateurs industriels, aujourd'hui, sont en fait des appareils actionnés mécaniquement par des actionneurs électriques, pneumatiques et hydrauliques sous le contrôle d'un ordinateur qui est désormais intégré à toute machine qui se réclame de la robotique moderne, et muni de routines système, dévolues aux fonctions générales du robot et d'un programme relatif aux opérations spécifiques qu'il doit effectuer, ainsi qu'à leur enchaînement. Le programme est développé chez son utilisateur au moyen d'un apprentissage ou par écriture en langage spécialisé.

Les robots manipulateurs industriels se substituent aux êtres humains de façon plus efficace. En effet, ils peuvent travailler sans pause et encore moins la grève. Ils ne connaissent ni la fatigue ni l'ennui.

La problématique de la commande des systèmes non linéaires à grandes dimensions et des systèmes interconnectés que constituent les robots manipulateurs est à l'heure actuelle l'une des préoccupations majeures et l'un des axes de recherche privilégié dans le domaine de la robotique. En effet, la complexité de la dynamique des robots, due aux interconnexions entre les différents axes du robot mais aussi aux variations paramétriques par rapport à la charge manipulée généralement inconnue, rend le contrôle pratiquement impossible par les régulateurs standards (classiques).

Par ailleurs les exigences de l'industrie concernant les performances statiques et dynamiques en rapidité et en précision mais aussi les considérations pratiques font que la production exige des manipulateurs de plus en plus légers, réalisant des tâches à grande vitesse et précision tout en gardant une grande stabilité.

Etant donné que les tâches à exécuter, sont naturellement définies dans notre espace (espace opérationnel), il est très important de signaler l'importance de générer les trajectoires dans ce même espace, ce qui signifie la description en fonction du temps de la position et de l'orientation de l'outil terminal du bras manipulateur qui est l'objet du présent mémoire traitant le problème de la commande en position et orientation d'un bras manipulateur, dont l'application sera faite sur un robot à six degrés de liberté non redondant du type PUMA 560.

Notre mémoire s'articule autour de cinq chapitres :

- Chapitre 1 : représente un ensemble de généralités et de notions de base sur la robotique. Il est aussi l'occasion de préciser le vocabulaire utilisé en robotique.
- Chapitre 2 : consacré à la modélisation géométrique, cinématique et dynamique des robots, basée sur les formalismes de Denavit-Hartenberg et d'Euler-Lagrange. Ainsi qu'une application au PUMA 560.
- Chapitre 3 : dans ce chapitre nous aborderons les techniques de génération des trajectoires et de planification des tâches dans l'espace opérationnel, suivies de deux applications: tâche planaire et tâche spatiale.
- Chapitre 4 : nous entamerons la commande par découplage non linéaire (théorie, simulations et interprétations) puis, nous verrons quelques approches de robustification.
- Chapitre 5 : nous verrons une autre approche de la commande robuste basée sur la théorie de Lyapunov (théorie, simulations et interprétations).

Notions Fondamentales Sur La Robotique

1.1 INTRODUCTION

Les robots manipulateurs ont des morphologies variées ; leurs structures dépendent essentiellement des tâches à accomplir, on peut dire qu'un système robotique est un assemblage de solides (corps) déformables ou non ; dans le cadre de notre étude les solides sont non déformables. Chaque solide est soumis à des actions mécaniques à distance ou de contact. Un mécanisme est un système mécanique particulier, dont tous les corps sont soumis à des actions mécaniques de contact.

Un robot manipulateur, ou plus simplement manipulateur, est une chaîne cinématique possédant deux corps particuliers et des actionneurs. Le premier corps particulier, appelé base, est le socle du robot ; le second, appelé outil terminal, est le support d'un préhenseur ou d'un outil ; le rôle du manipulateur est, en effet, de transporter ou de façonner des objets. Nous ne considérons, par la suite, que des manipulateurs ayant leurs bases fixes et nous pouvons, alors, présenter et utiliser la notion de degré de mobilité. De ce fait, nous supposons que le mouvement du manipulateur est engendré par n actionneurs qui commandent les n paramètres de configuration.

Les robots manipulateurs peuvent être classés en deux catégories : ceux qui ont une structure de chaîne cinématique ouverte et ceux qui ont une structure de chaîne cinématique fermée.

Dans ce chapitre, nous rappelons, les classes des mécanismes des robots manipulateurs. Après cela, nous procédons à la définition de quelques notions de base de la robotique.

1.2 CONSTITUANTS MECANIQUES DES ROBOTS [3]

Un robot manipulateur est constitué par deux sous-ensembles distincts, un (ou plusieurs) organes terminaux et une structure mécanique articulée :

- sous le terme *organe terminal*, on regroupe tout dispositif destiné à manipuler des objets (dispositifs de serrage, dispositifs magnétiques, à dépression ...) ou à les transformer (outils, torche de soudage, pistolet de peinture...). Il s'agit donc d'une interface permettant au robot d'interagir avec son environnement.
- Le rôle de la *structure mécanique articulée* est d'amener l'organe terminal dans une situation (position et orientation) donnée, selon des caractéristiques de vitesse et d'accélération données. Son architecture est une chaîne cinématique de corps généralement rigides, assemblés par des liaisons appelées *articulations*. Les chaînes peuvent être soit *ouvertes simples* (figure 1.1), soit *arborescentes* (figure 1.2), soit *fermées* (figure 1.3).

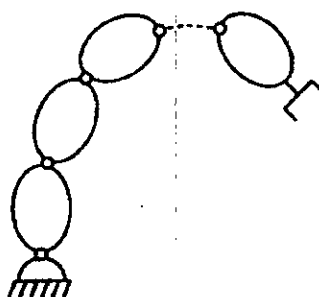


Fig. 1.1. Structure ouverte simple.

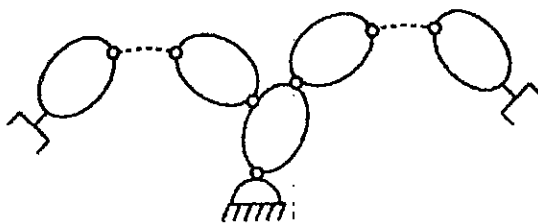


Fig. 1.2. Structure arborescente.

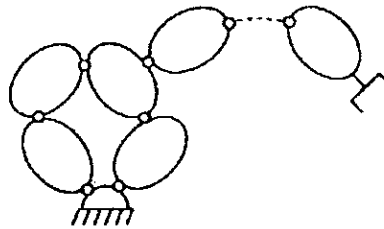


Fig. 1.3. Structure fermée.

La figure 1.4 montre une génération différente des robots – les robots parallèles- dans laquelle l'organe terminal est relié à la base du mécanisme par plusieurs chaînes parallèles.

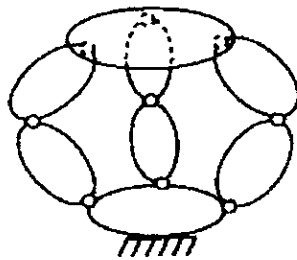


Fig. 1.4. Robot parallèle.

1.3 CLASSIFICATION STRUCTURELLE DES ROBOTS MANIPULATEURS [2]

Il s'agit de l'ensemble mécanique que constitue le support de base du robot, son bras articulé et son poignet ; chacun de ces sous-ensembles possède généralement six degrés de liberté (trois axes de translation et trois axes de rotation).

On classe structurellement les robots en fonction des systèmes de coordonnées dans lesquels ils travaillent :

Cartésien : trois axes de translation.

Cylindrique : deux axes de translation, un axe de rotation.

Sphérique : un axe de translation, deux axes de rotations.

Articulé : trois axes de rotation.

Le système de coordonnées dans lequel il faut travailler dépend souvent de l'application à mettre en œuvre. C'est ainsi qu'un robot cylindrique convient bien à une presse d'estampage, tandis qu'un robot articulé pourra effectuer des soudures à divers endroits invisibles depuis la base du robot.

On rencontre dans la littérature anglo-saxonne une notation abrégée pour décrire le type et le nombre d'articulations d'un robot ; en désignant une translation (sliding) par la lettre S et une rotation par la lettre R, un robot articulé se désignerait sous le sigle «RRR», tandis qu'un cylindrique pourrait répondre à l'appellation «SSR».

1.4 LES DIFFERENTES MODELESATIONS D'UN ROBOT MANIPULATEUR [2] [4]

Un robot manipulateur, quelle que soit la fonction qui lui est attribuée, doit positionner et orienter son organe terminal dans l'espace opérationnel, car le concepteur perçoit de façon naturelle, par la vision, la tâche à accomplir dans cet espace. La difficulté de la commande provient du fait que les actionneurs, dont est doté le robot manipulateur, n'agissent pas directement sur la situation de l'organe terminal, mais agissent sur la configuration du robot manipulateur. D'où la nécessité de définir les passages entre les espaces opérationnel et généralisé (articulaire).

Ainsi nous pouvons distinguer les modèles suivants :

- Le modèle géométrique direct, qui consiste à calculer les coordonnées opérationnelles x , en fonction des coordonnées généralisées q ; ce modèle est noté : $x = f(q)$.
- Le modèle géométrique inverse, qui consiste à calculer les coordonnées généralisées q , en fonction des coordonnées opérationnelles x ; ce modèle est noté : $q = g(x)$.
- Le modèle cinématique direct, qui consiste à calculer les vitesses des coordonnées opérationnelles \dot{x} , en fonction de vitesses des coordonnées généralisées \dot{q} ; ce modèle est noté : $\dot{x} = J \dot{q}$ (J est la matrice jacobienne du robot manipulateur).
- Le modèle cinématique inverse, qui consiste à calculer les vitesses des coordonnées généralisées \dot{q} , en fonction de vitesses des coordonnées opérationnelles \dot{x} ; ce modèle est noté : $\dot{q} = J^{-1} \dot{x}$ (J^{-1} est la matrice jacobienne inverse du robot manipulateur).
- Les modèles dynamiques direct et inverse, qui relient les coordonnées, vitesses et accélérations généralisées aux efforts généralisés (ces efforts généralisés sont les forces appliquées au niveau des liaisons prismatiques ou les couples au niveau des liaisons rotoïdes). Ces modèles tiennent compte des efforts extérieurs appliqués sur l'organe terminal.

1.5 DEFINITIONS GENERALES [3]

1.5.1 Degré de mobilité d'une chaîne cinématique :

Le degré de mobilité d'une chaîne cinématique est le nombre de paramètres indépendants (variables) qui déterminent la géométrie (configuration) de la chaîne. Ces paramètres sont appelés coordonnées généralisées.

1.5.2 Articulations :

Une articulation lie deux corps successifs en limitant le nombre de degrés de liberté de l'un par rapport à l'autre. Soit m le nombre de degrés de liberté résultant, encore appelé *mobilité de l'articulation*. La mobilité est telle que $0 \leq m \leq 6$.

Lorsque $m = 1$, ce qui est le cas le plus fréquent en robotique, l'articulation est soit *rotoïde*, soit *prismatique*.

1.5.3 Articulation rotoïde :

Il s'agit d'une articulation de type pivot réduisant le mouvement entre deux corps à une rotation autour d'un axe qui leur est commun. La situation relative entre les deux corps est donnée par l'angle autour de cet axe. L'articulation rotoïde est représentée par le symbole de la figure 1.5 et est notée R.

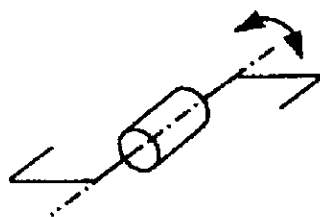


Fig. 1.5. Symbole de l'articulation rotoïde.

1.5.4 Articulation prismatique :

Il s'agit d'une articulation de type glissière réduisant le mouvement entre deux corps à une translation le long d'un axe commun. La situation relative entre les deux corps est mesurée par la distance le long de cet axe. La figure 1.6 donne sa représentation symbolique. On la note P.

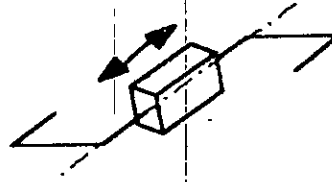


Fig. 1.6. Symbole de la liaison prismatique.

1.5.5 Espace articulaire :

L'espace articulaire d'un robot est celui dans lequel est représentée la situation de tous ses corps. La solution la plus simple consiste à utiliser les *variables ou coordonnées articulaires*. Sa dimension N est égale au nombre de variables articulaires indépendantes et correspond au nombre de degrés de liberté de la structure mécanique.

1.5.6 Espace opérationnel :

L'espace opérationnel est celui dans lequel est représentée la situation de l'organe terminal (on considère donc autant d'espaces opérationnels qu'il y a d'organes terminaux). Sa dimension est égal au nombre de paramètres indépendants nécessaires pour décrire la situation de l'organe terminal dans l'espace. Dans l'espace tridimensionnel ce nombre est de six (trois pour placer un point du corps en un point quelconque de cet espace et trois pour orienter ce corps de façon quelconque).

1.5.7 Redondance :

Un robot est *redondant* lorsque le nombre de degrés de liberté de l'organe terminal est inférieur au nombre de degrés de liberté de l'espace articulaire (nombre d'articulations motorisées). Cette propriété permet d'augmenter le volume du domaine accessible et de préserver les capacités de déplacement de l'organe terminal en présence d'obstacles.

1.5.8 Configuration singulière :

Pour tous les robots, qu'ils soient redondants ou non, il se peut que dans certaines configurations dites *singulières*, le nombre de degrés de liberté de l'organe terminal soit inférieur à la dimension de l'espace opérationnel.

Les configurations singulières ou singularités se traduisent, physiquement, par un comportement anormal du mécanisme et, analytiquement, par la nullité du déterminant de la matrice jacobienne J .

Les singularités sont, relativement, bien abordées, dans le cas des chaînes cinématiques simples et ouvertes ; elles le sont beaucoup moins dans le cas des chaînes cinématiques complexes fermées. Nous exposons les principaux résultats obtenus dans le premier cas (robots séries) et dans un cas particulier du second, à savoir les chaînes parallèles.

1.5.9 Morphologie des robots manipulateurs :

Il existe différentes architectures possibles, on ne considère que deux paramètres : le type d'articulation (rotoïde ou prismatique) et l'angle que font deux axes articulaires successifs (0° ou 90°). Les robots manipulateurs sont composés de deux blocs, le premier appelé *porteur* et le deuxième appelé *poignet* (figure 1.7).

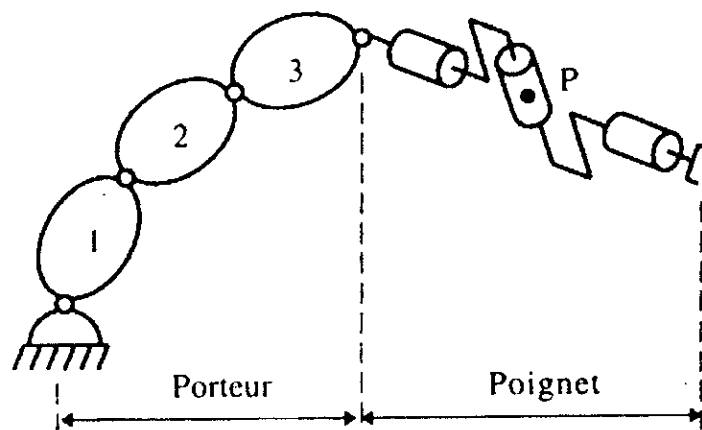


Fig. 1.7 Morphologie des robots manipulateurs

1.6 CONCLUSION

Dans ce chapitre, nous avons rappelé les différentes notions nécessaires pour l'étude des robots manipulateurs et présenté les différentes morphologies des robots- manipulateurs et de leurs constituants.

Pour finir, nous avons précisé la définition de certains termes qui sont nécessaires, pour l'étude des robots manipulateurs, nous présentons dans le chapitre suivant les notions et outils mathématiques mis en œuvre pour la modélisation des robots.

Modélisation Des Robots Manipulateurs

2.1 INTRODUCTION [1] [2] [4] [5]

Dans l'automatique, modéliser un système consiste à établir un ensemble de relations mathématiques qui permettent de décrire avec précision suffisante les interactions entre ce système et son environnement extérieur. Lorsque les relations sous-citées sont issues de la physique, le modèle obtenu est dit modèle de connaissance, ces relations découlent des observations disponibles sur le système. On aboutit aussi au modèle de représentation, en passant par l'identification.

L'utilisation de la paramétrisation de Denavit-Hartenberg (**D-H**) facilite la description géométrique du manipulateur, cette dernière nous permet d'aboutir au modèle cinématique et géométrique direct et inverse du robot. La même transformation offre une souplesse dans le calcul du modèle dynamique direct en utilisant le formalisme d'Euler-Lagrange.

Dans ce chapitre, nous allons introduire la notion de paramétrisation de Denavit-Hartenberg ainsi que le formalisme d'Euler-Lagrange, afin de les utiliser pour le calcul des différents modèles de représentation du robot Puma 560; nous commençons par la modélisation géométrique, après l'obtention du modèle direct, nous présentons une méthode analytique pour le calcul des modèles inverses du robot. Puis, nous passons à la modélisation dynamique, nous présenterons dans un premier temps l'aspect théorique de cette modélisation et par la suite l'application de cette approche sur le robot Puma560.

2.2 MODELISATION GEOMETRIQUE [1]

Dans la modélisation géométrique on s'intéresse au mouvement sans tenir compte des forces qui le provoque. Elle s'intéresse à l'études de la géométrie du robot en vu de décrire ses paramètres géométriques : position et orientation.

2.2.1 Modèle géométrique direct : [1] [3] [7]

Une tache fréquente d'un robot est d'aller prendre une pièce et de l'amener à un autre endroit, lorsque nous le faisons manuellement, c'est très simple car on voit la pièce à prendre et on amène notre main devant.

Pour un robot c'est moins simple, on doit trouver une façon de définir la position de l'objet de manière à ce qu'il sache où elle se trouve par rapport à un système de référence connu. La façon de faire est de fixer un système de coordonnées sur l'objet ainsi que sur l'outil et d'essayer de trouver les transformations nécessaires que doit subir le système de coordonnées de l'outil pour coïncider avec celui de l'objet.

L'étude de la géométrie directe des robots manipulateurs comprend plusieurs étapes pour arriver aux objectifs visés par cette modélisation. Dans un premier temps on fixe les repères aux différentes parties du mécanisme et on décrit les relations entre ces repères, et aussi on localise ces repères lorsque le manipulateur s'articule.

Cette partie est consacrée à la méthode de calcul de la position et de l'orientation de l'élément terminal par rapport à la base du robot en fonction des variables articulaires.

2.2.1.1 Description de la position : [1]

Nous décrivons la position d'un point dans l'espace par un vecteur position. On pourra également écrire le vecteur en coordonnées cylindriques ou sphériques.

Une fois le repère de référence fixé, nous pouvons localiser tout point dans l'espace par un vecteur position de dimension 3×1 . Du fait que nous définissons souvent plusieurs systèmes de coordonnées absolus, les vecteurs doivent porter la marque concernant l'information identifiant le système de coordonnées dans lequel ils sont définis. Les vecteurs écrits avec un indice supérieur indiquent le système de coordonnées dans lequel ils sont référenciés. Par exemple ${}^A P$; signifie que les composantes de ${}^A P$ sont les projections du vecteur sur les axes du système de coordonnées $\{A\}$.

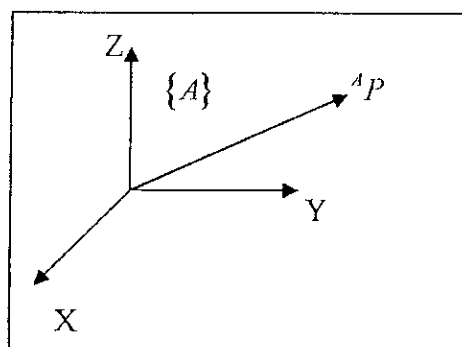


Fig. 2.1 - vecteur relatif à un repère.

La figure 2.1 représente un système de coordonnées, $\{A\}$. Un point ${}^A P$ est représenté par un vecteur et peut être vu similairement comme une position dans l'espace, ou simplement un ensemble ordonné de 3 nombres. Les éléments du vecteur sont notés par les indices x, y et z :

$${}^A P = \begin{bmatrix} P_x \\ P_y \\ P_z \end{bmatrix} \quad (2.1)$$

2.2.1.2 Description de l'orientation : [1]

L'orientation d'un repère est donnée par une matrice 3x3 représentant les 3 vecteurs unitaires. On a cependant indiqué que dans ces 9 valeurs, plusieurs sont redondantes et qu'en fait, il est possible de donner l'orientation en donnant simplement 3 valeurs. Supposons, par exemple, qu'on désire avoir un robot qui suive une certaine trajectoire dans l'espace en lui donnant un certain nombre de points intermédiaires à passer. Si chacun des points est donné par 3 positions et une matrice d'orientation 3x3, cela risque de prendre assez de mémoire. Il est préférable de ne prendre que 3 chiffres pour l'orientation. En fait il y a une infinité de choix possibles pour définir une orientation d'un repère par rapport à un autre. L'idée est de trouver 3 transformations qui vont faire passer le premier dans le deuxième. On peut adopter la paramétrisation d'Euler ZYZ.

Chaque liaison d'un manipulateur fait des rotations ou des translations par rapport au référentiel d'inertie fixe, par exemple un repère fixé à la base du robot. Le calcul des coordonnées des liaisons du manipulateur exprimées dans le référentiel d'inertie de la base est relativement difficile. Cette difficulté augmente suivant l'ordre de la liaison jusqu'à l'élément terminal. Pour ne pas alourdir les calculs et ramener toutes les informations géométriques au repère d'inertie de la base, il est judicieux de les localiser à leurs articulations correspondantes et

situer chaque liaison à son propre référentiel. Le passage d'un référentiel à un autre est garanti par les transformations homogènes.

Lorsqu'on a uniquement des rotations on se satisfait à une matrice de transformation R de troisième ordre, et lorsqu'il existe une translation autour d'un point on est obligé de passer à une matrice de quatrième ordre pour permettre au référentiel de se déplacer à un autre référentiel en translation. Dans ce cas le vecteur de position ${}^A p$ sera augmenté par une quatrième composante pour avoir un vecteur de position ${}^A p'$ exprimé par ses coordonnées homogènes.

$${}^A p = \begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix} \quad (2.2)$$

Le vecteur homogène correspondant est :

$${}^A p' = \begin{bmatrix} p_x \\ p_y \\ p_z \\ 1 \end{bmatrix} \quad (2.3)$$

La matrice augmentée de transformation aura la forme suivante :

$$T = \begin{bmatrix} r_{11} & r_{12} & r_{13} & d_x \\ r_{21} & r_{22} & r_{23} & d_y \\ r_{31} & r_{32} & r_{33} & d_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.4)$$

Avec r_{ij} les composantes de la matrice de rotation R , et d_x, d_y et d_z sont les composantes du vecteur de translation d qui comporte les coordonnées du repère destination dans le repère source. Si on appelle T la matrice de transformation du référentiel (x_1, y_1, z_1) vers le référentiel (x_2, y_2, z_2) alors :

$$p'_{x_2 y_2 z_2} = T \cdot p'_{x_1 y_1 z_1} \quad (2.5)$$

2.2.1.3 Paramètres de Denavit-Hartenberg [6] [1] [3]

Denavit et Hartenberg ont proposé une méthode qui repose sur l'assignation d'un repère unique pour chaque lien. Cette convention est une méthode systématique. Elle permet le passage entre articulations adjacentes d'un système robotique. Elle concerne les chaînes cinématiques ouvertes où l'articulation possède uniquement un degré de liberté, les surfaces adjacentes restent en contact. Le choix adéquat des repères dans les liaisons facilite le calcul des matrices homogènes de Denavit-Hartenberg et permet d'arriver à exprimer rapidement des informations de l'élément terminal dans la base et vice versa.

La figure 2.2 représente l'utilisation de cette notation pour deux liens successifs. L'axe z_i du repère est concourant avec l'axe de l'articulation i . Quant à l'axe x_i , il est sur la droite perpendiculaire aux axes z_{i-1} et z_i . Quatre paramètres sont alors utilisés pour décrire la forme géométrique d'un lien et sa position par rapport au lien précédent.

La notation de Denavit-Hartenberg ne fonctionne que pour des chaînes cinématiques sérielles (pour des chaînes arborescentes des ambiguïtés apparaissent).

Cette notation, apparue très tôt dans le domaine de la robotique est encore largement utilisée par la communauté scientifique pour décrire les robots en vue de leur analyse et/ou modélisation. Quelques variantes relativement proches sont aussi courantes, comme par exemple la notation de Paul. Cette notation se différencie essentiellement de celle de Denavit-Hartenberg par l'assignation des paramètres relativement aux liens (décalage des indices).

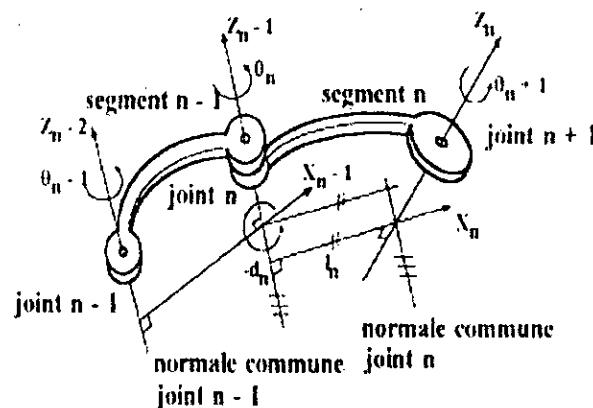


Fig. 2.2: Système de coordonnées et paramètres de Denavit-Hartenberg.

Les étapes à suivre pour cette technique sont les suivantes :

1. Numérotation des segments constitutifs du bras manipulateur de la base vers l'élément terminal. On associe le référentiel zéros à la base de celui-ci, et l'ordre n à l'élément terminal (effecteur) ;
2. Définition des axes principaux de chaque segment :
 - Si z_i et z_{i-1} ne se coupent pas on choisit x_i de manière à être la parallèle avec l'axe perpendiculaire à z_i et z_{i-1} .
 - Si z_i et z_{i-1} sont colinéaires on choisit x_i dans le plan perpendiculaire à z_{i-1} .
3. Fixer les quatre paramètres géométriques : d_i, θ_i, a_i et α_i (voir la figure 2.2) pour chaque articulation tels que :
 - d_i coordonnées de l'origine o_i sur l'axe z_{i-1} . Pour une glissière d_i est une variable et pour une charnière d_i est une constante.
 - θ_i est l'angle que l'on obtient par vissage de x_{i-1} vers x_i autour de l'axe z_{i-1} . Pour une glissière θ_i est une constante et pour une charnière θ_i est une variable.
 - a_i est la distance entre les axes z_i et z_{i-1} mesurée sur l'axe x_i négatif à partir de son origine jusqu'à l'intersection avec l'axe z_{i-1} .
 - α_i est l'angle entre l'axe z_i et z_{i-1} obtenu en vissant z_{i-1} vers z_i autour de x_i .
4. On forme enfin, la matrice homogène de Denavit-Hartenberg de déplacement qui lie la rotation et la translation. La partie supérieure gauche définit la matrice de rotation R_{i-1}^i , et le vecteur de translation à droite d_{i-1}^i .

Par la suite on aboutit à la matrice de transformation de Denavit-Hartenberg suivante:

$$T_i^{i-1} = \begin{bmatrix} \cos \theta_i & -\sin \theta_i & 0 & a_{i-1} \\ \sin \theta_i \cos \alpha_{i-1} & \cos \theta_i \cos \alpha_{i-1} & -\sin \alpha_{i-1} & -\sin \alpha_{i-1} d_i \\ \sin \theta_i \sin \alpha_{i-1} & \cos \theta_i \sin \alpha_{i-1} & \cos \alpha_{i-1} & -\cos \alpha_{i-1} d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.6)$$

Finalement on peut écrire le modèle géométrique direct sous la forme :

$$X=f(q) \quad (2.7)$$

Avec $X \in \mathbb{R}^6$ les coordonnées cartésiennes, et $q \in \mathbb{R}^6$ les coordonnées articulaires.

On va admettre quelques hypothèses dans le but de simplifier la modélisation des robots. Ces hypothèses sont les suivants :

- Les liaisons du manipulateur sont rigides.
- Les jeux dans les articulations sont négligeables.
- Les capteurs ont un gain unitaire et de dynamique négligeable.

2.2.2 Modèle géométrique inverse [1] [4] [5]

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.

Le problème posé est le suivant : étant donné la position et l'orientation de l'outil par rapport à la station de travail, comment calculer l'ensemble des angles articulaires qui accomplissent cet objectif ? La réponse à cette question constitue le modèle géométrique inverse du robot manipulateur.

La solution du problème, concernant la recherche des angles articulaires nécessaires pour positionner le repère de l'outil, par rapport au repère de la station de travail, est décomposé en deux parties. En premier lieu, sont déterminées les transformations nécessaires pour trouver le repère du poignet, par rapport au repère de la base et après, le modèle géométrique inverse est utilisé pour trouver les angles des articulations.

Une approche analytique a été utilisée pour le robots de type série [Craig], elle consiste à éliminer à chaque étape une des coordonnées généralisées (articulaires) par la multiplication de la matrice de transfert finale T_6^0 par les matrices de transformation intermédiaires.

Pour le cas d'un robot à six degrés de liberté la matrice de transformation T_6^0 s'écrit sous la forme :

$$T_6^0 = \begin{bmatrix} r_{11} & r_{12} & r_{13} & d_x \\ r_{21} & r_{22} & r_{23} & d_y \\ r_{31} & r_{32} & r_{33} & d_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = T_1^0(\theta_1).T_2^1(\theta_2).T_3^2(\theta_3).T_4^3(\theta_4).T_5^4(\theta_5).T_6^5(\theta_6) \quad (2.8)$$

Par exemple la première étape sera comme suit :

On multiplie de part et d'autre par la matrice $T_1^0(\theta_1)$, le résultat sera :

$$[T_1^0]^{-1}.T_6^0 = T_6^1 \quad (2.9)$$

On s'intéresse toujours à la dernière colonne de la matrice, qui contient à chaque étape les équations découplées qui permettent de résoudre le problème du modèle géométrique inverse.

Toutefois le modèle géométrique comporte aussi des inconvénients:

- 1) la non unicité du modèle géométrique inverse implique qu'il existe plusieurs "chemins" pour se rendre d'un point à un autre.
- 2) le traitement par incrément peut amener à des imprécisions.
- 3) des singularités, mécaniques et/ou mathématiques apparaissent.

Ces inconvénients sont évités ou contournés de la manière suivante:

- 1) Le problème des multiples solutions du modèle inverse n'intervient pas. En effet, le robot possède une configuration initiale connue, et se rend à une autre position à partir de celle-ci. Il ne sert à rien de calculer des configurations du robot qui seraient impossibles à atteindre depuis la position courante.
- 2) Une haute précision des solutions obtenues n'est pas nécessaire puisqu'il suffit de fournir à l'utilisateur une vision globale, le calcul des accroissements est à chaque fois effectué à partir d'une nouvelle configuration exacte du robot.
- 3) Quant au problème des singularités, il existe plusieurs méthodes mathématiques pour les traiter ou les éviter.

2.3 MODELISATION DU ROBOT PUMA 560

2.3.1 Modèle géométrique [6]

Le bras manipulateur PUMA 560 possède six degrés de liberté caractérisées par des mouvements de rotations ; Les trois premières articulations de ce manipulateur [Waist , Shoulder, Elbow] caractérisent pour la première une rotation autour d'un axe vertical, la seconde et la troisième suivant deux axes horizontaux dont les mouvements sont identifiés par les variables q_1, q_2 et q_3 . Les trois dernières articulations [Wrist rotation, Wrist bend, Flange] qui constituent le poignet du bras de robot sont caractérisées par les variables de rotation q_4, q_5, q_6 .

La représentation du bras de robot PUMA 560 suivant la convention de Denavit-Hartenberg est représentée dans la figure 2.3.

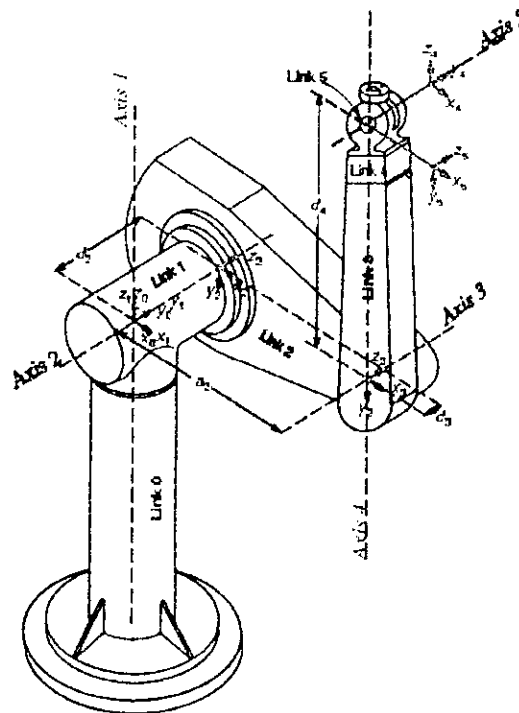


Fig. 2.3: Système de coordonnées des liaisons et paramètres des articulations pour le bras de robot PUMA 560 à sa position d'origine.

En respectant la position d'origine, la définition des repères des liaisons présentées par la figure 2.3, les paramètres du bras de robot PUMA 560 suivant la convention de Denavit et Hartenberg sont représentés dans le tableau 2.1 [6].

i	α_{i-1}	θ_i	a_{i-1}	d_i
Numéro de la liaison	[degrés]	variable	[mètres]	[mètres]
1	0	q_1	0	0
2	-90	q_2	0	0.2435
3	0	q_3	0.4318	-0.0934
4	90	q_4	-0.0203	0.4331
5	-90	q_5	0	0
6	90	q_6	0	0

Tableau 2.1 : Paramètres de Denavit-Hartenberg du bras de robot PUMA 560.

La distance d_6 n'est pas représentée dans le tableau 2.1. Cette distance varie suivant l'effecteur utilisé pour l'application (l'effecteur est l'outil attaché au poignet sur la dernière articulation du robot pour la manipulation des objets). Dans notre application, la distance entre la fin de l'effecteur et l'axe du poignet est supposée égale à $L_6=0.05625$ m ($d_6=5.625$ cm).

Les transformations appropriées en utilisant le formalisme de Denavit-Hartenberg pour les trois premières articulations sont :

$$T_1^0 = \begin{bmatrix} c_1 & -s_1 & 0 & 0 \\ s_1 & c_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \hline 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_2^1 = \begin{bmatrix} c_2 & -s_2 & 0 & 0 \\ 0 & 0 & 1 & d_2 \\ -s_2 & -c_2 & 0 & 0 \\ \hline 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_3^2 = \begin{bmatrix} c_3 & -s_3 & 0 & a_2 \\ s_3 & c_3 & 0 & 0 \\ 0 & 0 & 1 & d_3 \\ \hline 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_4^3 = \begin{bmatrix} c_4 & -s_4 & 0 & a_3 \\ 0 & c_3 & -1 & -d_4 \\ s_6 & c_6 & 0 & 0 \\ \hline 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_5^4 = \begin{bmatrix} c_5 & -s_5 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -s_5 & -c_5 & 0 & 0 \\ \hline 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_6^5 = \begin{bmatrix} c_6 & -s_6 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ s_6 & c_6 & 0 & 0 \\ \hline 0 & 0 & 0 & 1 \end{bmatrix}$$

Avec $c_i = \cos \theta_i$; $s_i = \sin \theta_i$; $c_{ij} = \cos(\theta_i + \theta_j)$; $s_{ij} = \sin(\theta_i + \theta_j)$.

On peut calculer la matrice de transformation 0T pour pouvoir exprimer la position et l'orientation d'un objet dans l'espace dans un référentiel lié à la base, en utilisant la relation suivante :

$$T_6^0 = \begin{bmatrix} r_{11} & r_{12} & r_{13} & d_x \\ r_{21} & r_{22} & r_{23} & d_y \\ r_{31} & r_{32} & r_{33} & d_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = T_1^0(\theta_1).T_2^1(\theta_2).T_3^2(\theta_3).T_4^3(\theta_4).T_5^4(\theta_5).T_6^5(\theta_6)$$

Avec :

$$r_{11} = [(c_1 c_{23} c_4 - s_1 s_4) c_5 - c_1 s_{23} s_5] c_6 - (c_1 c_{23} s_4 + s_1 s_4) s_6 \quad (2.9)$$

$$r_{12} = -[(c_1 c_{23} c_4 - s_1 s_4) c_5 - c_1 s_{23} s_5] s_6 - (c_1 c_{23} s_4 + s_1 s_4) c_6 \quad (2.10)$$

$$r_{13} = [(c_1 c_{23} c_4 - s_1 s_4) s_5 + c_1 s_{23} s_5] \quad (2.11)$$

$$r_{21} = [(s_1 c_{23} c_4 + c_1 c_4) c_5 - s_1 s_{23} s_5] c_6 + (-s_1 c_{23} s_4 + c_1 s_4) s_6 \quad (2.12)$$

$$r_{22} = -[(s_1 c_{23} c_4 + c_1 s_4) c_5 - s_1 s_{23} s_5] s_6 + (-s_1 c_{23} s_4 + c_1 c_4) c_6 \quad (2.13)$$

$$r_{23} = [(s_1 c_{23} c_4 + c_1 s_4) s_5 + s_1 s_{23} c_5] \quad (2.14)$$

$$r_{31} = [-s_{23} c_4 c_5 + c_{23} s_5] c_6 + s_{23} s_4 s_6 \quad (2.15)$$

$$r_{32} = -[-s_{23} c_4 c_5 + c_{23} s_5] s_6 + s_{23} s_4 c_6 \quad (2.16)$$

$$r_{33} = s_{23} c_4 c_5 + c_{23} c_5 \quad (2.17)$$

$$d_x = a_2 c_1 c_2 - (d_2 + d_3) s_1 + a_3 c_1 c_{23} + d_4 c_1 s_{23} \quad (2.18)$$

$$d_y = a_2 s_1 c_2 + (d_2 + d_3) c_1 + a_3 s_1 c_{23} + d_4 s_1 s_{23} \quad (2.19)$$

$$d_z = -a_2 s_2 - a_3 s_{23} - d_4 c_{23} \quad (2.20)$$

2.3.2 Modèle géométrique inverse [1] [4]

La portée de la charge est caractérisée par un point acentrique dans l'espace qui définit une position en coordonnées cartésiennes et donne à l'élément terminal la position et l'orientation pour manipuler les objets. La géométrie inverse est imposée. Pour cela nous pouvons citer quelques solutions qui existent :

- Méthode analytique [Craig 89],
- Transformation inverse, par Paul et Al 1981.
- Approche géométrique, par Lee et Ziegler 1984.
- Méthode des matrices duelles, par Denavit 1956.
- Quaternions duaux, par Yang et Arenden Stein 1964.

Nous nous intéressons à la méthode analytique.

On a :

$$T_0^6 = T_0^1 \times T_1^6 \quad (2.21)$$

$$T_0^6 = \begin{bmatrix} r_{11} & r_{12} & r_{13} & p_x \\ r_{21} & r_{22} & r_{23} & p_y \\ r_{31} & r_{32} & r_{33} & p_z \\ \hline 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.22)$$

On veut calculer le modèle géométrique inverse par rapport au poignet, puisque pratiquement il y a toujours l'outil lié au poignet alors on fait le changement de variable suivant pour avoir la bonne transformation.

On pose :

$$p_x = p_x - r_{13} L_6 \quad (2.23)$$

$$p_y = p_y - r_{23} L_6 \quad (2.24)$$

$$p_z = p_z - r_{33} L_6 \quad (2.25)$$

Avec : r_{13}, r_{23}, r_{33} les orientations autour l'axe Z, et L_6 .

En utilisant les transformations de Denavit-Hartenberg déjà calculées précédemment on trouve le modèle géométrique inverse :

$$q_1 = A \tan 2(p_y, p_x) - A \tan 2\left(d_2 + d_3, \pm \sqrt{p_x^2 + p_y^2 - (d_2 + d_3)^2}\right) \quad (2.26)$$

$$q_3 = -A \tan 2(a_3, d_4) + A \tan 2\left(k, \pm \sqrt{a_3^2 + d_4^2 - k^2}\right) \quad (2.27)$$

Avec

$$k = (p_x^2 + p_y^2 + p_z^2 - a_2^2 - a_3^2 - d_4^2 - (d_2 + d_3)^2) / 2a_2 \quad (2.28)$$

$$k_1 = \frac{[-(a_3 + a_2)p_z + (c_1 p_x + s_1 p_y)(a_2 s_3 + d_4)]}{[p_z^2 + (c_1 p_x + s_1 p_y)^2]} \quad (2.29)$$

$$k_2 = \frac{[(d_4 + a_2 s_3)p_z + (c_1 p_x + s_1 p_y)(a_2 c_3 + a_3)]}{[p_z^2 + (c_1 p_x + s_1 p_y)^2]} \quad (2.30)$$

$$q_2 = A \tan 2(k_1, k_2) - q_3 \quad (2.31)$$

$$k_3 = -s_1 r_{13} + c_1 r_{23} \quad (2.32)$$

$$k_4 = c_1 c_{23} r_{13} + s_1 c_{23} r_{23} - s_{23} r_{33} \quad (2.33)$$

$$q_4 = A \tan 2(k_3, k_4) \quad (2.34)$$

$$k_5 = r_{13} [c_1 k_2 c_4 - s_1 s_4] + r_{23} [s_1 k_2 c_4 + c_1 s_4] + r_{33} k_1 c_4 \quad (2.35)$$

$$k_6 = r_{13} c_1 s_{23} + r_{23} s_1 s_{23} + r_{33} c_{23} \quad (2.36)$$

$$q_5 = A \tan 2(k_5, k_6) \quad (2.37)$$

$$k_7 = -r_{11} [c_1 c_{23} s_4 + s_1 c_4] + r_{21} [-s_1 c_{23} s_4 + c_1 c_4] + r_{31} s_{23} s_4 \quad (2.38)$$

$$k_8 = -r_{12} [c_1 c_{23} s_4 + s_1 c_4] + r_{22} [-s_1 c_{23} s_4 + c_1 c_4] + r_{32} s_{23} s_4 \quad (2.39)$$

$$q_6 = A \tan 2(k_7, k_8) \quad (2.40)$$

2.4 MODELISATION CINEMATIQUE [1]

Le modèle cinématique est, littéralement, un modèle des vitesses. Il exprime les relations entre les vitesses articulaires de chaque joint et les vitesses cartésiennes d'un point de la chaîne cinématique, généralement l'organe terminal. Ce modèle est donc un modèle par accroissements élémentaires: chaque variation élémentaire de la valeur d'une articulation implique une variation de position de l'organe terminal, et inversement. Lorsque ces variations infinitésimales sont exprimées par rapport au temps, on peut les considérer comme des vitesses.

Le modèle cinématique permet donc non seulement de compléter éventuellement le modèle géométrique en tenant compte des vitesses, mais aussi de remplacer le modèle géométrique: en agissant par accroissements successifs, on peut se déplacer d'un point donné à un autre.

Le modèle cinématique possède une propriété essentielle: il est une différentiation du modèle géométrique. Il est donc une linéarisation du système d'équations non linéaires représentant le modèle géométrique. Par conséquent on peut toujours facilement obtenir les transformations inverses puisqu'elles proviennent de l'inversion d'un problème linéaire.

2.4.1 Modélisation cinématique directe [1] [2]

Le modèle cinématique directe d'un robot-manipulateur décrit les vitesses des coordonnées opérationnelles en fonction des vitesses articulaires. Il est noté : $\dot{x} = J(q)\dot{q}$

Où $J(q)$ désigne la matrice jacobienne de dimension $(m \times n)$ du mécanisme, égale à $\frac{\partial X}{\partial q}$ et est

fonction de la configuration articulaire q . La même matrice jacobienne intervient dans le calcul du modèle différentiel direct qui donne les variations élémentaires dx des coordonnées opérationnelles en fonction des variations dq , soit : $dx = J(q)dq$

Soit : $\dot{q}_d = \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \\ \dot{q}_4 \\ \dot{q}_5 \\ \dot{q}_6 \end{bmatrix}_d$ le vecteur des vitesses articulaires désirées.

Pour les vitesses cartésiennes on a :

$$\begin{bmatrix} \dot{p}_x \\ \dot{p}_y \\ \dot{p}_z \end{bmatrix}_d \Rightarrow \begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix}_d$$

Et pour les vitesses angulaires $\Omega_x, \Omega_y, \Omega_z$ on fait une transformation à l'aide de la paramétrisation d'Euler pour trouver les vitesses réelles.

$$\begin{bmatrix} \dot{\alpha} \\ \dot{\beta} \\ \dot{\gamma} \end{bmatrix}_d \Rightarrow \begin{bmatrix} \Omega_x \\ \Omega_y \\ \Omega_z \end{bmatrix}_d = \underbrace{\begin{bmatrix} 0 & -s_\alpha & c_\alpha s_\beta \\ 0 & c_\alpha & s \\ 1 & 0 & c_\beta \end{bmatrix}}_R \begin{bmatrix} \dot{\alpha} \\ \dot{\beta} \\ \dot{\gamma} \end{bmatrix}_d \quad (2.41)$$

Où : α, β, γ sont les angles d'Euler qui définissent l'orientation selon la convention ZYZ.

(Pour ce qui des angles d'Euler, voir l'annexe)

Finalement on peut écrire le modèle cinématique sous la forme :

$$\begin{bmatrix} v_x \\ v_y \\ v_z \\ \Omega_x \\ \Omega_y \\ \Omega_z \end{bmatrix} = J \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \\ \dot{q}_4 \\ \dot{q}_5 \\ \dot{q}_6 \end{bmatrix} \quad (2.42)$$

Où J est la matrice jacobienne.

La matrice jacobienne :

L'outil principalement utilisé pour traiter le problème de la cinématique des robots est la matrice jacobienne. Elle représente un opérateur permettant de lier les vitesses des corps d'un robot exprimées dans différents espaces vectoriels.

L'intérêt de la matrice jacobienne est multiple :

- elle est à la base du modèle cinématique inverse, permettant de calculer une solution locale des variations articulaires dq connaissant les variations opérationnelles dx ;
- en statique, on utilise le jacobien pour établir la relation liant les efforts exercés par l'organe terminal sur l'environnement aux forces et couples des actionneurs ;
- elle facilite le calcul des singularités et de la dimension de l'espace opérationnel accessible du robot.

Calcul de la matrice jacobienne [3]

Il existe plusieurs méthodes pour calculer la matrice jacobienne. Si on écarte les méthodes de calcul symbolique permettant de dériver les équations du modèle géométrique direct, il faut calculer les influences de chaque articulation sur l'organe terminal. Cette approche, très répandue pour le calcul cinématique, peut s'effectuer suivant différents moyens qui sont plus ou moins efficaces au niveau des algorithmes et du nombre d'opérations nécessaires. Ces méthodes se caractérisent par le repère dans lequel la matrice jacobienne est exprimée ainsi que par le corps auquel elle correspond, c'est-à-dire par le choix de i et j dans le calcul de J_{ij} : matrice jacobienne du corps j exprimée dans le repère i .

La plupart du temps, on calcule la matrice jacobienne relativement à l'organe terminal E , elle est aussi souvent exprimée dans le repère de base R_0 que dans le repère de l'organe terminal R_E .

Calcul de la matrice jacobienne par dérivation du MGD :

Le calcul de la matrice jacobienne peut se faire en dérivant le MGD, $x = f(q)$, à partir de la relation :

$$J_{ij} = \frac{\partial f_i(q)}{\partial q_j} \quad i = 1, \dots, m ; j = 1, \dots, n \quad (2.43)$$

Où : J_{ij} est l'élément (i,j) de la matrice jacobienne J .

Cette méthode est facile à mettre en œuvre pour les robots à deux ou trois degrés de liberté, mais elle devient de plus en plus compliquée dans le cas du manipulateur à six degré de liberté.

Calcul de la matrice jacobienne par la méthode de W. Khalil : [3]

Considérons la $K^{\text{ième}}$ articulation d'une chaîne articulée. La vitesse \dot{q}_k induit sur le repère terminal une vitesse de translation $V_{k,n}$ et une vitesse de rotation $\omega_{k,n}$. Deux cas se présentent :

-si l'articulation est prismatique ($\sigma_k = 1$)

$$\begin{cases} V_{k,n} = a_k \dot{q}_k \\ \omega_{k,n} = 0 \end{cases} \quad (2.44)$$

Où le a_k est le vecteur unitaire porté par l'axe z_k de l'articulation k ;

-si l'articulation est rotoïde ($\sigma_k = 0$)

$$\begin{cases} V_{k,n} = a_k \dot{q}_k \times L_{k,n} \\ \omega_{k,n} = a_k \dot{q}_k \end{cases} \quad (2.45)$$

Pour notre cas on prend une articulation rotoïde donc :

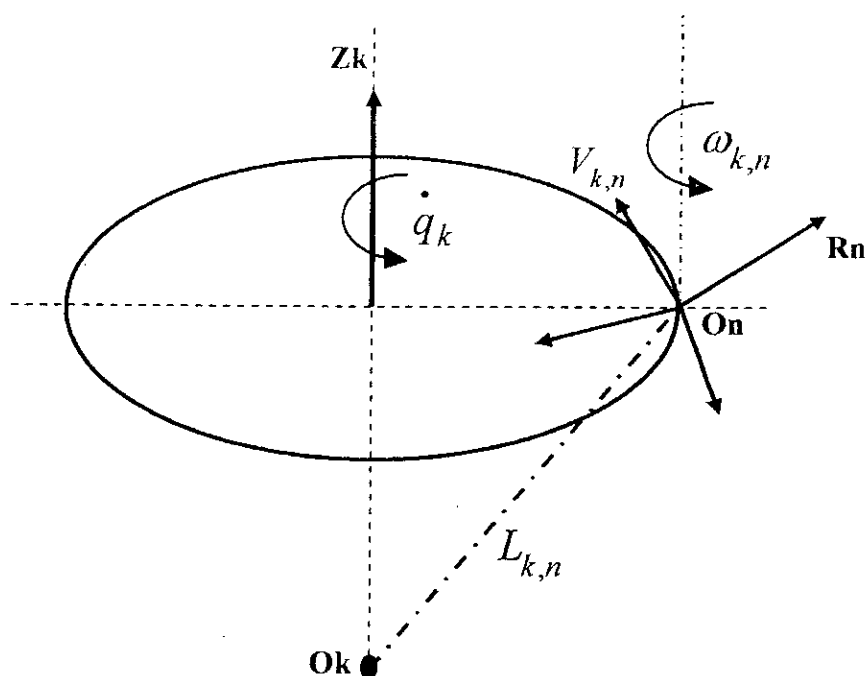


Fig.2.4 Articulation rotoïde

En appliquant le théorème de composition des vitesses, les vitesses de translation et de rotation du repère terminal s'écrivent :

$$\begin{cases} V_n = \sum_{k=1}^n V_{k,n} = \sum_{k=1}^n [\bar{\sigma}_k (a_k \times L_{k,n})] \dot{q}_k \\ \omega_n = \sum_{k=1}^n \omega_{k,n} = \sum_{k=1}^n [\bar{\sigma}_k a_k \times \dot{q}_k] \end{cases} \quad (2.46)$$

En mettant ce système sous forme matricielle et en l'identifiant à la relation :

$$\begin{bmatrix} V_n \\ \omega_n \end{bmatrix} = {}^0 J_n \dot{q} \quad (2.47)$$

On obtient :

$${}^0J_n = \begin{bmatrix} \bar{\sigma}_1(a_1 \times L_{1,n}) & \dots & \bar{\sigma}_n(a_n \times L_{n,n}) \\ \bar{\sigma}_1 a_1 & \dots & \bar{\sigma}_n a_n \end{bmatrix} \quad (2.48)$$

En projetant les éléments de la relation 5 dans un repère R_i , on obtient le jacobien $J_{n,i}$.

En effet, on remarque bien que le produit vectoriel $a_k \times L_{n,k}$ peut se transformer en $\hat{a}_k L_{n,k}$, la $K^{\text{ième}}$ colonne de $J_{n,i}$ noté $J_{n,i}(k)$ et de façon analogue la $K^{\text{ième}}$ colonne de 0J_n devient :

$${}^0J_n(k) = \begin{bmatrix} \bar{\sigma}_k \hat{a}_{0,k} \times (P_{0,n} - P_{0,k}) \\ \bar{\sigma}_k a_{0,k} \end{bmatrix} \quad (2.49)$$

Donc les éléments de la colonne k de la matrice jacobienne s'obtiennent à partir de ceux de la matrice $T_{0,k}$ et du vecteur $P_{0,n}$ que nous avons déjà calculés précédemment.

2.4.2 Modélisation cinématique inverse [1]

L'objectif du modèle cinématique inverse est de calculer, à partir d'une configuration q donnée, les vitesses articulaires \dot{q} qui assurent au repère terminal une vitesse opérationnelle \dot{x} imposée. On peut déterminer la différentielle articulaire dq correspondante à une différentielle des coordonnées opérationnelles dx spécifiée. 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, la plus répandue étant fondée sur la notion de pseudo inverse : les algorithmes traitent de façon unifiée les cas réguliers, singuliers et redondants. Elles nécessitent un temps de calcul relativement important.

On peut écrire le modèle cinématique inverse sous la forme :

$$\begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \\ \dot{q}_4 \\ \dot{q}_5 \\ \dot{q}_6 \end{bmatrix} = J^{-1} \begin{bmatrix} v_x \\ v_y \\ v_z \\ \Omega_x \\ \Omega_y \\ \Omega_z \end{bmatrix} \quad (2.50)$$

Où : J^{-1} est l'inverse du jacobien.

: $\begin{bmatrix} \dot{q}_1, \dot{q}_2, \dot{q}_3, \dot{q}_4, \dot{q}_5, \dot{q}_6 \end{bmatrix}$ Le vecteur des vitesses articulaires.

$\begin{bmatrix} v_x, v_y, v_z, \Omega_x, \Omega_y, \Omega_z \end{bmatrix}$ Le vecteur des vitesses opérationnelles.

2.5 MODELISATION DYNAMIQUE

Les modèles dynamiques des bras manipulateurs sont décrits par un ensemble d'équations mathématiques qui décrivent la dynamique de ceux-ci et peuvent être simulées sur ordinateur dans le but de synthétiser une commande conditionnée par des performances désirées. L'ensemble des équations dynamiques peut être déterminé par des lois mécaniques classiques Newtoniennes et Lagrangiennes. Les approches d'Euler-Lagrange et Newton-Euler permettent d'aboutir aux équations du mouvement des robots.

2.5.1 Approche d'Euler-Lagrange [1] [2] [4]

L'approche d'Euler-Lagrange sert à modéliser et représenter la dynamique des robots à travers les équations du mouvement. Elle s'adapte aux calculs manuels ainsi qu'aux calculs assistés par ordinateur. Le formalisme d'Euler-Lagrange et la transformation homogène de Denavit-Hartenberg amène à un algorithme compact pour représenter les équations dynamiques du mouvement.

L'équation d'Euler-Lagrange est :

$$\frac{d}{dt} \left[\frac{\partial L}{\partial \dot{q}_i} \right] - \frac{\partial L}{\partial q_i} + \frac{\partial E_D}{\partial q_i} = T_i \quad i=1, \dots, n \quad (2.51)$$

Où :L est le Lagrangien qui s'exprime par :

$$L = E_c - E_p$$

E_c : énergie cinétique totale de toutes les liaisons ;

E_p : énergie potentielle totale de toutes les liaisons ;

E_D : énergie de dissipation en cas de présence de frottement visqueux ;

T_i : force généralisée a la $i^{ème}$ articulation ;

n : nombre de degré de liberté ;

q_i : coordonnée généralisée d'ordre i ;

\dot{q}_i : dérivée de la coordonnée généralisée.

2.5.2 Modèle dynamique [6]

Le modèle dynamique d'un robot manipulateur rigide à n degrés de liberté peut être représenté par un système d'équation différentielle non linéaire du second ordre à n entrées formant le vecteur de forces ou couples généralisées Γ , et n sorties qui forment le vecteur position q . Les équations de ce système à n liaisons, décrites dans l'espace des coordonnées articulaires, sont données sous forme matricielle comme suit :

$$A(q)\ddot{q} + B(q) \begin{bmatrix} \dot{q} \\ q \end{bmatrix} + C(q) \begin{bmatrix} \dot{q}^2 \\ q^2 \end{bmatrix} + G(q) + F(\dot{q}) = \Gamma(t) \quad (2.52)$$

Avec $q \in \mathcal{R}^n$, $\dot{q} \in \mathcal{R}^n$, $\ddot{q} \in \mathcal{R}^n$ représente respectivement les positions, les vitesse et les accélérations articulaires et :

$A(q) \in \mathcal{R}^{n \times n}$: Matrice symétrique définie positive des accélérations inertielles dans l'élément A_{ij} de cette matrice est l'inertie de la $i^{ème}$ articulation sur la $j^{ème}$ articulation et vice versa ;

$C(q) \in \mathcal{R}^{n \times n}$: Matrice des couples centrifuges ;

$B(q) \in \mathcal{R}^{n \times \frac{n(n-1)}{2}}$: Matrice des couples de Coriolis ;

$F(\dot{q}) \in \mathfrak{R}^n$: représente les frottements sec et visqueux ;

$\Gamma(t) \in \mathfrak{R}^n$: Vecteur de force ou couples généralisées ;

$\begin{bmatrix} \dot{q} \\ q \end{bmatrix} \in \mathfrak{R}^{\frac{n(n-1)}{2}}$: Vecteur de produit des vitesses généralisées ;

$\begin{bmatrix} \dot{q}^2 \end{bmatrix} \in \mathfrak{R}^n$: Vecteur de carrée des vitesses généralisées ;

Les éléments de A, B, C, G et F sont généralement des fonctions très compliquées et non linéaires par rapport aux coordonnées généralisées du manipulateur. Le précédent modèle du robot est complexe mais vérifie certaines propriétés fondamentales qui peuvent être exploitées pour l'analyse du comportement du système et le calcul de la commande.

Propriété 1 :

La matrice $A(q)$ est symétrique définie positive (SDP), par conséquent tous les éléments diagonaux de cette matrice sont positifs $A(q) > 0$
 $i=1, \dots, n$.

Propriété 2 :

Les matrices A, B, C et les vecteurs G, F sont uniformément bornés.

Propriété 3 :

L'entrée de commande est indépendante pour chaque articulation du manipulateur.

Propriété 4 :

Le vecteur des frottements visqueux et secs $F(\dot{q})$ est caractérisé par les n éléments $\{F_i(\dot{q}_i) \quad i=1, \dots, n\}$, tel que :

$$F_i(\dot{q}_i) = f_{vi} \dot{q}_i + f_{si} \operatorname{sgn}(\dot{q}_i) \quad (2.53)$$

avec f_{vi} et f_{si} sont respectivement les coefficients des frottements visqueux et secs de la $i^{\text{ème}}$ articulation.

Les propriétés 1-4 découlent de la nature physique du robot manipulateur. La propriété 3 est due au fait que les flexibilités des articulations et des structures n'ont pas été prises en compte. Dans ce cas de figure chaque degré de liberté (DDL) est piloté par un actionneur (moteur à courant continu par exemple).

Le modèle dynamique complet du robot Puma 560 est donné dans la référence [6].

2.6 CONCLUSION

La modélisation des robots manipulateurs est basée sur la modélisation de la structure mécanique, les actionneurs et les capteurs. Elle permet la détermination des relations directes et inverses entre les coordonnées généralisées et les coordonnées cartésiennes et leurs dérivées respectives.

La convention de Denavit-Hartenberg fournit une souplesse dans le calcul pour la localisation de la position de l'élément terminal. La position initiale du robot influe sur toute la modélisation. La définition des axes selon Denavit-Hartenberg influe sur le modèle dynamique parce que les moments d'inertie des différents axes changent.

Le passage de l'espace des coordonnées cartésiennes de l'élément terminal à celui des coordonnées généralisées ou articulaires présente des singularités physiques (point en dehors de l'espace de travail), ces singularités apparaissent sous forme de contraintes lors du calcul du modèle géométrique inverse.

Les formalismes de Lagrange et Euler permettent d'établir un système d'équations différentielles reliant les coordonnées généralisées aux forces et/ou couples généralisés. Ce formalisme présente plusieurs approches dont le résultat est le même, mais pour des applications différentes il est parfois nécessaire d'avoir des séparations au niveau des termes d'inerties, centrifuges, Coriolis et de gravité.

Les modèles obtenus mettent en évidence la difficulté de commander ce genre de bras manipulateurs, étant donné les fortes non linéarités qui régissent le fonctionnement dynamique de ceux-ci.

Génération De Trajectoire

3.1 Introduction [3]

La génération de trajectoires est une étape très importante dans la commande des robots manipulateurs, elle consiste à calculer les consignes de référence en position, en vitesse, et en accélération qui décrivent le mouvement désiré du robot.

Cependant la trajectoire est l'évolution de la position, et ses dérivées temporelles en fonction du temps pour chacune des articulations.

Le mouvement le plus simple est d'aller d'un point initial à un point final, ainsi le robot est commandé de changer sa configuration initial vers une configuration final.

Ce type de mouvement convient aux taches de transfert d'objets quand l'espace de travail ne comporte aucun obstacle. Pour certaines raisons telles qu'éviter les obstacles, le chemin à suivre par l'élément terminal peut être contraint par l'addition de points intermédiaires aux configurations initial et final.

Pour assurer le fonctionnement normal du mécanisme on choisira des mouvements continus, pour cela on définit une fonction lisse (dérivées première et secondaire continues) afin d'éviter les risques d'usure et de vibrations pouvant exciter les modes propres du manipulateur.

En résumé, le parcours peut être planifié de différentes manières dont on distingue :

- le mouvement entre deux points avec trajectoire libre entre les points;
- le mouvement entre deux points via des points intermédiaires avec trajectoire libre entre les points intermédiaires;
- le mouvement entre deux points avec trajectoire contrainte entre les points (trajectoire rectiligne);
- le mouvement entre deux points via des points intermédiaires avec trajectoire contrainte entre les points intermédiaires ;

Dans les deux premiers cas la génération de la trajectoire peut se faire directement dans l'espace articulaire. Quant aux deux derniers, la trajectoire étant décrite dans l'espace opérationnelle, il est préférable de raisonner dans cet espace.

3.2 Génération de la trajectoire dans l'espace articulaire [3]

La génération de la trajectoire dans l'espace articulaire donne en résultat un ensemble de données : position, vitesse et accélération articulaires $(\theta_d, \dot{\theta}_d, \ddot{\theta}_d)$ qui sont utilisées comme un signal de référence. Pour une position initiale et finale données dans l'espace de travail on utilise la géométrie inverse pour déterminer les angles articulaires correspondant à cette position et même pour l'orientation, la position initiale du manipulateur devient un ensemble d'angles articulaires de départ et la position finale correspond à un autre ensemble d'angles articulaires d'arrivée, ce qui est demandé pour la planification de la trajectoire est de trouver une fonction lisse par chaque articulation dont la valeur à l'instant t_0 est la position initiale de l'articulation et dont la valeur à l'instant t_f est la position désirée de cette même articulation. Il y a beaucoup de fonctions lisses $\theta(t)$ qui pourraient être utilisées pour interpoler les valeurs d'angles articulaires. On cite :

3.2.1 La fonction polynomiale cubique [1] [2]

Il s'agit d'une interpolation simple en effectuant un simple mouvement continu, au moins quatre contraintes sur $\theta(t)$ sont évidentes pour avoir une fonction polynomiale cubique de la forme :

$$\theta(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3 \quad (3.1)$$

Ainsi on obtient la vitesse articulaire :

$$\dot{\theta}(t) = a_1 + 2a_2 t + 3a_3 t^2 \quad (3.2)$$

Et aussi l'accélération articulaire :

$$\ddot{\theta}(t) = 2a_2 + 6a_3 t \quad (3.3)$$

Sachant qu'un polynôme du 3ème degré, admet quatre coefficients, il peut être donc construit à partir de quatre contraintes, deux sont obtenues à partir du choix des valeurs initiales et finales de la position :

$\theta(0) = \theta_0$, $\theta_d(t_f) = \theta_f$ quant aux deux autres contraintes, elles proviennent du fait que

l'articulation démarre et arrive avec une vitesse nulle : $\dot{\theta}(0) = 0$, $\dot{\theta}(t_f) = 0$ Et en combinant les

deux fonctions : $\theta(t)$ et $\dot{\theta}(t)$ avec les quatre contraintes, on obtient quatre équations à quatre inconnues :

$$\theta_0 = a_0 \quad (3.4)$$

$$\theta_f = a_0 + a_1 t_f + a_2 t_f^2 + a_3 t_f^3 \quad (3.5)$$

$$0 = a_1 \quad (3.6)$$

$$0 = a_1 + 2a_2 t_f + 6a_3 t_f^2 \quad (3.7)$$

En résolvant ce système d'équations on obtient :

$$a_0 = \theta_0 \quad (3.8)$$

$$a_1 = 0 \quad (3.9)$$

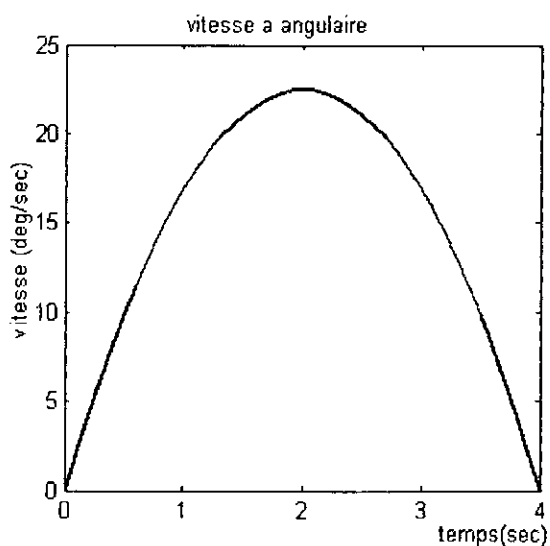
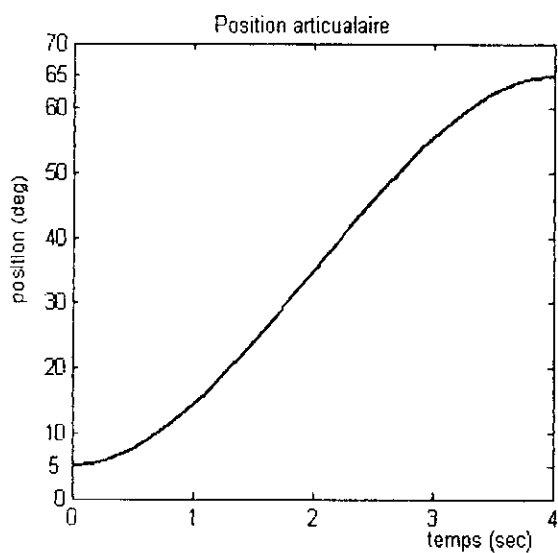
$$a_2 = \frac{3}{t_f^2} (\theta_f - \theta_0) \quad (3.10)$$

$$a_3 = -\frac{2}{t_f^3} (\theta_f - \theta_0) \quad (3.11)$$

Avec ces quatre coefficients, on peut calculer le polynôme cubique qui connecte n'importe quelle position désirée.

Exemple de génération d'une trajectoire articulaire :

On veut générer la trajectoire articulaire pour passer d'un angle initial de 5° à un angle final de 65° dans 4 secondes.



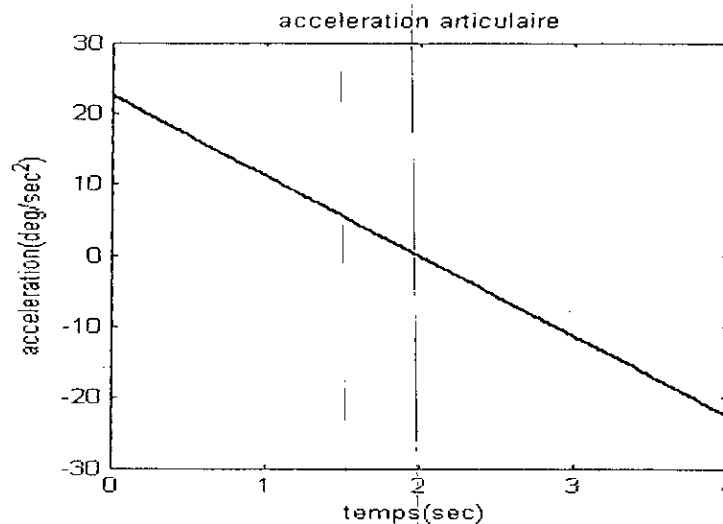


Fig. 3.1 Génération d'une trajectoire polynomiale

3.2.2 Polynôme de degré cinq [2] [3]

On peut avoir des polynômes d'ordre supérieur pour la planification de la trajectoire, ils sont quelque fois utilisés dans le cas où on voudrait indiquer la position, la vitesse, et l'accélération au début et à la fin de la trajectoire et aussi sont utilisés quand les robots manipulateurs fonctionnent à une grande vitesse, il est nécessaire d'assurer la continuité des accélérations afin d'éviter l'excitation du mécanisme.

Il faut avoir un polynôme d'ordre cinq de la forme :

$$\theta(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3 + a_4 t^4 + a_5 t^5 \quad (3.12)$$

Pour déterminer les coefficients de ce polynôme il faut avoir au moins six contraintes, deux sur la position et deux sur la vitesse et deux autres sur l'accélération :

$$\theta_0 = a_0 \quad (3.13)$$

$$\dot{\theta}_0 = a_1 \quad (3.14)$$

$$\ddot{\theta}_0 = 2a_2 \quad (3.15)$$

$$\theta_f = a_0 + a_1 t_f + a_2 t_f^2 + a_3 t_f^3 + a_4 t_f^4 + a_5 t_f^5 \quad (3.16)$$

$$\dot{\theta}_f = a_1 + 2a_2 t_f + 3a_3 t_f^2 + 4a_4 t_f^3 + 5a_5 t_f^4 \quad (3.17)$$

$$\ddot{\theta}_f = 2a_2 + 6a_3 t_f + 12a_4 t_f^2 + 20a_5 t_f^3 \quad (3.18)$$

En résolvant le système d'équations on trouve :

$$a_0 = \theta_0 \quad (3.19)$$

$$a_1 = \dot{\theta}_0 \quad (3.20)$$

$$a_2 = \ddot{\theta}_0 \quad (3.21)$$

$$a_3 = \frac{20\theta_f - 20\theta_0 - (8\dot{\theta}_f + 12\dot{\theta}_0)t_f + (3\ddot{\theta}_0 - \ddot{\theta}_f)t_f^2}{2t_f^3} \quad (3.22)$$

$$a_4 = \frac{30\theta_0 - 30\theta_f + (14\dot{\theta}_f + 16\dot{\theta}_0)t_f - (3\ddot{\theta}_0 - 2\ddot{\theta}_f)t_f^2}{2t_f^4} \quad (3.23)$$

$$a_5 = \frac{12\theta_f - 12\theta_0 - (6\dot{\theta}_f + 6\dot{\theta}_0)t_f - (\ddot{\theta}_0 - \ddot{\theta}_f)t_f^2}{2t_f^5} \quad (3.24)$$

3.2.3 Loi Bang-Bang avec palier de vitesse (loi trapèze) [2]

Une autre voie pour générer les consignes est ce qu'on appelle segment linéaire avec portions paraboliques ou bien loi Bang-Bang avec palier de vitesse. Ce type de trajectoire est approprié quand on désire avoir une vitesse constante le long d'une portion du parcours. Cette trajectoire est telle que la vitesse est rampée en haut à sa valeur spécifique initialement et puis rampée en bas au but (position désirée). Pour accomplir tout cela on spécifie la position désirée en trois phases. La première phase de l'instant t_0 à l'instant t_b est polynomiale quadratique.

À l'instant t_b la position change de forme et devient linéaire. Cela correspond à une vitesse constante. Finalement à l'instant $t_f - t_b$ la trajectoire de position redevient polynomiale quadratique de sorte que la vitesse soit linéaire.

On choisit l'instant t_b de façon à ce que la courbure de position soit symétrique comme il est montré à la figure **. par convention on suppose que $t_0 = 0$ et $\dot{q}(t_f) = \dot{q}(0) = 0$. Puis entre les instants 0 et t_b on a :

$$q^d(t) = a_0 + a_1 t + a_2 t^2 \quad (3.25)$$

De façon que la vitesse soit :

$$\dot{q}^d(t) = a_1 + 2a_2 t \quad (3.26)$$

Les contraintes $q(0) = 0$ et $\dot{q}(0) = 0$ impliquent que :

$$a_0 = q_0 \quad (3.27)$$

$$a_1 = 0 \quad (3.28)$$

Puis que à l'instant t_b on veut que la vitesse soit égale à une constante donnée on aura :

$$\dot{q}_d(t_b) = 2a_2 t_b = V \quad (3.29)$$

Ce qui implique que :

$$a_2 = \frac{V}{2t_b} \quad (3.30)$$

Par conséquent, la trajectoire requise entre les instants 0 et t_b est donnée par :

$$q_d(t) = q_0 + \frac{V}{2t_b} t^2 = q_0 + \frac{a}{2} t^2 \quad (3.31)$$

$$\dot{q}_d(t) = q_0 + \frac{V}{t_b} t = at \quad (3.32)$$

$$\ddot{q}_d = \frac{V}{t_b} = a \quad (3.33)$$

Où a est l'accélération.

Maintenant, entre les instants t_f et $t_f - t_b$, la trajectoire est un segment linéaire qui correspond à une vitesse constante V .

$$q_d(t) = \alpha_0 + \alpha_1 t = \alpha_0 + Vt \quad (3.34)$$

Par symétrie :

$$q_d(t) = \frac{q_0 + q_1}{2} \quad (3.34)$$

On aura

$$\frac{q_0 + q_1}{2} = \alpha_0 + V \frac{t_f}{2} \quad (3.35)$$

Ce qui implique que :

$$\alpha_0 = \frac{q_0 + q_1 - Vt_f}{2} \quad (3.36)$$

Comme les deux segments coïncident à l'instant t_b on requiert

$$q_0 + \frac{V}{2} t_b = \frac{q_0 + q_1 - Vt_f}{2} + Vt_b \quad (3.37)$$

Ce qui donne l'expression :

$$t_b = \frac{q_0 - q_1 + Vt_2}{V} \tag{3.38}$$

Il à noté qu'on a la contrainte :

$$0 < t_b \leq \frac{t_f}{2} \tag{3.39}$$

Ce-ci mène à l'inéquation :

$$\frac{q_f - q_0}{V} < t_f \leq \frac{2(q_f - q_0)}{V} \tag{3.40}$$

Autrement dit

$$\frac{q_f - q_0}{t_f} < V \leq \frac{2(q_f - q_0)}{t_f} \tag{3.41}$$

Par conséquent, la vitesse spécifiée doit être entre ces deux limites ou bien le mouvement ne sera pas possible.

La portion de la trajectoire entre $t_f - t_b$ et t_f est maintenant déterminé par des considérations de symétrie. La trajectoire complète est donnée par :

$$q_d(t) = \begin{cases} q_0 + \frac{a}{2}t^2 \dots\dots\dots 0 \leq t \leq t_b \\ \frac{q_f + q_0 - Vt_f}{2} + Vt \dots\dots\dots t_b < t \leq t_f - t_b \\ q_f - \frac{a}{2}t_f^2 + at_f t - \frac{a}{2}t^2 \dots\dots\dots t_f - t_b < t \leq t_f \end{cases} \tag{3.42}$$

La figure montre une telle trajectoire où la vitesse maximale est $V = 60$.dans ce cas $t_b = \frac{1}{3}$.les courbes de vitesse et d'accélération sont données dans les figures respectivement.

3.3 Génération de la trajectoire dans l'espace cartésien :

Les trajectoires dans l'espace articulaire sont plus faciles à calculer et elles sont plus simples, ce pendant elles deviennent complexes si elles sont décrites dans l'espace cartésien, puisque dans le premier cas on ne fait aucune correspondance continue entre l'espace articulaire et l'espace cartésien il n'y a aucun problème avec les singularités du mécanisme.

La trajectoire dans l'espace cartésien donne comme résultat la position, la vitesse et l'accélération cartésiennes (x, \dot{x}, \ddot{x}) , pour certaines structures de commande spécifiques ces données doivent être converties en grandeurs équivalentes dans l'espace articulaire, une analyse complète devrait utiliser la géométrie inverse, l'inverse du jacobien et la dérivée du jacobien pour pouvoir calculer les angles, les vitesses et les l'accélérations articulaires $(\theta_d, \dot{\theta}_d, \ddot{\theta}_d)$ qui vont être utilisées comme un signal de référence pour les régulateurs synthétisés dans l'espace articulaire.

Dans l'espace de travail, la forme du parcours prise par l'élément terminal n'est pas simple, mais plutôt une forme compliquée qui dépend de la géométrie du manipulateur, les formes du parcours sont décrites en terme des fonctions du temps qui déterminent la position et l'orientation cartésiennes, la forme spatiale du parcours entre ses points peut être : une ligne droite, circulaire, sinusoidale, ellicoïdale ou d'autres formes.

Pour la planification de trajectoire basée sur l'espace cartésien, les fonctions qui forment une trajectoire sont des fonctions du temps qui représentent les variables cartésiennes. Ces parcours peuvent être planifiés directement à partir de la définition, par l'utilisateur, des points de parcours qui sont les spécifications sur le parcours désiré.

3.4 Problèmes géométriques dans un parcours cartésien [1]

Plusieurs problèmes sont posés dans la génération des trajectoires cartésiennes à cause de la correspondance continue qui se trouve entre la forme de la trajectoire décrite dans l'espace cartésien et celle décrite dans l'espace articulaire. On en cite trois problèmes majeurs :

Problème du type 1 :

C'est le problème des points intermédiaires inaccessibles, comme il est présenté dans la figure1, si on veut que le robot démarre du point A pour arriver au point B en passant par une trajectoire désirée qui est une ligne droite par exemple, malgré que les points initial et final du

parcours sont compris dans l'espace de travail du robot, il y a certains points qui appartiennent à cette trajectoire et qui n'appartiennent pas à l'espace de travail du manipulateur. Ces points deviennent inaccessibles, ce qui rend impossible d'effectuer cette tâche dans l'espace opérationnel. Néanmoins il n'y aurait pas ce type de problème dans l'espace articulaire.

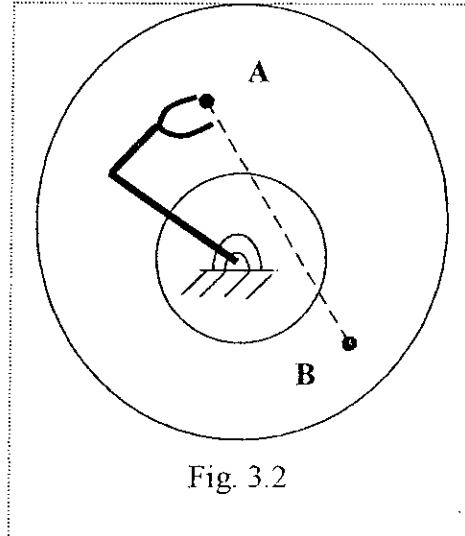


Fig. 3.2

Problème de type 2 :

C'est le problème de la vitesse articulaire qui est trop élevée en passant près de la singularité. Certains parcours dans l'espace cartésien sont impossible à exécuter par le robot manipulateur surtout dans le cas où il doit suivre une trajectoire en s'approchant d'une configuration singulière du mécanisme, (par exemple une ligne). Une ou plusieurs vitesses articulaires peuvent augmenter vers l'infini. Puisque les vitesses des articulations sont bornées supérieurement, cette situation a pour conséquence la déviation du robot manipulateur de son parcours désiré. La figure 2 montre un robot à deux articulations ayant la même longueur se déplaçant le long du parcours du point A au point B, la trajectoire désirée correspond au mouvement de l'élément terminal du manipulateur à vitesse linéaire constante le long d'un parcours rectiligne. Sur la figure on montre plusieurs positions intermédiaires du manipulateur qui ont été dessinées pour mieux voir le mouvement, et tous les points de la trajectoire sont accessibles, mais dès que le robot passe par la position centrale du parcours, la vitesse de l'articulation 01 devient très élevée. Plus le parcours s'approche étroitement de l'axe articulaire 01, plus la vitesse sera grande.

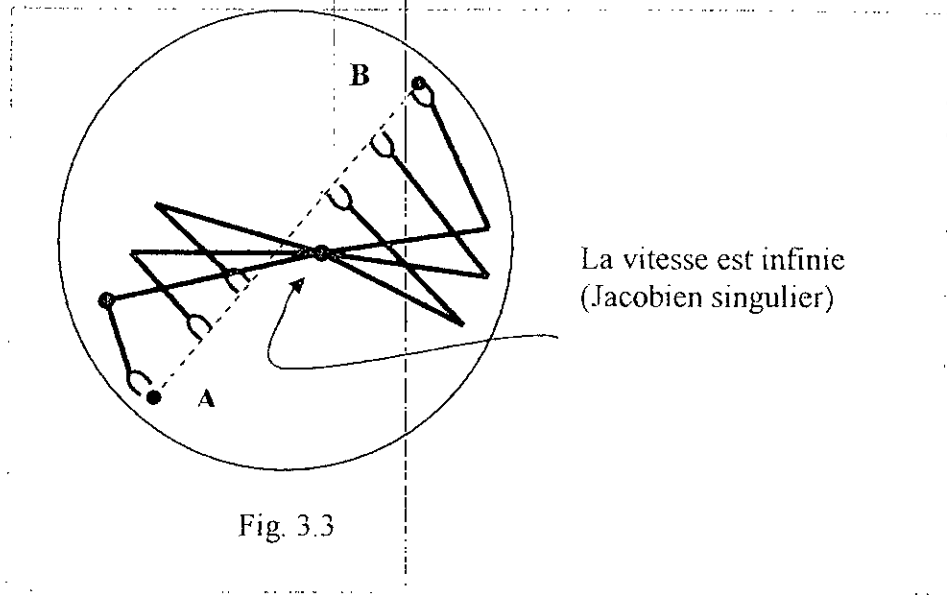


Fig. 3.3

Problème de type 3 :

Le troisième problème se produit quand il y aura plusieurs solutions pour atteindre un point donné dans la trajectoire, la figure 3 montre un robot manipulateur à deux liaisons de même longueur ayant des butées au niveau des articulations qui restreignent le nombre de solutions avec lesquelles il peut atteindre un point donné dans l'espace. En particulier, un problème surviendrait si le point d'arrivée ne peut pas être atteint dans la même solution physique quand le robot est au point de départ. Comme il se voit dans la figure 3 le manipulateur peut atteindre tous les points du parcours pour une certaine solution, mais pas pour n'importe quelle solution. Dans cette situation le système ne peut pas se déplacer à causes des butées mécaniques.

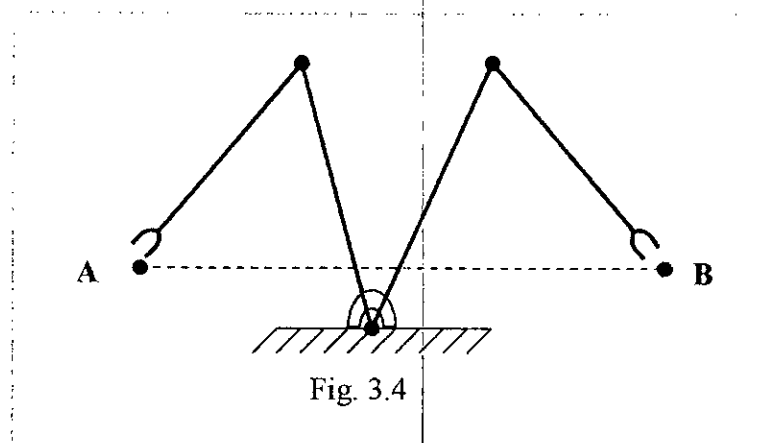


Fig. 3.4

3.5 Étude comparative et discussion du choix d'une trajectoire [1] [3]

Les deux approches étudiées précédemment présentent plusieurs avantages et inconvénients. La génération du mouvement dans l'espace articulaire présente les avantages suivants :

- elle nécessite moins de calcul en ligne, puisqu'il n'y a pas d'appel au modèle géométrique ou cinématique inverse ;
- le mouvement n'est pas affecté par le passage par les configurations singulières ;
- les contraintes de vitesses et de couples maximaux sont directement déduites des limites physiques des actionneurs.

En contre partie, la géométrie de la trajectoire de l'organe terminal dans l'espace opérationnel est imprévisible bien qu'elle soit répétitive : il y a donc risque de collision lorsque le robot évolue dans un environnement encombré. Ce type de mouvement est par conséquent approprié pour réaliser des déplacements rapides dans un espace libre.

La génération de mouvement dans l'espace opérationnel permet de contrôler la géométrie de la trajectoire. En revanche :

- elle exige la transformation en coordonnées articulaires de chaque point de la trajectoire ;
- elle peut être mise en échec lorsque la trajectoire calculée passe par une position singulière ;
- elle est mise en échec chaque fois que les points de la trajectoire engendrée ne sont pas dans l'espace accessible par le robot ou chaque fois que la trajectoire impose une reconfiguration du mécanisme (changement d'aspect en cours de trajectoire);
- les limites en vitesse et en couple dans l'espace opérationnel varient selon la configuration du robot. On exprime alors ces limites par des valeurs traduisant des performances moyennes, satisfaites quelle que soit la configuration du robot. On impose donc au robot de travailler en deçà de ses capacités réelles.

Le choix d'une méthode de génération de mouvement dépend de l'application considérée. Chaque approche a ses propres limites, inhérentes au fait que les contraintes sont exprimées soit dans l'espace articulaire (vitesses, couples, buttés) soit dans l'espace opérationnel (précision, obstacles).

3.6 Génération de trajectoire dans l'espace cartésien :

3.6.1 Description d'une trajectoire planeaire :

On veut générer une trajectoire dans l'espace cartésien, pour imposer une position et une orientation au manipulateur à suivre, on veut que l'élément terminale du bras manipulateur suive un demi cercle qui se situe dans le plan $O'XZ$, le centre du cercle est repéré par les coordonnées suivantes (x',z') et dont le rayon est R .

On impose une orientation à l'outil terminal de façon à ce qu'il soit radial au sens rentrant et en arrivant à l'extrémité de l'arc il conserve son orientation perpendiculairement au tronçon rectiligne.

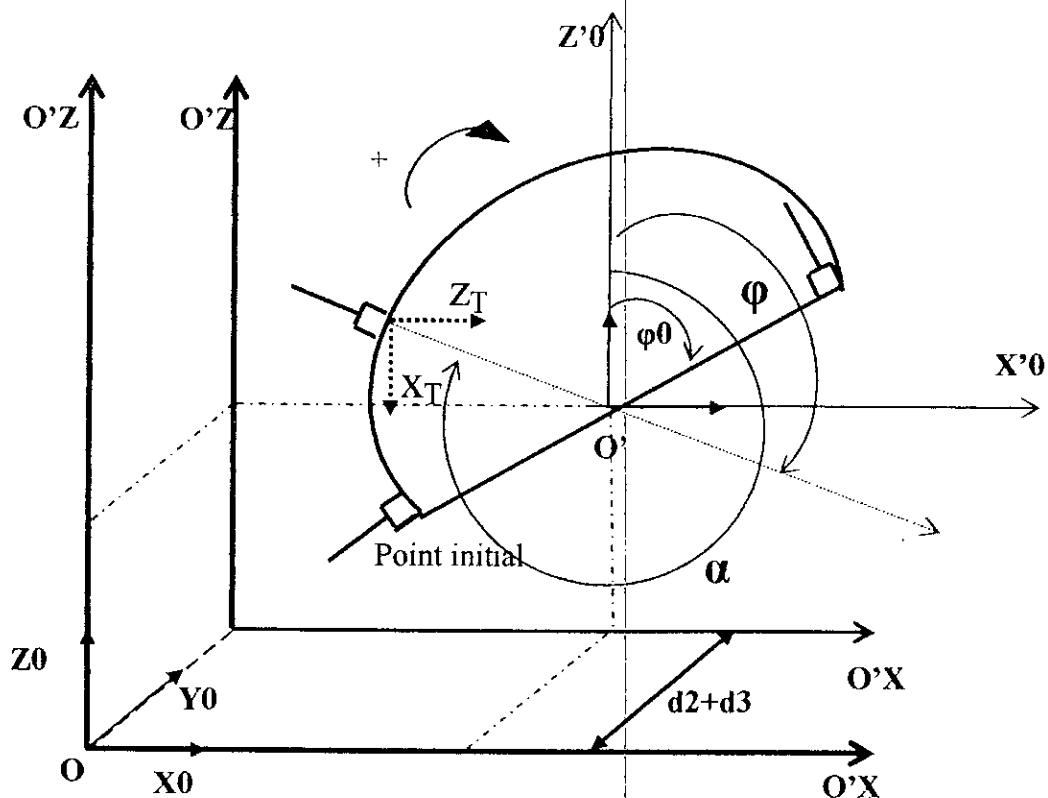


Fig. 3.5 Schéma représentatif d'une trajectoire dans le plan

On désire que l'outil parcoure le demi cercle dans le sens de la flèche. L'orientation de l'outil terminal est représentée par l'angle $\varphi(t)$ pour lequel le point concerné sur le demi cercle est repéré par l'angle $\alpha(t)$.

Ainsi nous obtenons les équations suivantes :

$$\begin{cases} x_T = ox' + R \sin(\alpha) \\ z_T = oz' + R \cos(\alpha) \\ y_T = d_2 + d_3 = oy' \end{cases} \quad (3.43)$$

On impose à l'angle φ une variation linéaire en fonction de α .

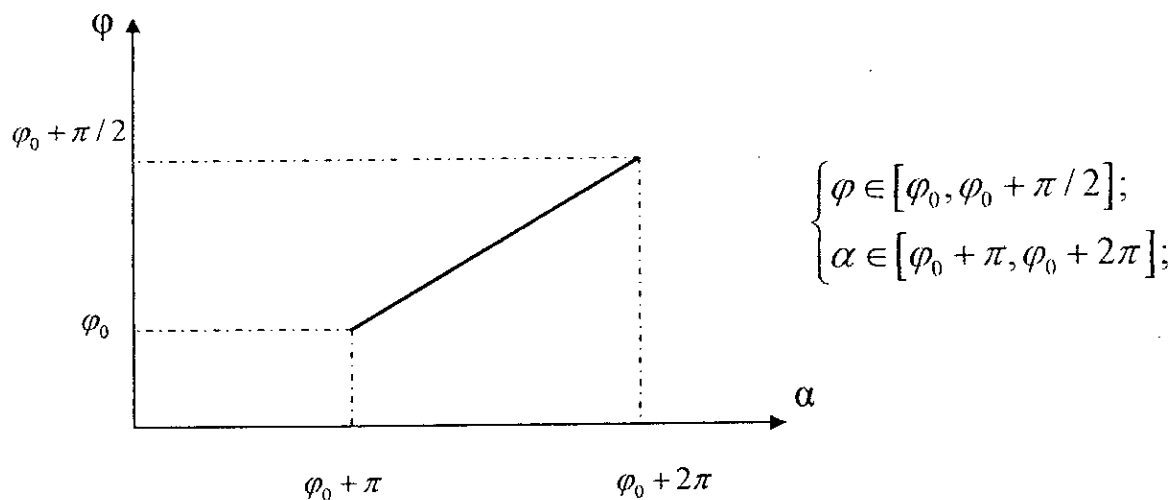


Fig. 3.6 variation de l'angle φ en fonction du α

D'où $\varphi(t)$ est donné par :

$$\varphi(t) = \frac{1}{2} \alpha(t) + \frac{1}{2} \varphi_0 - \frac{\pi}{2}; \quad (3.44)$$

3.6.1.1 Génération de la trajectoire sur le 1^{er} tronçon :

L'évolution de l'angle $\alpha(t)$ est imposée telle qu'il suive une loi Bang-Bang avec palier de vitesse.

En résumé :

Position :

$$\begin{cases} x_T = ox' + R \sin(\alpha) \\ z_T = oz' + R \cos(\alpha) \\ y_T = d_2 + d_3 = oy' \\ \alpha = \alpha(t) \\ \varphi(t) = \frac{\alpha(t)}{2} + \frac{\varphi_0}{2} - \frac{\pi}{2} \end{cases} \quad (3.45)$$

Vitesse :

$$\begin{cases} \dot{x}_T = R \cos(\alpha) \dot{\alpha}; \\ \dot{z}_T = -R \sin(\alpha) \dot{\alpha}; \\ \dot{y}_T = 0; \\ \dot{\varphi}(t) = \dot{\alpha}(t) / 2; \end{cases} \quad (3.46)$$

Accélération :

$$\begin{cases} \ddot{x}_T = R \sin(\alpha) \dot{\alpha}^2 - R \cos(\alpha) \ddot{\alpha}; \\ \ddot{z}_T = R \cos(\alpha) \dot{\alpha}^2 + R \sin(\alpha) \ddot{\alpha}; \\ \ddot{y}_T = 0; \\ \ddot{\varphi}(t) = \ddot{\alpha}(t) / 2; \end{cases} \quad (3.47)$$

3.6.1.2 Génération de la trajectoire sur le 2^{ème} tronçon :

L'outil se déplace le long du diamètre AO'B tel qu'il reste perpendiculaire à AB. Donc l'orientation de l'outil est maintenue constante de sorte que l'angle φ vérifie toujours

l'équation : $\varphi = \varphi_0 + \frac{\pi}{2}$

Cependant le point figuratif est sur l'axe O'Z'';

Les équations horaires donnant les coordonnées du wrist sont les suivantes :

La position :

$$\begin{cases} x_T = o'_x + z'' \sin(\varphi_0); \\ y_T = d_2 + d_3; \\ z_T = o'_z + z'' \cos(\varphi_0); \\ \varphi = \varphi_0 + \frac{\pi}{2}; \end{cases} \quad (3.48)$$

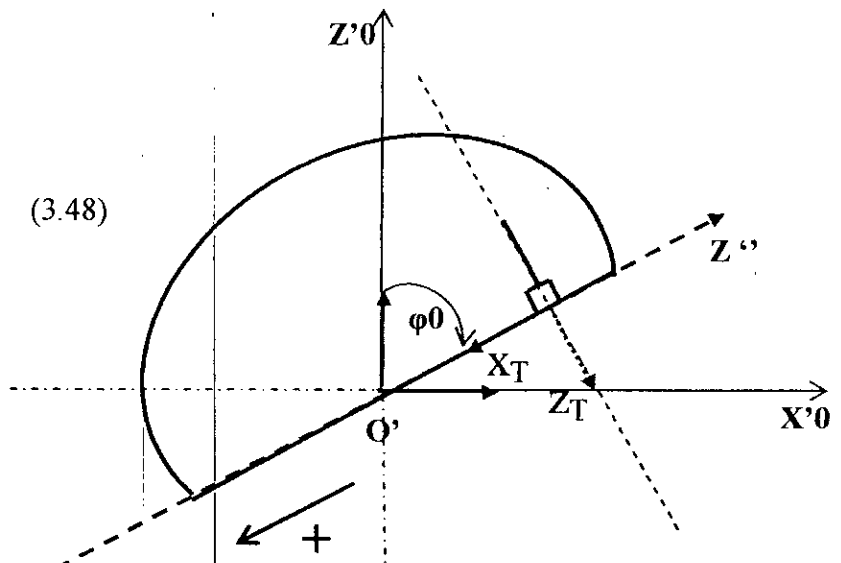


Fig. 3.7 Génération du deuxième tronçon

La vitesse :

$$\begin{cases} v_{Tx} = v_z \cdot \sin(\varphi_0); \\ v_{Ty} = 0; \\ v_{Tz} = v_z \cdot \cos(\varphi_0); \\ \dot{\varphi} = 0; \end{cases} \quad (3.49)$$

L'accélération :

$$\begin{cases} a_{Tx} = a_z \cdot \sin(\varphi_0); \\ a_{Ty} = 0; \\ a_{Tz} = a_z \cdot \cos(\varphi_0); \\ \ddot{\varphi} = 0; \end{cases} \quad (3.50)$$

3.6.2 Description d'une trajectoire spatial :

On veut que l'outil terminal suive la trajectoire décrite ci-dessus (glissement sur la surface d'un huitième de sphère de rayon R et de centre $O'(x_o', y_o', z_o')$ donné). L'orientation est calculée pour assurer la direction radiale entrante à l'outil afin qu'il puisse réaliser la tâche demandée (polissage par exemple) sachant que nous avons adopté la représentation d'Euler concernant les orientations.

Les angles φ et ψ sont des paramètres auxiliaires qui servent à déterminer la position et l'orientation de l'outil terminal dans le repère lié à la sphère. Et par une simple translation de vecteur $[x_o', y_o', z_o']^T$ on retrouvera les coordonnées cartésiennes de l'outil dans le repère de la base.

En fonction des paramètres φ et ψ on peut exprimer le vecteur des positions et orientations comme suit :

$$\begin{cases} x = R \sin \varphi \cos \psi + x_o' \\ y = R \sin \varphi \sin \psi + y_o' \\ z = R \cos \varphi + z_o' \\ \alpha = \psi \\ \beta = \varphi - \pi \\ \gamma = \psi \end{cases} \quad (3.51)$$

$$\text{Avec : } \begin{cases} \varphi \in [-\frac{\pi}{2}, 0] \\ \psi \in [0, \frac{\pi}{2}] \end{cases} \quad (3.52)$$

Où : x, y et z représentent la position de l'outil et α, β , et γ sont les angles d'Euler correspondantes à l'orientation de l'outil terminal exprimées dans le repère de base.

Les angles φ et ψ obéissent à une loi d'évolution dans le temps du type Bang-Bang avec palier de vitesse comme il est schématisé ci-dessous.

Les vitesses :

$$\begin{cases} \dot{x} = R(\cos \varphi \dot{\varphi} \cos \psi - \sin \varphi \sin \psi \dot{\psi}) \\ \dot{y} = R(\cos \varphi \dot{\varphi} \sin \psi + \sin \varphi \cos \psi \dot{\psi}) \\ \dot{z} = -R \sin \varphi \dot{\varphi} \\ \dot{\alpha} = \dot{\psi} \\ \dot{\beta} = \dot{\varphi} \\ \dot{\gamma} = \dot{\psi} \end{cases} \quad (3.53)$$

Les accélérations :

$$\begin{cases} \ddot{x} = R(-\sin \varphi \dot{\varphi}^2 \cos \psi + \cos \varphi \ddot{\varphi} \cos \psi + -\cos \varphi \dot{\varphi} \sin \psi \dot{\psi} - \cos \varphi \dot{\varphi} \sin \psi \dot{\psi} - \sin \varphi \cos \psi \dot{\psi}^2 - \sin \varphi \sin \psi \ddot{\psi}) \\ \ddot{y} = R(-\sin \varphi \dot{\varphi}^2 \sin \psi + \cos \varphi \ddot{\varphi} \sin \psi + \cos \varphi \dot{\varphi} \cos \psi \dot{\psi} + \cos \varphi \dot{\varphi} \sin \psi \dot{\psi} + \sin \varphi \cos \psi \dot{\psi}^2 + \sin \varphi \sin \psi \ddot{\psi}) \\ \ddot{z} = -R(\cos \varphi \dot{\varphi}^2 + \sin \varphi \ddot{\varphi}) \\ \ddot{\alpha} = \ddot{\psi} \\ \ddot{\beta} = \ddot{\varphi} \\ \ddot{\gamma} = \ddot{\psi} \end{cases}$$

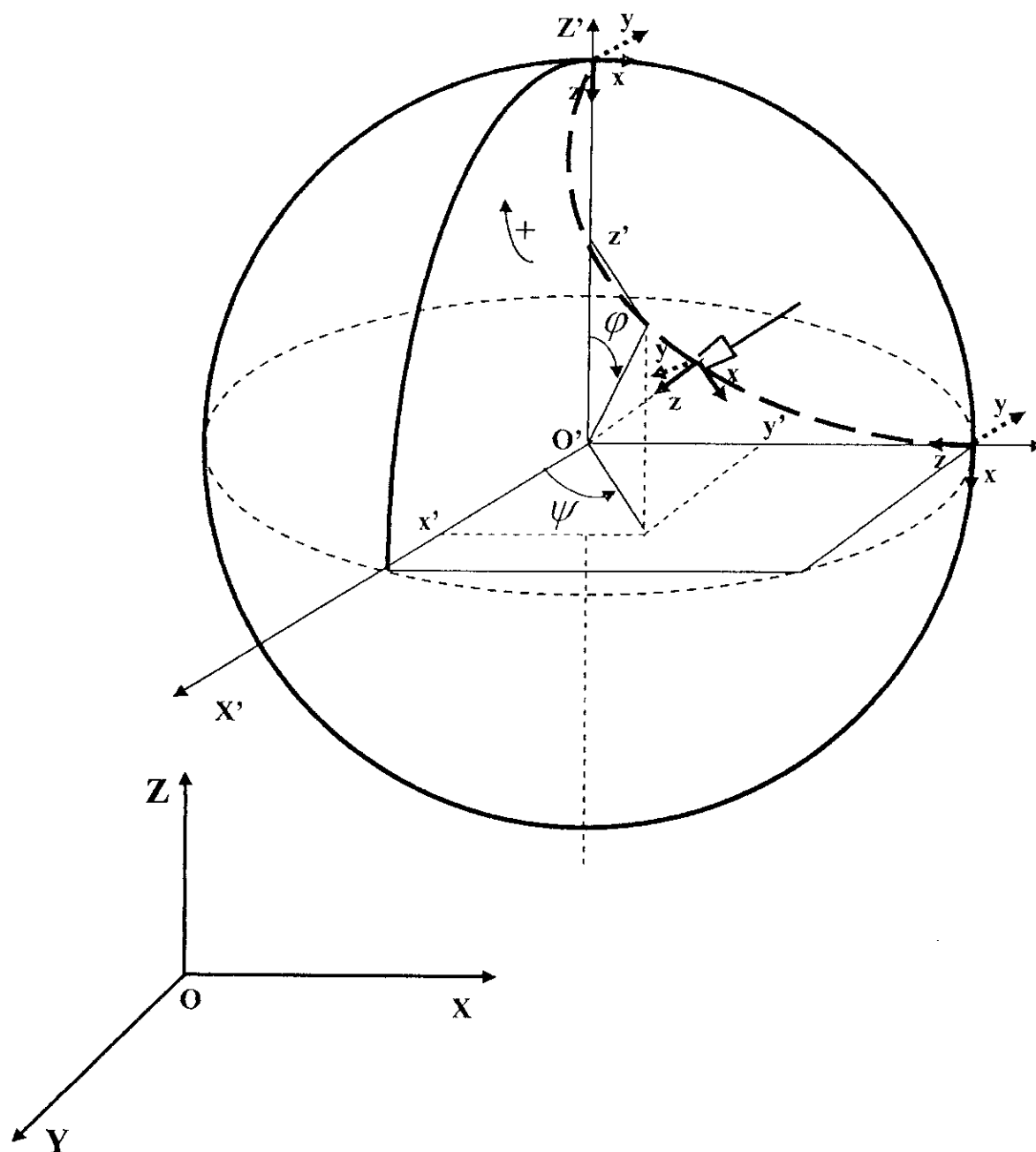


Fig.3.8 Génération d'une trajectoire tridimensionnelle

Conclusion :

Ce chapitre a permis de présenter plusieurs méthodes de génération de trajectoire dans l'espace opérationnel, en vue de réaliser des tâches réelles, couramment utilisées en robotique. Différents types d'interpolation ont été détaillés, notamment la loi Bang-Bang avec palier de vitesse qui est implantée sur la plus part des contrôleurs industriels.

Nous avons vu que le choix de l'une des méthodes de génération dépendait de la tâche à réaliser et de la façon dont sont exprimées les contraintes.

Synthèse De La Commande Par Couple Calculé

4.1 INTRODUCTION [3]

La commande par découplage non linéaire est considérée comme la solution idéale pour la commande des robots manipulateurs. Lorsque l'application exige des évolutions rapides des robots et une grande précision dynamique, il est nécessaire de concevoir un système de commande plus sophistiqué qui prenne en compte tout ou partie des forces d'interaction dynamiques. L'utilisation de la commande par découplage non linéaire constitue une bonne approche en ce sens. Ce type de commande est aussi connu sous le nom de *commande dynamique* (ou "couple calculé", *computed torque* dans la littérature anglo-saxonne) parce qu'il est fondé sur l'utilisation du modèle dynamique.

Théoriquement, il assure le découplage et la linéarisation des équations du modèle, ayant pour effet une réponse uniforme quelle que soit la configuration du robot.

La mise en œuvre de cette méthode exige le calcul du modèle dynamique en ligne et la connaissance des valeurs numériques des paramètres inertiels et de frottements, ce qui ne constitue plus maintenant une limite rédhibitoire. Le problème du calcul en ligne est en effet résolu pratiquement grâce aux méthodes de modélisation et grâce aux évolutions technologiques en micro-informatique. Le développement des techniques d'identification permet une bonne évaluation des paramètres dynamiques.

La commande par découplage non linéaire consiste à transformer par retour d'état le problème de commande d'un système non linéaire en un problème de commande d'un système linéaire. Dans le cas général, le problème de linéarisation par retour d'état d'un système non linéaire n'est pas facile à résoudre. Cependant, dans le cas des robots manipulateurs rigides, l'élaboration d'une loi de commande qui linéarise et découple les équations est simplifiée par le fait que le nombre d'actionneurs est égal au nombre de variables articulaires et que le modèle dont on dispose est un modèle inverse qui exprime l'entrée Γ du système en fonction du vecteur d'état (q, \dot{q}) et de \ddot{q} . Ces propriétés font que les équations du robot définissent un *système plat* dont les sorties plates sont les variables articulaires q .

Etant donné que la loi de commande utilise seulement les variables d'état q et \dot{q} , on qualifie cette loi, qui est équivalente à une commande plate, de commande par découplage statique. Dans ce qui suit, on développe cette méthode, tout d'abord dans l'espace articulaire, puis dans l'espace opérationnel.

Lorsque la tâche est définie dans l'espace opérationnel, deux approches (structures) sont proposées pour synthétiser la loi de commande du manipulateur en utilisant la technique du découplage non linéaire :

- On transforme la tâche définie dans l'espace opérationnel en un mouvement dans l'espace articulaire, puis on met en œuvre la loi de commande dans l'espace articulaire. Le signal d'erreur est exprimé dans l'espace articulaire.
- On calcule commande à base d'erreur exprimée dans l'espace opérationnel sans passer aux variables articulaires.

4.2 COMMANDE DANS L'ESPACE OPERATIONNEL AVEC CORRECTION DANS L'ESPACE ARTICULAIRE [2] [3]

On considère l'application des techniques de la commande non linéaire les plus complexes pour la poursuite des trajectoires par les robots manipulateurs. On considère les équations dynamiques d'un robot manipulateur à n degrés de liberté sous forme matricielle

$$A(q)\ddot{q} + C(q, \dot{q})\dot{q} + B\dot{q} + g(q) = u \quad (4.1)$$

Pour simplifier on écrit l'équation précédente sous la forme suivante :

$$A(q)\ddot{q} + h(q, \dot{q}) = u \quad (4.2)$$

Où $h(q, \dot{q}) = C(q, \dot{q})\dot{q} + B\dot{q} + g(q)$. L'idée de la dynamique inverse est de rechercher une loi de commande par bouclage non linéaire:

$$u = f(q, \dot{q}) \quad (4.3)$$

Laquelle, substituée dans (4.2), donne un système en boucle fermée linéaire. Pour la majorité des systèmes non linéaires une telle loi de commande peut être tout à fait difficile voire impossible de la trouver. Dans le cas des équations dynamiques du manipulateur (4.2), cependant, le problème est actuellement facile. En inspectant (4.2) on voit que si on choisit la commande u selon l'équation

$$u = A(q)v + h(q, \dot{q}) \quad (4.4)$$

Alors, puisque la matrice d'inertie est inversible, le système combiné (4.2), (4.4) se réduit à

$$\ddot{q} = v \quad (4.5)$$

Le terme v représente une nouvelle entrée du système qui est encore à déterminer. L'équation (4.5) est un système double intégrateur et elle représente n double intégrateurs découplés. La loi de commande non linéaire (4.4) est appelée la commande par dynamique inverse (appelée aussi commande par couple calculé) et réalise par là un résultat remarquable, à savoir que le nouveau système (4.5) est linéaire et découplé. Cela signifie que chaque entrée v_k peut être calculée pour commander un système linéaire mono variable. D'ailleurs, en supposant que v_k est fonction seulement de q_k et ses dérivées, alors v_k affecterait q_k indépendamment du mouvement des autres articulations.

Puis que v_k peut maintenant être synthétisée pour commander un système linéaire simple du second ordre, le choix évident est de poser :

$$v = -K_0 q - K_1 \dot{q} + r \quad (4.6)$$

Où K_0 , K_1 sont deux matrices diagonales dont les éléments diagonaux représentent les gains de position et de vitesse, respectivement. Le système en boucle fermée est donc le système linéaire

$$\ddot{q} + K_1 \dot{q} + K_0 q = r \quad (4.7)$$

Maintenant, étant donné une trajectoire désirée

$$t \rightarrow (q_d(t), \dot{q}_d(t)) \quad (4.8)$$

Si on choisit l'entrée de référence $r(t)$ comme suit :

$$r(t) = \ddot{q}_d(t) + K_1 \dot{q}_d(t) + K_0 q_d(t) \quad (4.9)$$

Alors l'erreur de poursuite $e(t) = q_d(t) - q(t)$ satisfait l'équation :

$$\ddot{e}(t) + K_1 \dot{e}(t) + K_0 e(t) = 0 \quad (4.10)$$

Un choix évident pour choisir les matrices des gains est le suivant :

$$\begin{aligned} K_0 &= \text{diag}[\omega_1^2, \dots, \omega_n^2] \\ K_1 &= \text{diag}[2\omega_1, \dots, 2\omega_n] \end{aligned} \quad (4.11)$$

Ce qui conduit à un système en boucle fermée qui est globalement découplé, où chaque réponse articulaire sera celle d'un système linéaire du second ordre avec amortissement critique à une fréquence naturelle ω_i . En effet, la fréquence naturelle ω_i détermine le temps de réponse de l'articulation.

L'approche de la dynamique inverse est extrêmement importante comme base pour la commande des robots manipulateurs et il est intéressant d'essayer de la représenter par une autre approche. On peut donner une deuxième interprétation de la loi de commande (4.4) comme suit. On considère à nouveau les équations dynamiques du manipulateur (4.2). Puisque $A(q)$ est inversible pour $q \in R^n$ on peut dire que l'accélération du manipulateur est alors :

$$\ddot{q} = A^{-1} \left[u - h(q, \dot{q}) \right] \quad (4.12)$$

Supposons qu'on puisse spécifier l'accélération comme entrée du système et que nous disposions des actionneurs aptes à produire directement une accélération désirée (plutôt que de produire directement une force ou un couple). Alors les dynamiques du manipulateur, qui est après tout un dispositif de commande de position, pourraient être données par :

$$\ddot{q}(t) = v(t) \quad (4.13)$$

Où $v(t)$ représente le vecteur des accélérations d'entrée. On aboutit à nouveau au système double intégrateur. Il est à remarquer que (4.13) n'est en aucun sens une approximation; plutôt elle représente la dynamique en boucle ouverte actuelle du système à condition que l'accélération soit choisie comme entrée. Le problème de commande du système (4.13) est maintenant facile et l'accélération d'entrée peut être choisie comme indiqué précédemment selon (4.6).

En réalité, de tels actionneurs n'existent pas, par contre on a la possibilité de produire une force généralisée (couple) u , pour chaque articulation. En comparant les équations (4.12) et (4.13) on constate que le couple et l'accélération sont liés par :

$$A^{-1} \left[u(t) - h(q, \dot{q}) \right] = v(t) \quad (4.14)$$

La matrice d'inertie étant inversible, on peut alors résoudre pour l'entrée couple $u(t)$ comme suit :

$$u(t) = A(q)v(t) + h(q(t), \dot{q}(t)) \quad (4.15)$$

Qui est la même que l'expression dérivée précédemment (4.4). Ainsi le couple calculé peut être vu comme étant une transformation d'entrée de commande, vu la difficulté d'utiliser l'accélération comme entrée de commande.

On donne ci-dessous le schéma représentatif de la première structure de commande :

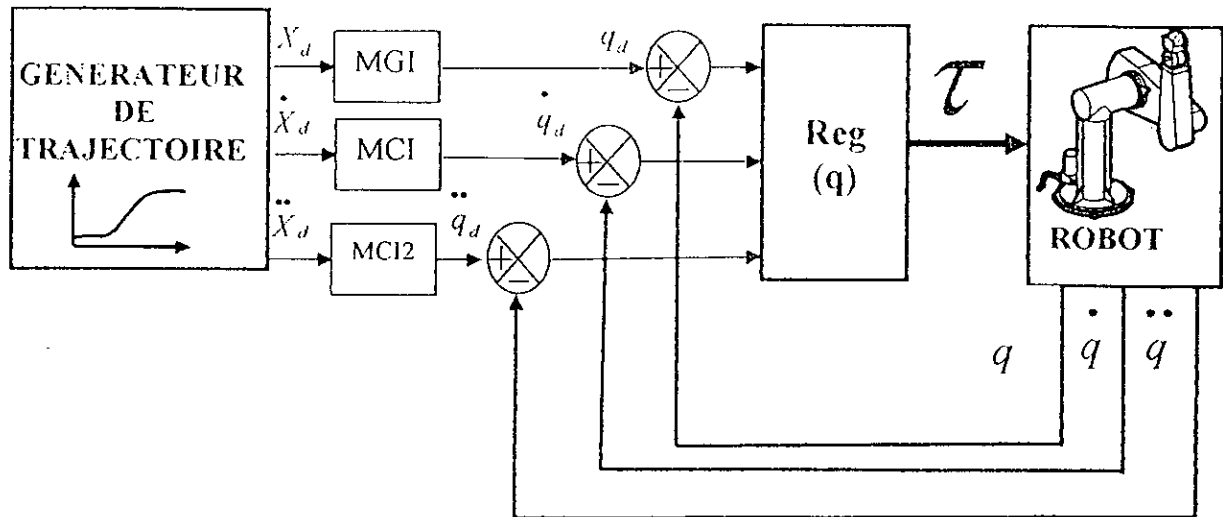


Fig. 4.1 structure de commande avec correction dans l'espace articulaire

Dans cette structure de commande la commande est calculée à base des erreurs articulaires ce qui exige l'utilisation des modèles inverses qui expriment les grandeurs articulaires à partir des grandeurs opérationnelles décrivant la tâche dans l'espace de travail.

4.3 COMMANDE DANS L'ESPACE OPERATIONNEL AVEC CORRECTION DANS L'ESPACE OPERATIONNEL [2] [3]

En substituant dans la relation (4.1) \ddot{q} par l'équation

$$\ddot{q}_d = J^{-1}(\ddot{X}_d - \dot{J}\dot{q}) \tag{4.16}$$

Où : $X_d, \dot{X}_d, \ddot{X}_d$ désignent la position, la vitesse et l'accélération opérationnelles désirées.

Le comportement dynamique du robot dans l'espace opérationnel est décrit par l'équation suivante (en omettant l'exposant d) :

$$u = AJ^{-1}(\ddot{X} - \dot{J}\dot{q}) + h(q, \dot{q}) \tag{4.17}$$

Comme il a été fait dans la structure précédente, une loi de commande qui linéarise et découple les équations dans l'espace opérationnel est donnée par :

$$u = AJ^{-1}(v(t) - \dot{J}\dot{q}) + h(q, \dot{q}) \tag{4.18}$$

Avec cette loi et en supposant un modèle parfait, le système est régi par l'équation du double intégrateur dans l'espace opérationnel suivante :

$$\ddot{X}(t) = v(t) \tag{4.19}$$

On détaille ici une correction PD, on pose alors :

$$v(t) = \ddot{X}_d + K_1(\dot{X}_d - \dot{X}) + K_0(X_d - X) \tag{4.20}$$

Avec cette loi, dans l'hypothèse d'une modélisation parfaite et d'une erreur initiale nulle, le comportement du robot n'est décrit par l'équation :

$$\ddot{e}_x + K_1 \dot{e}_x + K_0 e_x = 0 \tag{4.21}$$

Avec :
$$e_x = X_d - X \tag{4.22}$$

On donne ci-dessous le schéma représentatif de la deuxième structure de commande :

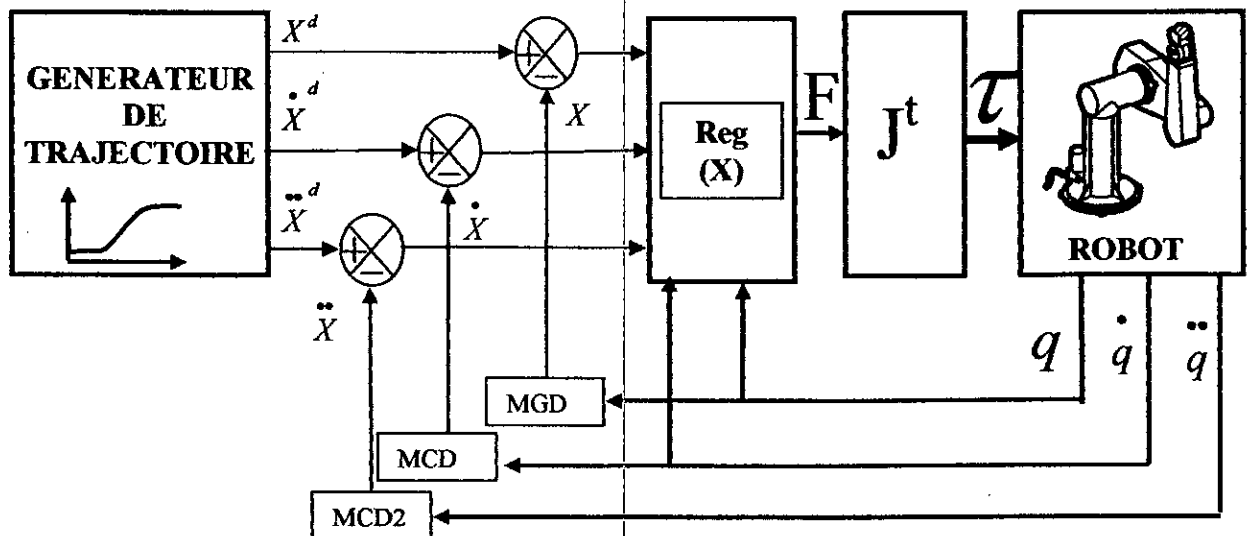


Fig. 4.2 structure de commande avec correction dans l'espace opérationnel

Dans cette structure de commande la commande est calculée à base des erreurs opérationnelles directement sans faire appel aux modèles inverses.

On note que l'implémentation de cette loi de commande exige le calcul de la matrice d'inertie $A(q)$, le vecteur des termes de coriolis, gravitationnels, de centrifuge et d'amortissement $h(q, \dot{q})$ en plus du terme J .

Cependant, la commande doit être calculée en ligne. Autrement dit, comme toute loi de commande par retour de sortie, elle peut être pré calculée off line et mémorisée comme dans le cas du couple calculé. Une question importante est liée à l'implémentation de la commande et la conception de l'architecture du calculateur pour les calculs précédents. Une méthode attirante pour implémenter ces calculs, et une qui va assurément se généraliser dans l'avenir, est d'utiliser une interface hardware dédiée, est une unité de calcul spéciale sous forme d'un circuit VLSI pour exécuter les calculs exigés en temps réel.

La méthode du couple calculé est ainsi très attirante du point de vue commande car le fort couplage dynamique non linéaire du manipulateur a été éliminé, et remplacé par un simple système linéaire découplé du second ordre. Cependant, de tels schémas d'élimination laissent ouvert beaucoup de processus liés à sensibilité et à la robustesse qui doivent être abordées.

4.4 ÉTUDE DE LA ROBUSTESSE DE LA COMMANDE PAR DECOUPLAGE NON LINEAIRE [2]

L'implémentation pratique de la loi de commande par couple calculé nécessite la connaissance des paramètres du modèle dynamique avec précision de plus l'équation complète du mouvement doit être calculée en temps réel, typiquement environ 60 MHz. Ces deux exigences sont difficiles à satisfaire en pratique. Dans tout système physique il y a un certain degré d'incertitude lorsque les valeurs des paramètres varient en fonction du temps.

Ceci est particulièrement vrai dans le cas d'un système assez compliqué comme le cas d'un robot, surtout si le robot supporte des charges inconnues. En pratique, il y aura toujours une inexacte annulation des non linéarités due à cette incertitude et aussi à l'arrondi dans les calculs, etc. En outre, le calcul en temps réel du modèle complet du manipulateur doit être lourd ou impossible dans les limites imposées par certaines architectures disponibles du calculateur. Dans ces cas-ci il est demandé de simplifier les équations du mouvement autant que possible en ignorant certains termes pour accélérer le calcul de la loi de commande.

$$u(t) = \hat{A}(q)v(t) + \hat{h}(q(t), \dot{q}(t)) \quad (4.23)$$

Où : \hat{A} , \hat{h} représentent la version nominale et calculée de A , h , respectivement. L'incertitude Ou erreur de modélisation, représentée par :

$$\Delta A = \hat{A}(q) - A(q) \quad (4.24)$$

$$\Delta h = \hat{h}(\dot{q}, q) - h(\dot{q}, q) \quad (4.25)$$

Est alors due aux (et est une mesure de) problèmes d'incertitudes sur les paramètres, etc. Il est à noter également que l'erreur de modélisation ci-dessus peut surgir en raison de la simplification intentionnelle du modèle. En effet, dans le cas extrême, on peut prendre $\hat{A} = I$ et $\hat{h} = 0$ dans ce cas la loi de commande se réduit en la boucle de commande v , qui pourrait par exemple simplement être une commande PID indépendante pour chaque articulation.

Avec la loi de commande non linéaire (4.23), le système devient :

$$A\ddot{q} + h = \hat{A}v + \hat{h} \quad (4.26)$$

Ainsi \ddot{q} peut être exprimée comme suit :

$$\begin{aligned} \ddot{q} &= A^{-1}\hat{A}v + A^{-1}\Delta h \\ &= v + (A^{-1}\hat{A} - I)v + A^{-1}\Delta h \end{aligned} \quad (4.27)$$

Où: $\Delta h = \hat{h} - h$ En définissant $E = A^{-1}\hat{A} - I$ et après arrangement on obtient :

$$v = \ddot{q}_d - K_1(\dot{q} - \dot{q}_d) - K_0(q - q_d) \quad (4.28)$$

On peut écrire l'équation précédente en fonction de l'erreur de poursuite $e = q - q^d$ comme suit :

$$\ddot{e} + K_1\dot{e} + K_0e = \eta \quad (4.29)$$

Où η , ci-après appelée l'incertitude, est donnée par l'expression :

$$\begin{aligned}\eta &= Ev + A^{-1} \Delta h \\ &= E(\ddot{q}_d - K_1 \dot{e} - K_0 e) + A^{-1} \Delta h\end{aligned}\tag{4.30}$$

Il est à noter que le système (4.29) reste un système non linéaire couplé tant que η est une fonction non linéaire de e . donc il n'est pas évident que le système soit stable même si on augmente les gains suffisamment du fait que la fonction non linéaire η dépend aussi de v , donné par l'équation (4.28), et par conséquent il peut s'accroître avec l'accroissement du gain.

4.5 SIMULATIONS ET INTERPRETATIONS :

La simulation consiste en l'implémentation de la commande et la visualisation des résultats pour voir la réponse du robot manipulateur vis-à-vis d'une trajectoire désirée.

Pour la résolution de l'équation dynamique du système on a utilisé l'algorithme de runge-kutta d'ordre 4.

Dans un premier temps on génère la trajectoire dans le plan décrite dans la figure 3.5, puis on génère une trajectoire spatiale, la génération de trajectoire est déjà mentionnée dans le chapitre 3 (loi Bang-Bang avec palier de vitesse).

Simulation 1 : (trajectoire planeaire)

1-Les simulations qui viennent concernent la première structure de commande qui est synthétisée dans l'espace articulaire.

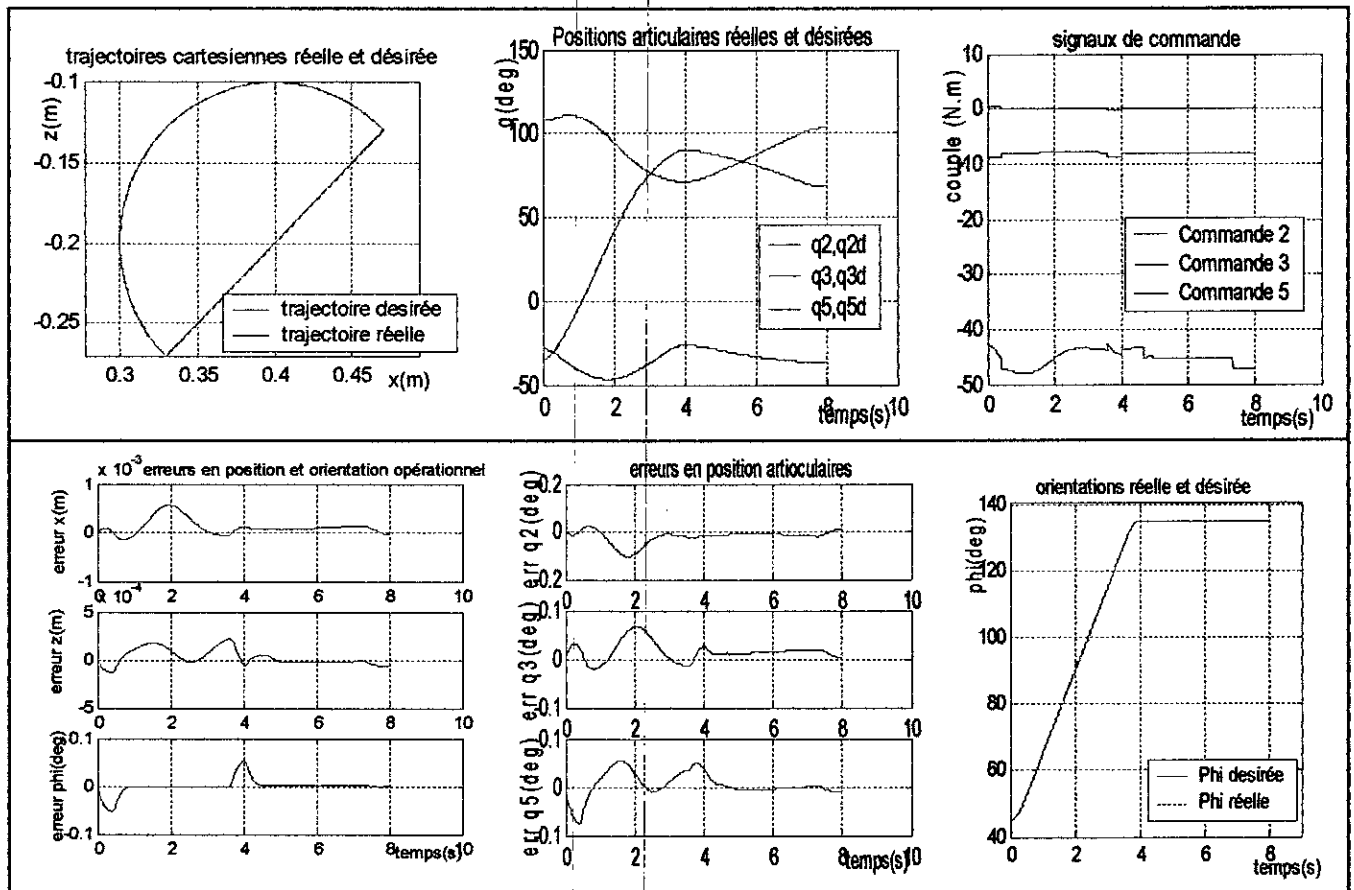


Fig.4.3 Réponse du robot en position et orientation (structure 1).

On constate une poursuite de la trajectoire désirée (les trajectoires réelle et désirée sont presque confondues), ce qui est confirmé par les erreurs de poursuite cartésiennes et articulaires.

Au démarrage, l'erreur varie entre 0.1mm et 0.5 mm, dans le premier tronçon de la trajectoire, et se stabilise environ 0.05 mm dans le deuxième tronçon. Ceci est dû à la nature de la trajectoire (circulaire ou rectiligne) qui engendre une variation rapide de la configuration du robot pour le premier tronçon et une légère variation de celle-ci dans le second tronçon surtout pour l'articulation 2 qui subit une forte variation entre $[-30^{\circ}, 90^{\circ}]$ dans un intervalle de 4 sec qui est le temps de parcours de l'arc. Il en est de même pour les erreurs articulaires, qui varient entre 0.01° et 0.05° .

En ce qui concerne l'orientation de l'outil terminal, on constate qu'elle poursuit l'orientation désirée à l'exception du démarrage et à la fin du premier tronçon où apparaît un pic d'erreur d'environ 0.05°.

Les couples appliqués aux articulations sont admissibles par les actionneurs utilisés. Concernant l'articulation 2, l'actionneur développe un couple important environ 50 N.m pour vaincre l'inertie de l'articulation qui pèse 17 Kg ainsi que l'étendue de la grandeur articulaire. Quant au poignet (articulation 5), le couple est très faible vu sa faible inertie et la légère étendue de la grandeur articulaire.

2- Les simulations qui viennent concernent la deuxième structure de commande qui est synthétisée dans l'espace opérationnel.

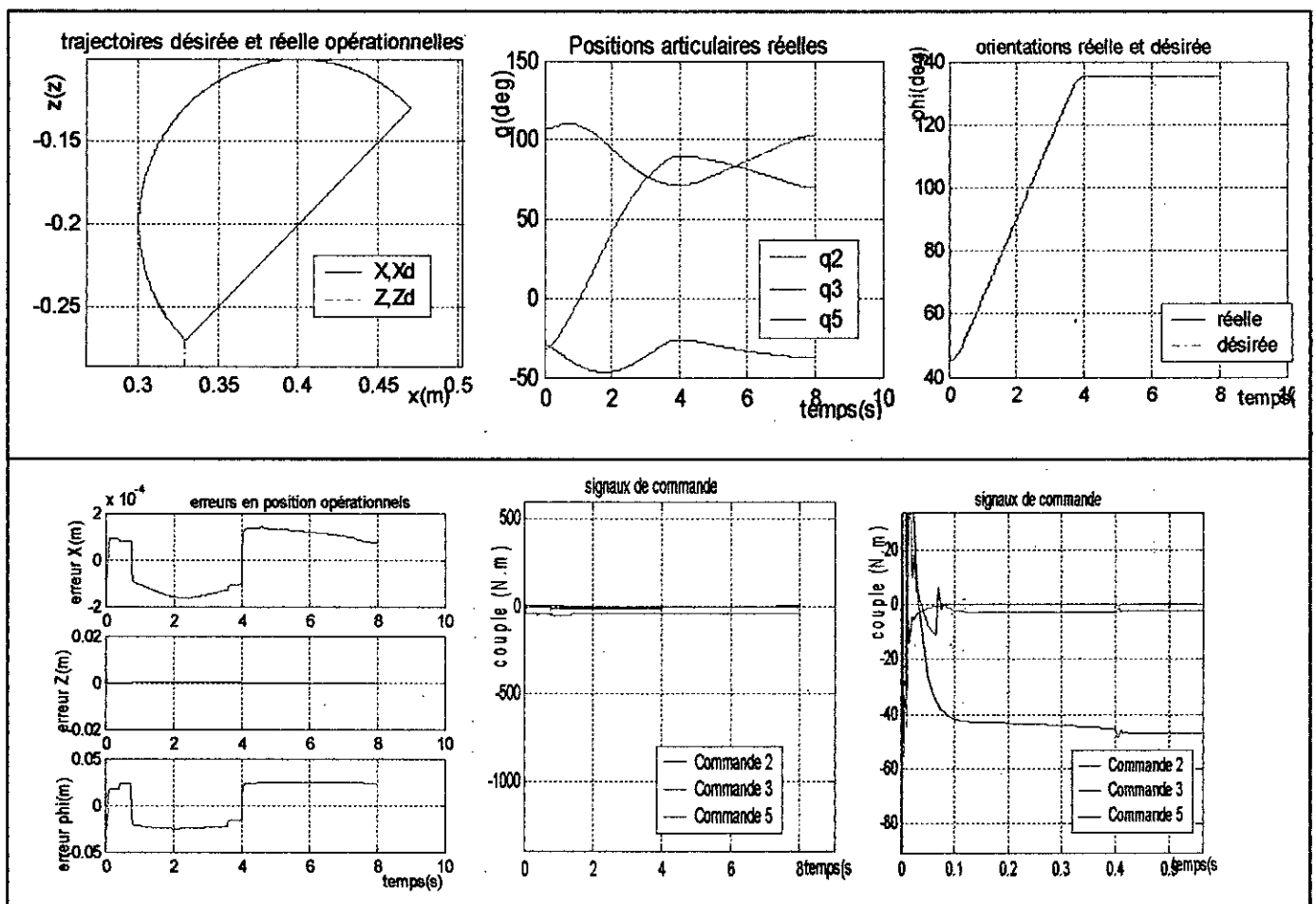


Fig.4.4 Réponse du robot en position et orientation (structure 2)

La poursuite de la trajectoire est réalisée, et on constate que l'erreur au démarrage est importante, cela est dû au fait que le point initial n'appartient pas à la trajectoire désirée.

La commande est importante au démarrage mais admissible (environ 1000 N.m pour la deuxième articulation) pour la simple raison que l'erreur est aussi importante.

Simulation 2 : (trajectoire spatiale)

1-Les simulations qui viennent concernent la première structure de commande qui est synthétisée dans l'espace articulaire.

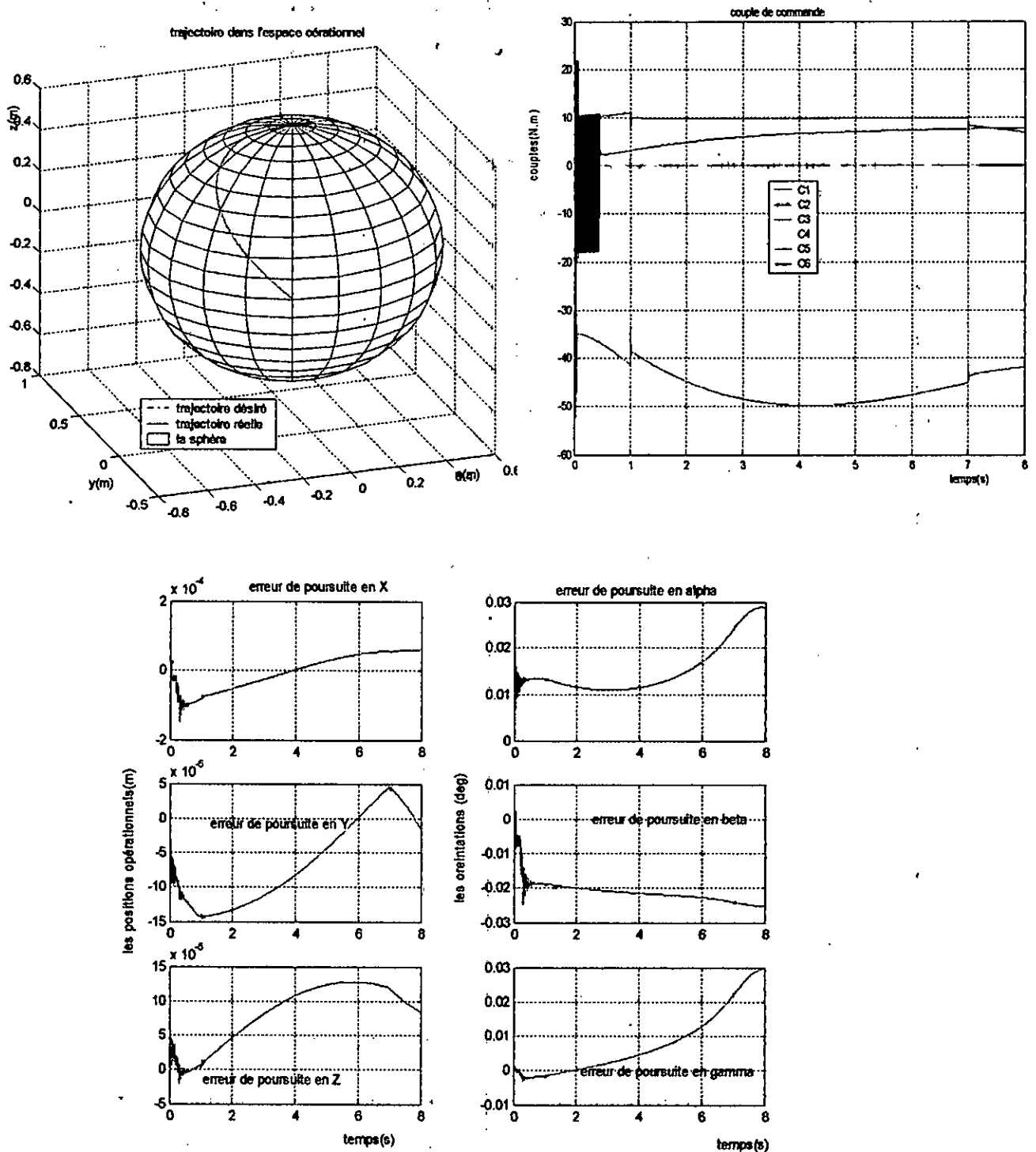


Fig.4.5 Réponse du robot en position et orientation (structure 1)

On constate une bonne poursuite de la trajectoire opérationnelle et des erreurs négligeables de l'ordre de quelques centièmes de mm pour la position et de quelques centièmes de degrés pour l'orientation.

Le couple de commande est un peu oscillatoire (articulation 3) au démarrage pourvu que la configuration du robot change fortement, mais il reste dans les limites tolérées par les actionneurs. Quant aux trois dernières articulations, leurs commandes sont trop faibles par ce que le robot est à vide (sans charge).

2- Les simulations qui viennent concernent la deuxième structure de commande qui est synthétisée dans l'espace opérationnel.

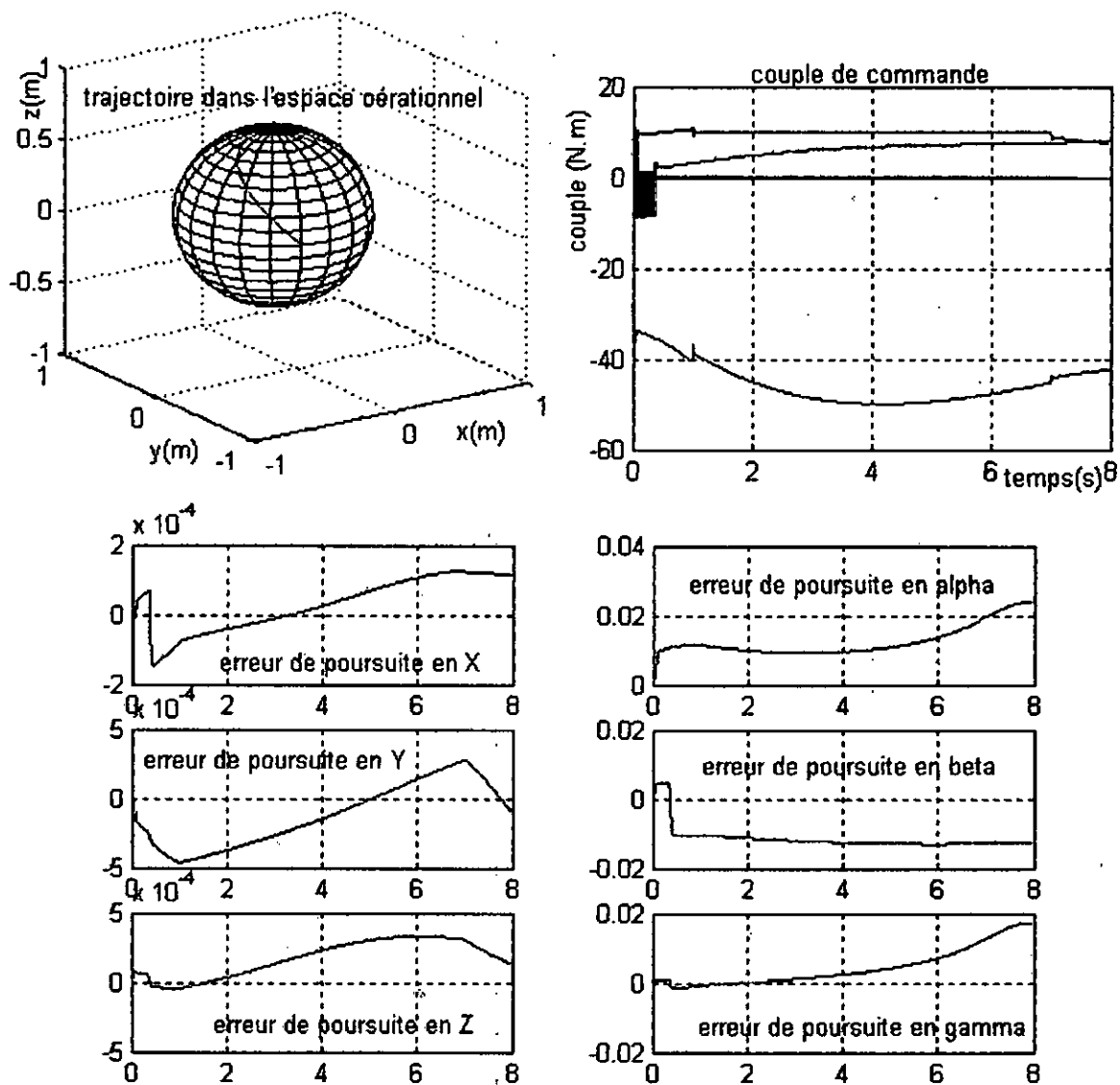


Fig.4.6 Réponse du robot en position et orientation (structure 2)

On constate une bonne poursuite de la trajectoire opérationnelle avec des erreurs faibles légèrement supérieures à celles obtenues dans la première structure de commande synthétisée dans l'espace articulaire.

Le couple de commande reste admissible par les actionneurs avec un couple important pour les trois premières articulations, quant aux trois dernières articulations, leurs commandes sont trop faibles.

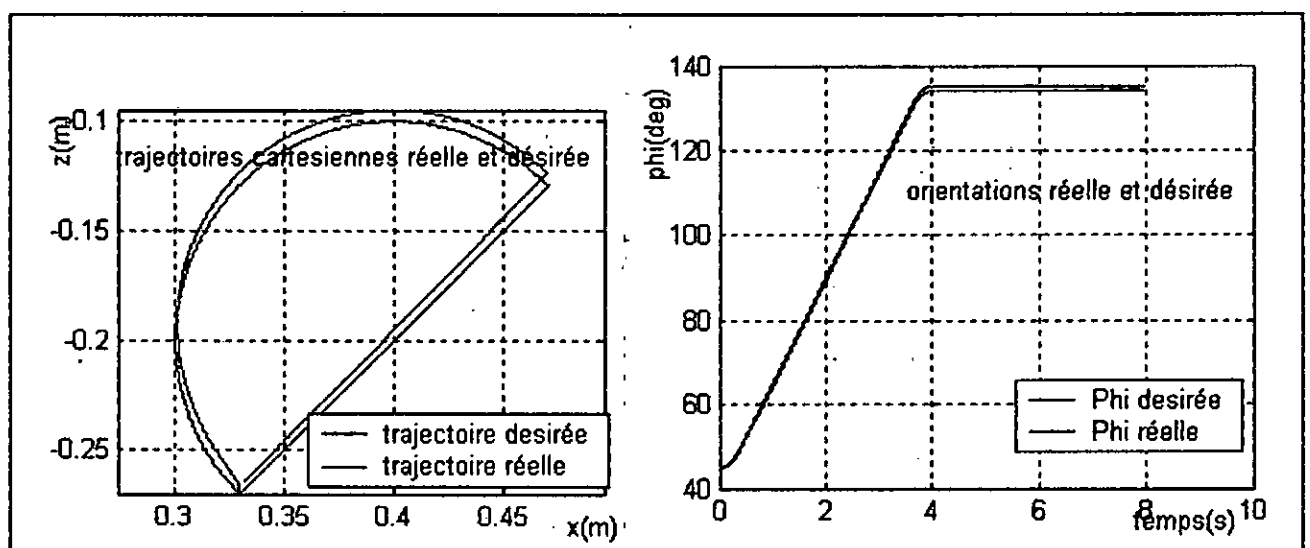
Conclusion1 :

Dans ce qui précède, nous avons effectué des simulations pour deux structures de commande différentes. La première consistait à calculer la commande en se basant sur les erreurs articulaires et la seconde calculait la commande à base d'erreurs opérationnelles.

En analysant et en comparant les résultats obtenus, pour les deux structures et pour les deux trajectoires, on trouve que les résultats obtenus par correction dans l'espace articulaire (première structure) sont meilleurs en terme de précision, gains de commande et temps de réponse du fait que le robot est commandé dans son espace naturel (espace des articulations).

Simulation3 : La commande par couple calculé avec une erreur de modélisation de 15%.

Pour étudier la robustesse de la commande, on a calculé la commande à base du modèle incertain qui présentait une augmentation de 15% du modèle nominale, et on a effectué les simulations sur le modèle certain (exacte). Les résultats obtenus sont donnés dans la figure suivante :



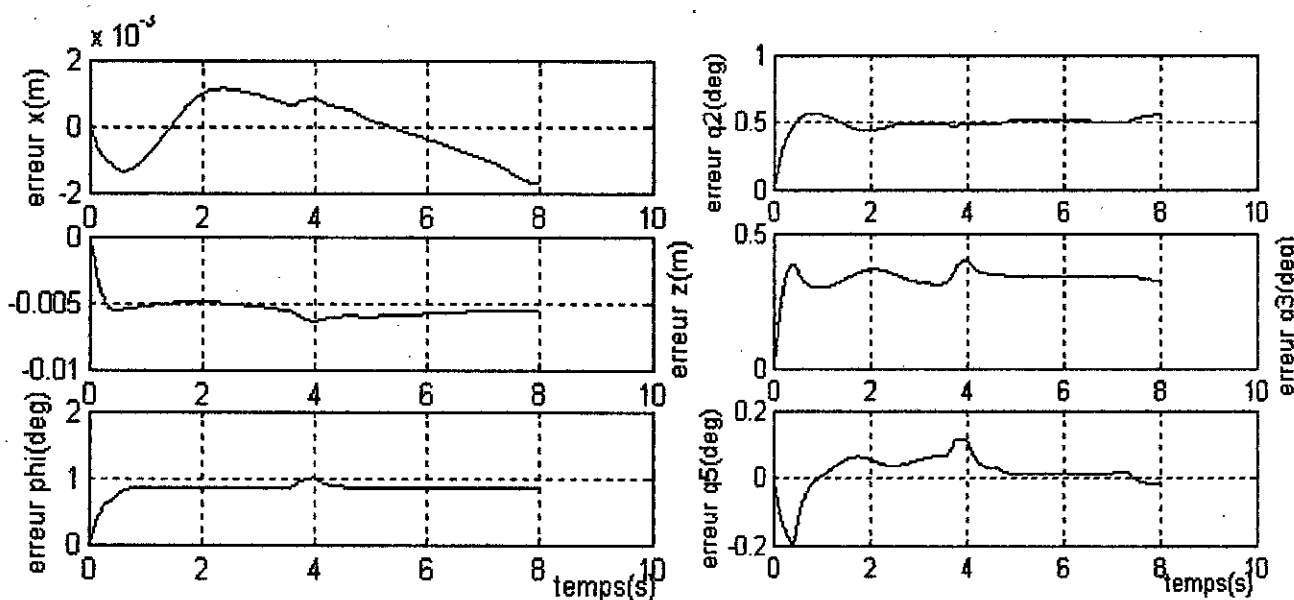


Fig.4.7 Réponse du robot en position et orientation (structure 1)

D'après la figure 4.7 on constate qu'il y a une erreur de poursuite importante du fait qu'elle atteint quelques centimètres pour la position cartésienne et quelques degrés pour l'orientation. La poursuite n'est pas réalisée ni en position ni en orientation, cela est dû au fait qu'on ne connaît pas le modèle avec certitude, à base duquel la commande a été calculée.

Conclusion 2:

La dépendance explicite du modèle lors du calcul de la commande par découplage non linéaire a montré qu'elle n'est pas robuste vis-à-vis des erreurs de modélisations. Car dans la plus part des cas le robot n'est pas complètement modélisé, mais plutôt on ne peut modéliser généralement que la matrice d'inertie.

Pour remédier à ce problème on essaiera de trouver une solution pour enlever la nécessité du modèle certain pour la commande par couple calculé, on montre dans la partie qui suit une approche pour compenser la partie non modélisable par un terme à adaptation floue.

4.6 ROBUSTIFICATION DE LA COMMANDE [11] [15] [16]

En robotique, une grande partie des approches de commande ont été proposées pour commander le robot à poursuivre une trajectoire désirée. En raison de la complexité de la dynamique du robot, plusieurs structures de commande tendent à simplifier ou également, à négliger le modèle mathématique du robot manipulateur. Une structure de commande simple qui est souvent utilisée aux applications des robots industriels est dite commande articulaire indépendante (*independant joint control* en Anglais) ou commande classique. Cette commande ignore complètement le modèle dynamique du robot et le traite comme étant des perturbations extérieures. Cependant, son utilisation est limitée aux faibles vitesses, au positionnement et aux applications de poursuite imprécises.

L'objectif de la commande est de forcer l'outil terminal du robot à poursuivre une trajectoire désirée dans l'espace cartésien ce qui revient à poursuivre une trajectoire articulaire correspondante.

Laissons nous définir le vecteur des erreurs :

$$e = \begin{bmatrix} e_1 \\ e_2 \end{bmatrix} = \begin{bmatrix} q_d - q \\ \dot{q}_d - \dot{q} \\ q_d - q \end{bmatrix} \quad (4.31)$$

Pour accomplir l'objectif de la commande, il est souhaitable de forcer la fonction coût n-dimensionnelle à zéro.

$$J(e_1, e_2) = C e_1 + e_2 = C e_1 + \dot{e}_1 \rightarrow 0 \quad (4.32)$$

Le régulateur proposé a une structure variable et est donné par: [11]

$$u = \tau = A(q)[\ddot{q}_d + C \dot{e}_1 + K \text{sign}(J)] \quad (4.33)$$

Où : K est un gain qu'on choisit au premier lieu, constant et qu'on adaptera par la suite par un système flou suivant les règles floues suivantes :

A chaque instant t, $J_i(x(t))$ est la valeur algébrique de la fonction coût J_i ($i=1, \dots, n$).

Ces règles sont données comme suit :

Règle 1 : si $|J_i(x(t))|$ est G alors $K_i(t)$ est G

Règle 2 : si $|J_i(x(t))|$ est M alors $K_i(t)$ est M

Règle 3 : si $|J_i(x(t))|$ est P alors $K_i(t)$ est P

Règle 4 : si $|J_i(x(t))|$ est Z alors $K_i(t)$ est Z

Où : G, M, P, Z signifient : grand, moyen, petit, nul respectivement.

Etant donné la valeur de $|J_i(x(t))|$ à l'instant t , la valeur du gain de la commande K_i est impliquée en utilisant les quatre règles floues précédemment définies, et la défuzzification par centre de gravité.

Cette stratégie de détermination du gain de la commande $K(t)$ présente les avantages suivant par rapport au gain fixe :

- Un grand gain est appliqué seulement lorsque la valeur algébrique de la fonction coût est grande.
- Quant la fonction coût est proche du zéro on utilise un petit gain. Par conséquent, cela présente une solution potentielle au problème de la réduction du zigzag du signal de commande à structure variable (*chattering*).

Remarque : on peut utiliser la fonction saturation (*sat*) pour éliminer le *chattering* et adoucir le signal de commande. On donne ci-dessous son graphe représentatif :

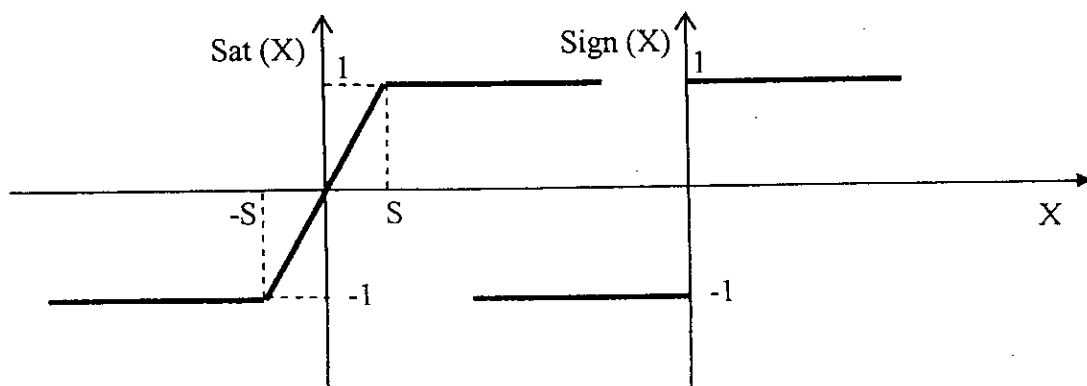


Fig.4.7 Allures des fonctions Sat et Sign

Cette partie propose la conception d'un régulateur pour la poursuite de trajectoire par un robot manipulateur en utilisant la logique floue. Le régulateur proposé est basé sur l'intégration la commande conventionnelle et la logique floue.

Les Systèmes flous [14] [12]

Les systèmes flous sont les systèmes qui ont une relation directe avec les concepts flous (ensembles flous, variables linguistiques... etc.) et la logique floue.

Un système flou est un système à base de connaissance particulière. L'architecture de base d'un système flou (figure 4.8) se compose de quatre modules principaux :

- La base des règles floues, ou base de connaissances, contient les règles floues décrivant le comportement du système; elle est le cœur du système entier dans le sens où tous les autres composants sont utilisés pour interpréter et combiner ces règles pour former le système final.
- Le moteur d'inférence flou transforme, à l'aide des techniques de raisonnement flou la partie floue issue de la fuzzification en une nouvelle partie floue.
- La fuzzification transforme la grandeur physique d'entrée en une grandeur floue.
- La défuzzification transforme la grandeur floue issue de l'inférence en une grandeur physique.

Le schéma suivant présente la configuration de base d'un système flou :

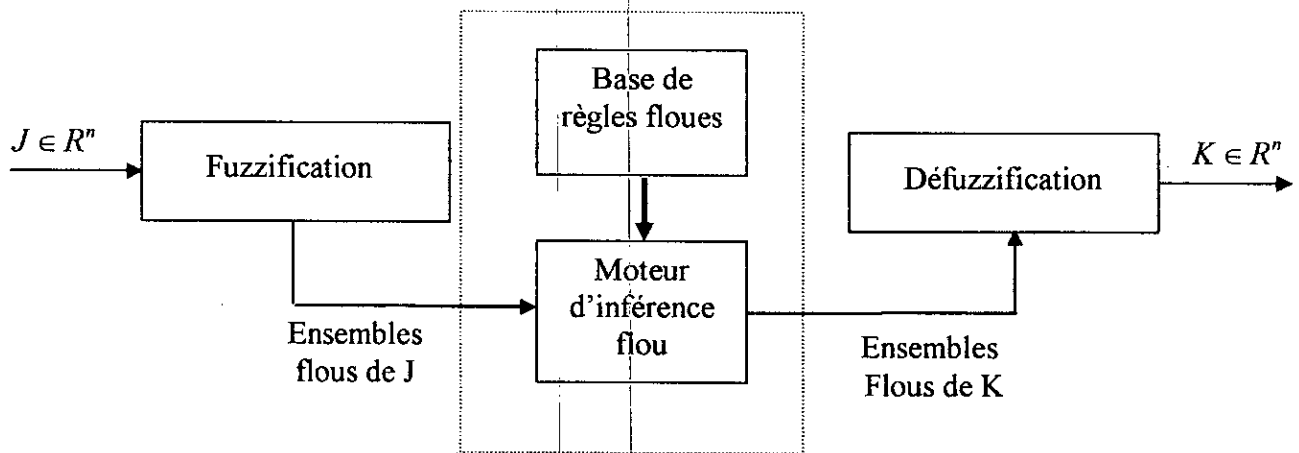
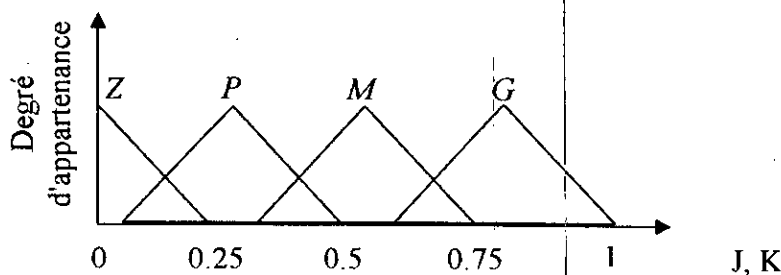


Fig 4.8 Configuration de base d'un système flou

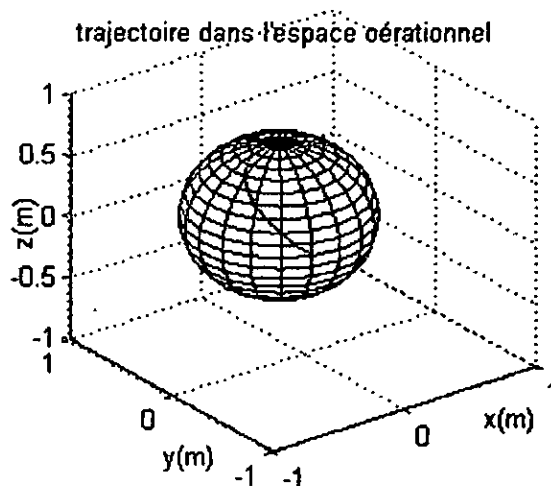
Les fonctions d'appartenance d'entrée et de sortie sont données par :



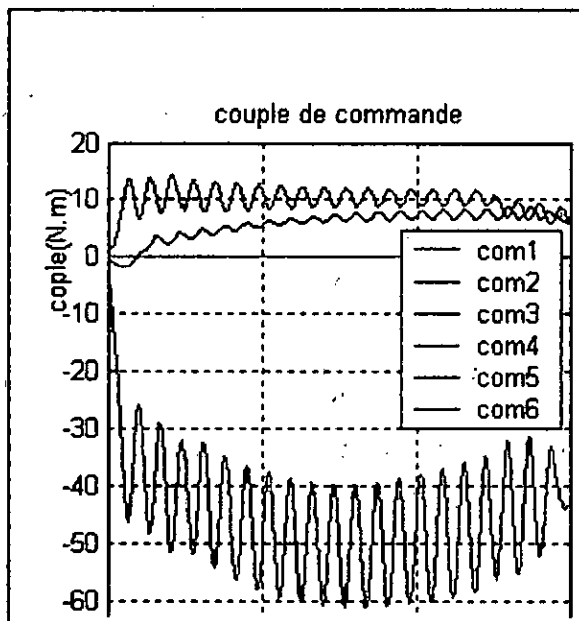
Résultats de simulation (robustification de la commande)

Dans cette partie de simulation on a ajouté un terme flou de correction pour compenser la partie non modélisable, et on a ajouté des erreurs au modèle pour voir la robustesse de la commande.

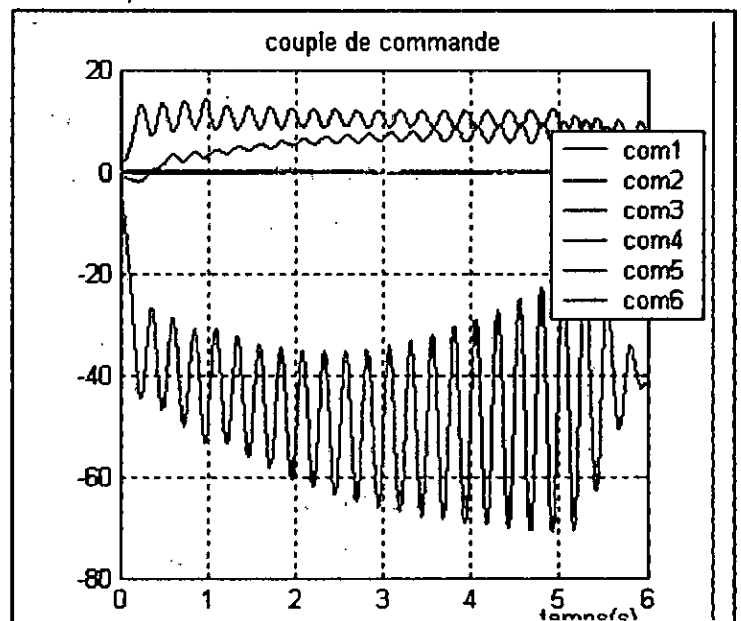
Nous sommes allés jusqu'au 30% comme erreurs de modélisation par rapport au modèle nominal ; les résultats de simulations sont donnés dans les figures qui suivent :



Simulation sans erreur de modélisation



Simulation avec 30% erreur dans le modèle



Simulation sans erreurs de modélisation et Simulation avec 30% d'erreurs dans le modèle

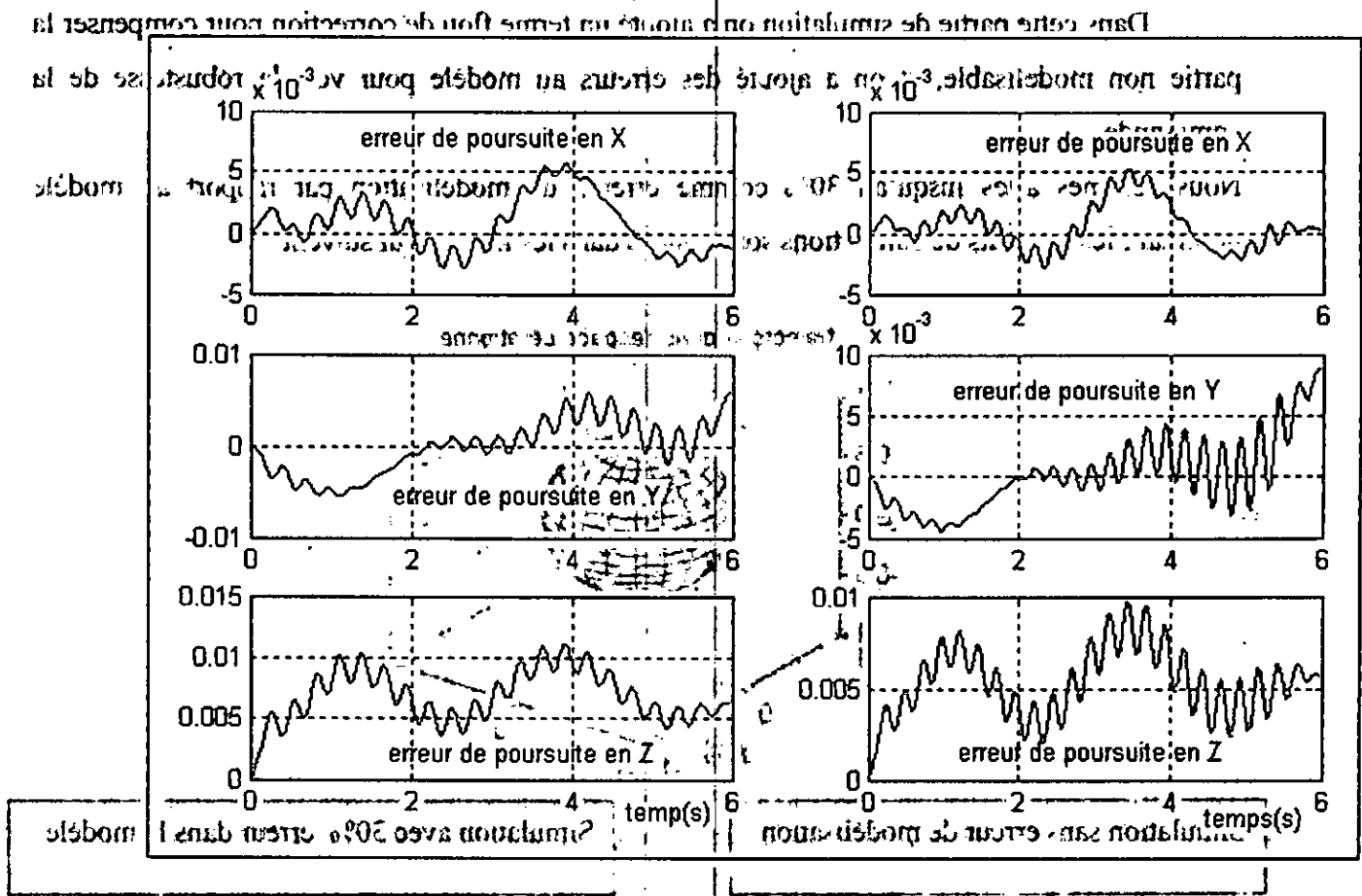


Fig.4.9 Réponse du robot en position et orientation.

D'après les simulations on constate qu'on a surmonté le problème des parties incertaines dans le modèle, les résultats obtenus montrent que même si on modélise le système avec une erreur de 30% dans le modèle et on peut aller plus, il n'y aura pas d'influence sur la poursuite.

Avec des couples d'actionneur un peut oscillatoire mais admissibles.

Conclusion :

L'introduction d'un terme de compensation calculé à base d'un adaptateur de gain flou a donné de bons résultats, et on est arrivé à robustifier la commande par découplage non linéaire. Le calcul de la commande ne nécessite pas le modèle complet du robot, et cela a permis de diminuer la sensibilité aux erreurs de modélisation.

Synthèse De La Commande Basée Sur Le Principe De Lyapunov

5.1 INTRODUCTION [8] [9]

Dans la conception des commandes robustes des systèmes non linéaires incertains, la théorie de la stabilité joue un rôle important. Dans l'analyse des systèmes automatiques, il est crucial d'avoir un système stable, puisqu'un système de commande instable est inutile. Le critère de stabilité de Lyapunov est une procédure générale, et utile pour étudier la stabilité des systèmes non linéaires. La théorie de stabilité de Lyapunov inclut deux méthodes: la méthode directe et la méthode indirecte.

La première méthode de Lyapunov est une technique qui exploite simplement l'idée de la linéarisation d'un système (*lowest order approximation*) autour d'un point de fonctionnement donné et peut seulement donner des résultats locaux de stabilité dans des régions restreintes.

La méthode directe de Lyapunov est l'outil le plus important pour la conception et l'analyse des systèmes non linéaires. La méthode directe de Lyapunov est directement appliquée aux systèmes non linéaires sans avoir besoin à la linéarisation et étudie ainsi la stabilité globale.

Le concept de base derrière la méthode directe de Lyapunov est que si toute l'énergie d'un système, électrique /mécanique; linéaire ou non linéaire, se dissipe continûment, alors le système par la suite atteint un point d'équilibre et reste en ce point. Par conséquent, la méthode directe de Lyapunov inclut deux étapes:

- La première étape consiste à trouver d'abord une fonction scalaire appropriée, dite fonction de Lyapunov.
- La deuxième étape consiste à évaluer sa dérivée temporelle du premier ordre le long de la trajectoire du système. Si la dérivée de la fonction de Lyapunov diminue le long de la trajectoire du système à mesure que le temps augmente, alors l'énergie du système se dissipe et le système se fixera ainsi par la suite.

Les définitions ci-dessous donnent un rapport plus formel des choix admissibles de fonction de Lyapunov.

5.2 NOTIONS PRELIMINAIRES [8] [9] [10]

5.2.1 Points d'Equilibre

La dynamique des systèmes non linaires peut être représentée par une équation de la forme :

$$\dot{x} = f(x, u, t) \quad (5.1)$$

Où x est le vecteur d'état.

U est le vecteur de commande.

f est un champ de vecteur non linéaire et lipchitzien.

Dans la théorie de la commande on classe les systèmes non linaires en deux classes :

➤ La classe des systèmes autonomes : $\dot{x} = f(x, u)$ (5.2)

➤ La classe des systèmes non autonomes : $\dot{x} = f(x, u, t)$ (5.3)

Si le signal de commande $u = 0$ alors le système est dit libre sinon il est forcé.

Définition : le point (x_e, u_e) est un point d'équilibre du système (1) si et seulement si :

$$0 = f(x_e, u_e, t) \quad \forall t \geq 0 \quad (5.4)$$

Remarque : Le point d'équilibre (x_e, u_e) peut être ramené à l'origine par la transformation :

$$z = x - x_e, v = u - u_e$$

Dans ce cas on a :

$$\dot{z} = f(z + x_e, v + u_e, t) \quad \text{Avec} \quad f(0, 0, t) = 0 \quad (5.5)$$

Exemple 1 :

La dynamique d'un pendule est donnée par :

$$m l^2 \ddot{\theta} + m y l \sin(\theta) - k l \dot{\theta} = 0 \quad (5.6)$$

Posons $x_1 = \theta, x_2 = \dot{\theta}$, les équations d'états deviennent

$$\begin{aligned}\dot{x}_1 &= x_2 \\ \dot{x}_2 &= \frac{k}{ml} x_2 - \frac{g}{l} \sin(x_1)\end{aligned}\quad (5.7)$$

Les points d'équilibre sont donnés par :

$x_2 = 0$ et $\sin(x_1) = 0 \Rightarrow (0,0), (\pi, \theta)$, ces points correspondent au pendule dans la position de la verticale 'Up and Down'.

5.2.2 Notions de Stabilité : On dit qu'un système est

- Stable si, déplacé de sa position d'équilibre, il tend à y revenir.
- Instable, s'il tend à s'en écarter.

Nous avons les définitions suivantes

Stabilité Au Sens De Lyapunov (L- Stable) :

Soit l'équation différentielle suivante :

$$\begin{aligned}\dot{x} &= f(x, t) \quad \forall t \\ f(0, t) &= 0\end{aligned}\quad (5.8)$$

L'origine de ce système est stable au sens de Lyapunov si

$$\forall t_0, \forall \varepsilon, \exists \eta \text{ tels que } \|x(t_0)\| < \eta \Rightarrow \|x(t)\| < \varepsilon \quad \forall t > t_0$$

C'est à dire : l'origine d'un système est (L- Stable) si seulement si une faible perturbation dans les conditions initiales entraîne une faible perturbation de la trajectoire ultérieure.

Stabilité Asymptotique Au Sens De Lyapunov :

L'origine du système précédent est asymptotiquement stable au sens de Lyapunov si :

Il est (L- Stable) et $\exists R$ tel que $\|x(t_0)\| < R \Rightarrow \|x(t)\| \rightarrow 0$ quand $t \rightarrow \infty$

Remarque. La stabilité asymptotique comporte évidemment la stabilité simple. La stabilité asymptotique est une notion plus restrictive que la stabilité tout court.

Tout équilibre asymptotiquement stable est stable ; mais inversement un équilibre peut être sans l'être asymptotiquement.

5.2.3 Stabilité Locale est Globale

Pour un système non linéaire, un point d'équilibre possède un domaine de stabilité, habituellement défini comme l'ensemble des conditions initiales ' $x(0)$ ' :

Le système après avoir été écarté reviendra à sa position d'équilibre d'une manière telle que le point représentant son état reste à l'intérieur du dit domaine. Mais s'il est écarté au de la de ce domaine, il se rendra à une autre position d'équilibre, ou tendra vers l'infini.

Le problème de stabilité non linéaire comporte ainsi deux étapes :

- Déterminer la stabilité ou l'instabilité de la position d'équilibre : problème de la stabilité locale.
- Si l'équilibre est stable, trouver le domaine de stabilité ; problème de la stabilité globale

Remarque. Dire qu'un équilibre possède la stabilité locale, c'est affirmer seulement que la position d'équilibre est stable.

Dire qu'il possède la stabilité globale, c'est affirmer que toutes les valeurs que peuvent prendre les variables appartiennent au domaine de stabilité.

La stabilité globale est dite illimitée si ce domaine comprend tout l'espace.

5.3 Analyse de la stabilité par Lyapunov

Pour la commande d'un système, la première question importante qui se pose concerne sa stabilité. Lyapunov a introduit deux méthodes :

- La méthode indirecte : dite aussi la méthode de linéarisation elle s'intéresse à la stabilité locale autour d'un point d'équilibre.
- La méthode directe : c'est une généralisation du concept d'énergie des systèmes mécaniques. Elle s'intéresse à la stabilité globale.

5.3.1 La méthode indirecte

La méthode indirecte de Lyapunov concerne la stabilité locale des systèmes non linéaires. Considérons le système autonome :

$$\begin{aligned}\dot{x} &= f(x) \\ f(0) &= 0\end{aligned}\tag{5.9}$$

où : $f(x)$ est une fonction continue et dérivable.

Le développement de Taylor de $f(x)$ au voisinage de $x=0$ s'écrit donc :

$$\dot{x} = f(0) + \left. \frac{\partial f}{\partial x} \right|_{x=0} x + f_h(x)\tag{5.10}$$

Où $f_h(x)$ contient les termes d'ordre supérieur lesquels sont négligés au voisinage de 0. Notons que le premier terme du développement de Taylor $f(0)$ est nul, car le '0' est un point

d'équilibre. Utilisons la matrice constante $A = \left. \frac{\partial f}{\partial x} \right|_{x=0}$.

On aura :

$$\dot{x} = Ax \quad (5.11)$$

Cette expression est l'approximation linéaire du système original non linéaire au tour du point d'équilibre '0'. Le résultat de cette méthode peut s'énoncer comme suit :

- Si le système linéarisé est asymptotiquement stable alors le système non linéaire est localement asymptotiquement stable.
- Si le système linéarisé est stable alors on ne peut rien dire sur la stabilité ou instabilité du système non linéaire.
- Si le système linéarisé est instable alors le système original est instable aussi.

5.3.2 La méthode directe

Le principe de base de cette méthode est l'extension mathématique de l'observation suivante : *'si l'énergie totale d'un système mécanique (ou électrique) décroît continuellement, alors le système doit éventuellement tendre vers un point d'équilibre'* ; ce qui permet de tirer des conclusions sur la stabilité du système.

L'idée de cette méthode c'est de construire pour le système une fonction (scalaire dérivable), qui est l'image de l'énergie du système puis de montrer que cette fonction est décroissante au cours du temps. La principale caractéristique de cette méthode c'est de permettre d'analyser la stabilité d'un système non linéaire sans connaître explicitement les solutions des équations différentielles qui le décrivent.

Pour tester la stabilité d'un système non linéaire, Lyapunov a annoncé un théorème qui est le suivant :

Le système autonome : $\dot{x} = f(x)$; aura un point d'équilibre $x=0$ globalement asymptotique stable s'il existe une fonction scalaire réelle $V(x)$ continue avec une dérivée partielle par rapport au temps \dot{V} continue ayant les propriétés suivantes :

- $V(0) = 0$
- $V(x) > 0 \quad \forall x \neq 0$
- $\lim V(x) = \infty$ (radialement non bornée) pour $\|x\| \rightarrow \infty$

Remarque : Dans les notions de stabilité on rencontre souvent deux notions qui sont la stabilité et la stabilité asymptotique. Ces deux notions de stabilité et stabilité asymptotique ont une traduction mathématique qui est la suivante :

- S'il existe une fonction V définie positive dont la dérivée soit négative pour tout (x, t) alors le système est stable au sens de Lyapunov.
- S'il existe une fonction V définie positive dont la dérivée vérifie tout (x, t) :

$$\dot{V}(x, t) < -\gamma(\|x\|) \leq 0$$

Où $\gamma(\cdot)$: est une fonction continue dérivable ; alors le système est asymptotiquement stable.

Exemple 5. Reprenons l'exemple précédent avec $V = x_1^2 + x_2^2 + x_3^2$. Dans ce cas, on a

$\dot{V} = -2(x_1 - 2x_2)^2 + x_3^2$. Expression qui est semi négative, on peut donc affirmer la stabilité et non pas son caractère asymptotique.

Remarque. Le dernier exemple illustre le fait qu'il est possible d'obtenir plusieurs fonctions de Lyapunov, conduisant à des résultats non identiques.

5.3.3 Le principe d'invariance de Lasalle

Si pour un certain x par où la trajectoire du système passe, on a $\dot{V}(x) = 0$, alors la trajectoire pourra ne pas évoluer vers l'origine. C'est à dire, si $\dot{V}(x) = 0$ dans un domaine on peut conclure uniquement que le système est stable (les trajectoires ne sortent pas d'un certain domaine). Et le théorème suivant est utilisé pour conclure sur la stabilité asymptotique :

Théorème de Lasalle : Si il existe une fonction V dont la dérivée $\dot{V} \leq 0$, alors les trajectoires convergent vers Γ l'ensemble invariant maximal contenu dans l'ensemble des points vérifiant $\dot{V} = 0$

Exemple 6. Soit le système :

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= 2bx_1 - 3x_1^2 - ax_2 \end{aligned} \quad (5.15)$$

Avec $a > 0$, $b > 0$, soit la fonction $V = \frac{1}{2}x_2^2 + bx_1^2 + x_1^3$ elle est définie positive pour $x_1 \geq 0$. On

a $\dot{V} = -ax_2^2$, qui est semi négative on peut dire que le système est stable. Par contre, si on utilise le théorème de Lasalle, on est amené aux considérations suivantes : soit $\Gamma = \{x / \dot{V} = 0\} = \{x / x_2 = 0\}$ qui est donc l'ensemble des points vérifiant $x_1(-2b - 3x_1) = 0$ qui est donc l'axe x_1 . L'ensemble Γ se réduit donc à l'origine. Et on conclut que le système est asymptotiquement stable.

Remarque Si on n'a pas pu trouver une fonction $V(x)$ vérifiant le théorème Lyapunov, on ne peut pas conclure que le système est instable.

5.4 Application à la commande des robots :

5.4.1 Introduction [2]

Les effets du couplage non linéaire entre les articulations sont vus comme des perturbations de chaque système individuellement. En réalité, l'équation dynamique du robot manipulateur forme un système complexe non linéaire multivariable fortement couplé. Dans le présent chapitre on traite le problème de la loi de commande dans le contexte de la commande non linéaire multivariable. Cette approche nous laisse faire une analyse plus rigoureuse des performances de la commande des systèmes, et nous laisse aussi concevoir des lois de commande non linéaires robustes qui garantissent la stabilité globale et la poursuite de trajectoires arbitraires.

5.4.2 Synthèse d'une première loi de commande [2]

Dans la commande par la méthode de Lyapunov on peut avoir plusieurs lois de commande qui assurent la stabilité et la poursuite de trajectoire, on donne premièrement une discussion mathématique rigoureuse de l'application de la commande Lyapunov d'un robot rigide. Laissons-nous reformuler les équations dynamiques non linéaires du manipulateur en une forme convenable. Les équations du mouvement du robot sont données par :

Equation mécanique du robot :

$$\sum_{j=1}^n a_{jk}(q) \ddot{q}_j + \sum_{i,j=1}^n c_{ijk}(q) \dot{q}_i \dot{q}_j + g_k(q) = \tau_k \quad (5.16)$$

Equation dynamique de l'actionneur :

$$Jm_k \ddot{\theta} m_k + (Bm_k + KbKm/R) \dot{\theta} m_k = Km/R v_k - r_k \tau_k \quad (5.17)$$

Où : Jm_k est le moment d'inertie de l'induit, r_k est le facteur de réduction, R est la résistance d'induit.

En Divisant (5.17) par r_k et en utilisant le fait que :

$$\theta m_k = \frac{1}{r_k} q_k \quad (5.18)$$

On écrit l'équation (5.17) comme :

$$\frac{1}{r_k} Jm \ddot{q}_k + \frac{1}{r_k} B \dot{q}_k = \frac{km}{r_k R} v_k - \tau_k \quad (5.19)$$

Quand $B = Bm_k + KbLm/R$ en substituant (5.19) dans (5.16) on aura :

$$\frac{1}{r_k} Jm \ddot{q}_k + \sum_{j=1}^n a_{jk} \ddot{q}_j + \sum_{i,j=1}^n c_{ijk} \dot{q}_i \dot{q}_j + \frac{1}{r_k} B \dot{q}_k + g_k = \frac{Km}{r_k R} v_k \quad (5.20)$$

Sous forme matricielle ces équations peuvent s'écrire comme suit :

$$A(q) \ddot{q} + C(q, \dot{q}) \dot{q} + B \dot{q} + g(q) = u \quad (5.21)$$

Où $A(q)$ est la matrice d'inertie. Le vecteur $g(q)$ et la matrice $C(q, \dot{q})$ sont définis par dans le chapitre 2 et le vecteur d'entrée u a pour composantes $u_k = \frac{Km}{r_k R} v_k$. On constate que u_k a les unités de couple.

Considérons la fonction de Lyapunov suivante :

$$V = 1/2 \dot{\tilde{q}}^T A(q) \dot{\tilde{q}} + 1/2 \tilde{q}^T K_p \tilde{q} \quad (5.22)$$

Où : $\tilde{q} = q_d - q$.

Le premier terme de (5.22) est l'énergie cinétique du robot et le second terme traduit le retour d'information proportionnel $K_p \tilde{q}$. On constate que V représente l'énergie cinétique total que résulterait si les actionneurs articulaires étaient remplacés par des ressorts ayant pour raideurs K_p est pour positions d'équilibre à q^d . Cependant V est une fonction positive à l'exception du but à atteindre $q = q^d, \dot{q} = 0$ pour lequel V est nulle. L'idée est de montrer que le long de n'importe quelle trajectoire du robot, la fonction V est décroissante et tend vers zéro (0).

Ceci implique que le robot s'approche de la configuration du but désiré.

Pour montrer ceci on constate que tant que q^d est constante, la dérivée temporelle de V est donnée par :

$$\dot{V} = \dot{q}^T A(q)\ddot{q} + 1/2 \dot{q}^T A(q)\dot{q} - \dot{q}^T K_p \tilde{q} \quad (5.23)$$

En résolvant pour $A(q)\ddot{q}$ dans (5.21) avec $g(q) = 0$ et on substituant l'expression résultante dans (5.23) il en résulte :

$$\begin{aligned} \dot{V} &= \dot{q}^T (u - C(q, \dot{q}) - B\dot{q}) + 1/2 \dot{q}^T A(q)\dot{q} - \dot{q}^T K_p \tilde{q} \\ &= \dot{q}^T (u - B\dot{q} - K_p \tilde{q}) + 1/2 \dot{q}^T (\dot{A}(q)\dot{q} - 2C(q, \dot{q}))\dot{q} \\ &= \dot{q}^T (u - B\dot{q} - K_p \tilde{q}) \end{aligned} \quad (5.24)$$

Où dans la dernière égalité on a utilisé le fait que (théorème) la matrice $\dot{A}(q)\dot{q} - 2C(q, \dot{q})$ est antisymétrique [2].

Une loi de commande indépendante à une articulation peut être écrite sous forme vectorielle comme :

$$u = K_p \tilde{q} - K_d \dot{q} \quad (5.25)$$

Où $\tilde{q} = q^d - q$ est l'erreur entre le déplacement articulaire désiré q^d et le déplacement articulaire actuel q , et K_p, K_d sont des matrices diagonales des gains (positif) proportionnelle et dérivée, respectivement. On montre d'abord que, en l'absence de la gravité, que la loi de commande parvient à une attraction de la position articulaire désirée. En effet, ceci reproduit le résultat dérivé récemment, mais plus rigoureux, dans le sens que les équations non linéaires du mouvement (5.16) ne sont pas approximées par une perturbation constante mais sont directement manipulées.

En substituant la loi de commande (5.25) pour u dans ce qui précède on trouve :

$$\dot{V} = -\dot{q}^T (K_d + B)\dot{q} \leq 0 \quad (5.26)$$

L'analyse précédente prouve que V est décroissante tant que \dot{q} n'est pas nulle. Ceci, lui-même n'est pas assurément pour prouver le résultat désiré car il est concevable que le manipulateur puisse atteindre une position pour laquelle $\dot{q} = 0$ mais $q \neq q^d$. Pour montrer que ceci ne peut jamais se produire, on peut s'inspirer du théorème de LaSalle. On suppose que $\dot{V} \equiv 0$. Alors (5.26) implique que $\dot{q} \equiv 0$ et par conséquent $\ddot{q} \equiv 0$. De l'équation du mouvement

$$A(q)\ddot{q} + C(q, \dot{q})\dot{q} = -K_p \tilde{q} - K_d \dot{q} \quad (5.27)$$

On doit avoir donc

$$-K_p \tilde{q} = 0 \quad (5.28)$$

Ce qui implique que $\tilde{q} = 0$, $\dot{q} = 0$. Le théorème de LaSalle a prouvé donc que le système est asymptotiquement stable.

En cas où il y a des termes gravitationnels présents dans (5.21) l'équation (5.24) doit être modifiée et devient comme suit :

$$\dot{V} = \dot{q}^T (u - g(q) - B\dot{q} - K_p \tilde{q}) \quad (5.29)$$

La présence du terme gravitationnel dans (5.29) signifie que la commande seule ne peut pas garantir l'attraction asymptotique. En pratique, il y aurait une erreur d'état ou offset. En adoptant que le système en boucle fermée est stable la configuration du robot q qui est accomplie satisfèrait

$$K_p (q^d - q) = g(q) \quad (5.30)$$

L'interprétation physique de (5.30) est que la configuration q doit être telle que le moteur génère un état invariant « couple de maintien » en anglais « holding torque » $K_p (q^d - q)$ suffisant de compenser la gravité $g(q)$. Cependant on voit que l'erreur peut être réduite en augmentant le gain de position K_p .

Pour éliminer cette erreur résiduelle on peut modifier la loi de commande de façon à comprendre le terme exprimant la gravité comme suit :

$$u = K_p \tilde{q} - K_d \dot{q} + g(q) \quad (5.31)$$

La loi de commande modifiée, en effet, élimine l'effet de la gravitation et on obtient ainsi la même équation que (5.26) précédente. La loi de commande (5.31) requiert une implémentation par microprocesseurs pour calculer à chaque instant le terme gravitationnel à partir de l'équation de Lagrange. Dans le cas où ces termes sont inconnus la loi de commande (5.31) ne serait pas applicable.

5.4.3 Synthèse d'une deuxième loi de commande [2] [8]

La loi de commande que nous venons de synthétiser ne fait appel au modèle dynamique, néanmoins on peut avoir d'autres lois de commandes basées sur la connaissance du modèle dynamique, on en décrit la loi commande suivante :

Le modèle dynamique du robot peut s'écrire comme suit :

$$A(q)\ddot{q} + V_m(\dot{q}, q)\dot{q} + F(\dot{q}) + G(q) = \tau \quad (5.32)$$

La commande de position basée sur Lyapunov est alors semblable à un couple précalculé :

$$\tau = A(q)\ddot{q}_r + V_m(\dot{q}_r, q)\dot{q}_r + F(\dot{q}_r) + G(q) + K_d s \quad (5.33)$$

Où :

$$\dot{q}_r = \dot{q}^d + K_p(q^d - q) \quad (5.34)$$

$$s = \ddot{q} + K_p \tilde{q} = \dot{q}_r - \dot{q} \quad (5.36)$$

Avec : K_p, K_d doivent être symétriques et définies positives.

On constate que la commande est filtrée, néanmoins elle est moins avantageuse vu la forte dépendance du modèle dynamique.

En appliquant la loi de commande de position (5,33) au robot, on aura la dynamique suivante :

$$A(q)\dot{s} + V_m(\dot{q}, q)s + K_d s = 0 \quad (5.37)$$

On peut alors prouver que $\lim_{t \rightarrow \infty} s = 0$

En choisissant la fonction de Lyapunov suivante $V = 1/2 s^T A(q)s$ qui est définie positive, on trouve que $s=0$ est asymptotiquement stable, puisqu'en utilisant la propriété que la matrice $\dot{A} - 2V_m(\dot{q}, q)$ est anti-symétrique, la dérivée de la fonction de Lyapunov est définie négative.[2]

En effet,

$$\begin{aligned} \dot{V} &= s^T A(q)\dot{s} + 1/2 s^T \dot{A} s \\ &= -s^T K_d s + 1/2 s^T [\dot{A} - 2V_m(\dot{q}, q)]s \\ &= -s^T K_d s < 0 \end{aligned} \quad (5.38)$$

Si on choisit K_p diagonale positive, selon l'équation :

$$\dot{\tilde{q}}_i + k_{pi} q_i = s_i \quad i = 1, \dots, n \quad (5.39)$$

Parce que $\lim_{t \rightarrow \infty} s = 0$ les erreurs de position et de vitesse convergent également à 0.

On peut alors approximer la dynamique par :

$$\dot{\tilde{q}}_i + k_{pi} q_i = 0 \quad i = 1, \dots, n \quad (5.40)$$

Le temps de réponse T_r peut alors être imposé en posant $K_{pi} = 3/T_r$

5.5 SIMULATIONS ET INTERPRETATIONS

Simulation 1 : (trajectoire planaire)

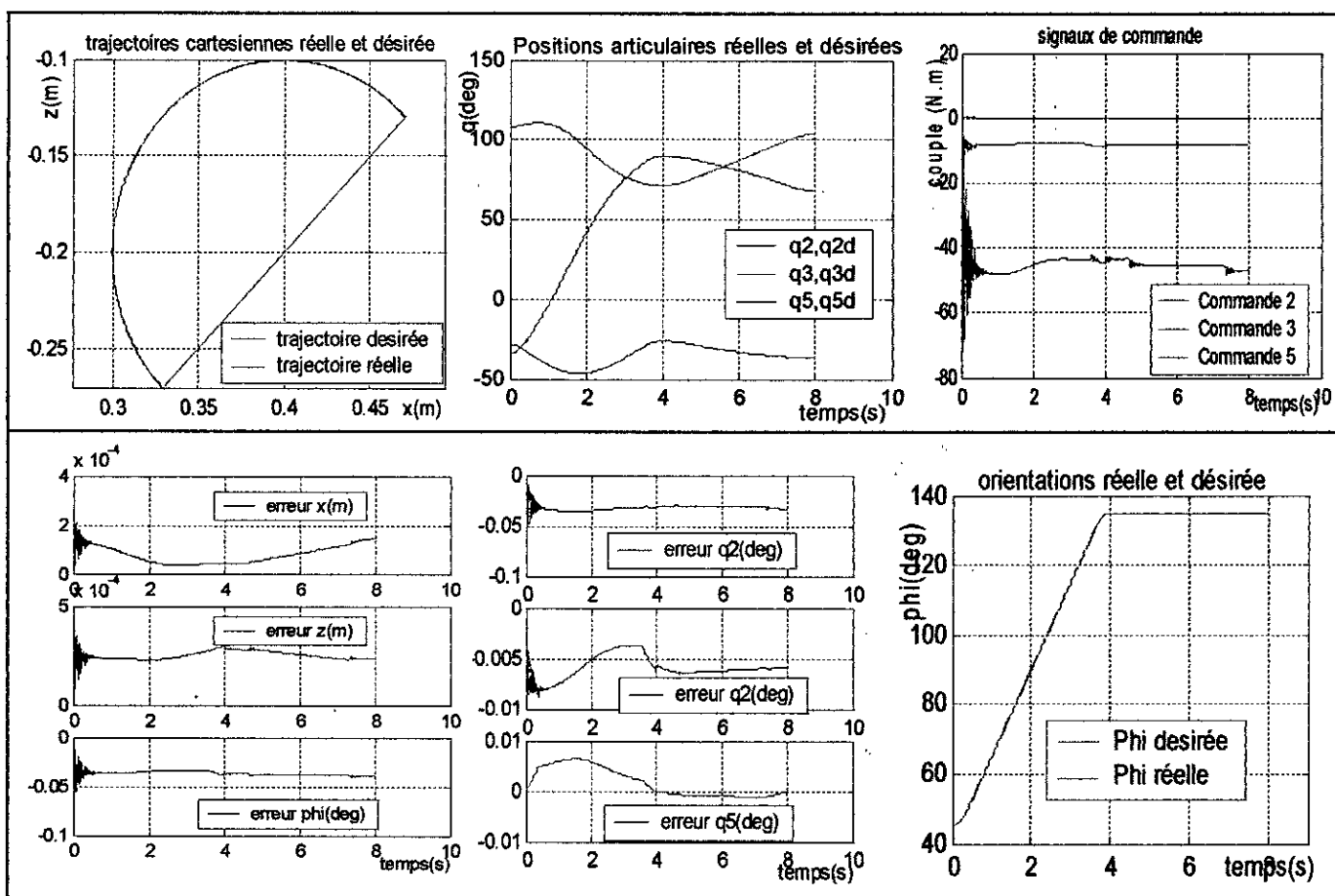


Fig.5.1 Réponse du robot en position et orientation

On observant les simulations ci-dessus, on peut dire qu'une bonne poursuite de la trajectoire désirée est réalisée (les trajectoires réelle et désirée sont presque confondues), ce qui est confirmé par les erreurs de poursuite cartésiennes et articulaires.

Au démarrage, l'erreur oscille entre 0.01mm et 0.2 mm, pour la position opérationnelle et entre 0.01° et 0.03° en ce qui concerne l'orientation de l'outil terminal et se stabilise à partir de 0.2 sec.

Les couples appliqués présentent de fortes oscillations au départ et qui disparaissent au bout de 0.3 sec, pour vaincre l'inertie du système au démarrage.

Les couples appliqués sont admissibles par les actionneurs utilisés et loin des limites maximales.

Simulation 2 (trajectoire spatiale):

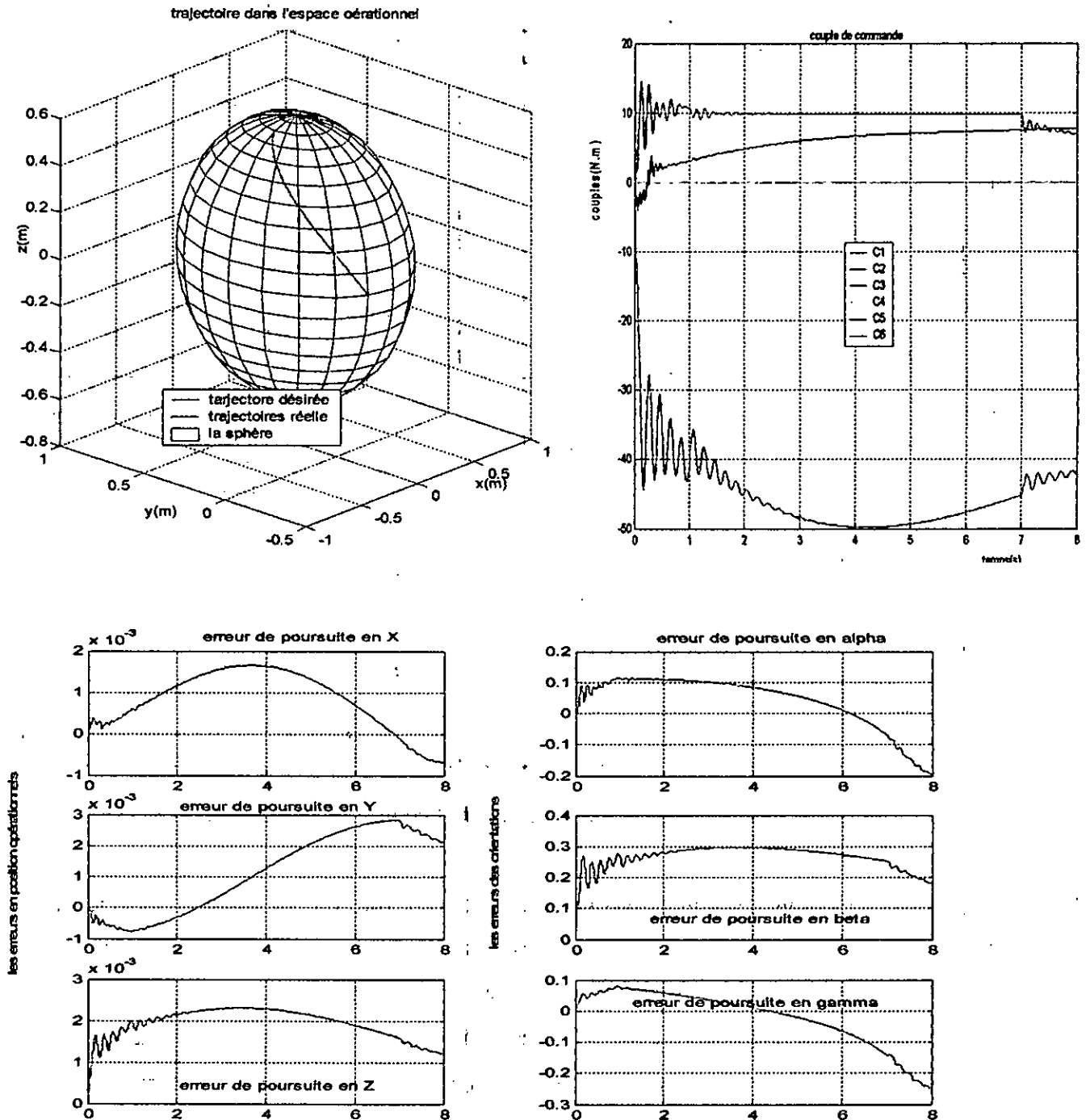
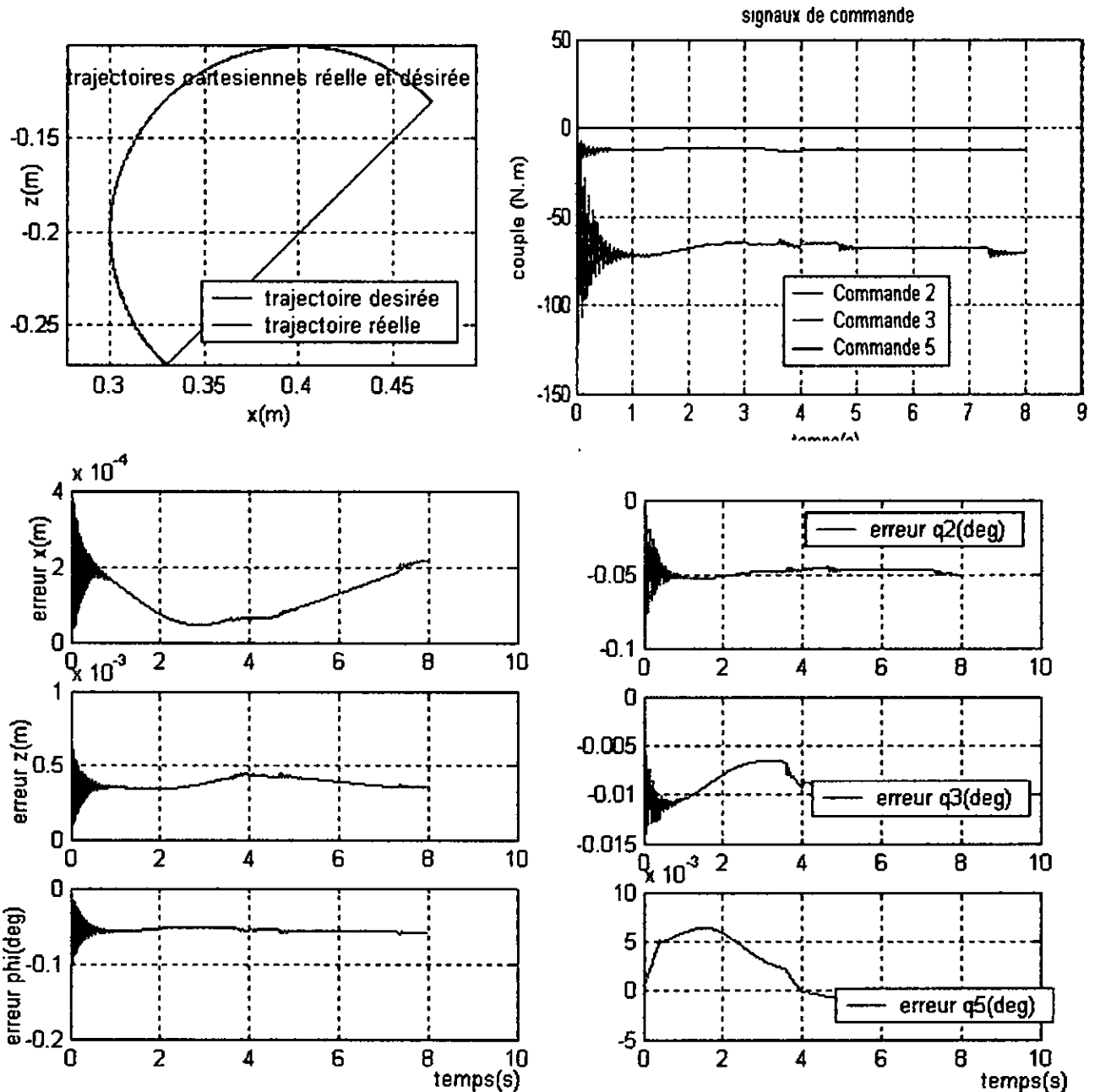


Fig.5.1 Réponse du robot en position et orientation

On constate qu'une bonne poursuite de la trajectoire désirée est réalisée (les trajectoires réelle et désirée sont presque confondues). Les erreurs sont de l'ordre 1 mm pour ce qui est de la position et 0.2° pour l'orientation.

Les couples appliqués sont admissibles par les actionneurs utilisés et loin des limites maximales.

Simulation3 : Cette partie concerne l'étude de la robustesse de la commande par la méthode de Lyapunov avec une erreur de modélisation de 50%.



On constate que la poursuite est réalisée malgré l'incertitude sur le modèle dynamique avec une commande admissible un peu élevée d'environ 70 N.m pour l'articulation 2, par rapport aux simulations précédentes (modèle certain).

CONCLUSION

D'après les simulations qu'on vient de présenter, on peut confirmer l'efficacité et la robustesse de la commande par la méthode de Lyapunov en terme de poursuite de trajectoire, rapidité et précision. Ceci revient au fait que la commande est à grand gain et ne dépend pas du modèle dynamique à l'inverse de la commande par découplage non linéaire qui, on l'a déjà vu, nécessitait une la connaissance du modèle dynamique avec exactitude pour ne pas détériorer les performances exigées..

CONCLUSION GENERAL

Bien que la commande de la plus part des robots industriels actuels soit encore conçue à partir des théories de l'automatique linéaire, des méthodes plus avancées, tenant compte du caractère non linéaire des structures articulées doivent être envisagées pour les applications exigeant de grandes performances dynamiques comme la précision dynamique, et la rapidité.

Pour cette raison, nous avons au cours de ce mémoire donné le principe de trois méthodes qui répondent à ces exigences : la commande par découplage non linéaire comme approche idéale pour la commande des robots, la commande basée sur le principe de la stabilité de Lyapunov des systèmes automatiques (PD) et la commande par découplage non linéaire robustifiée par logique floue.

Nous avons supposé que le système et la commande sont continus et nous avons négligé les dynamiques des capteurs et des actionneurs. Dans la pratique, la commande se réalise, par ordinateur, il faut tenir compte d'un retard pour l'acquisition des informations et pour le calcul des consignes. Cependant, pour pouvoir utiliser ces approches de commande, il faut que la période d'échantillonnage soit suffisamment petite par rapport à la bande passante mécanique du robot. Pour fixer un ordre de grandeur, une valeur autour d'1KHz suffit. On note que choisir une fréquence élevée permet d'augmenter le gain donc rendre la commande plus robuste [3].

Dans toutes les lois de commande étudiées, nous avons considéré qu'au moins les positions et vitesses opérationnelles sont disponibles. En général, les capteurs de vitesse articulaire fournissent des signaux fortement bruités, un problème qu'on résout par dérivation numérique de la position ou estimation de la vitesse en fonction de la position et couples appliqués aux articulations du bras manipulateur.

Dans ce travail, une étude comparative a été effectuée entre les différentes approches de commande appliquées à un bras manipulateur du type PUMA 560.

Les résultats de l'application de la commande par découplage non linéaire ont mis bien en évidence le problème du calcul en ligne du modèle dynamique complet avec certitude, ce qui est difficile à mettre en œuvre pratiquement d'où l'approche de robustification qui

consistait à compenser les termes incertains difficiles à modéliser par un terme qui dépendait d'une fonction coût et d'un autre terme de glissement à adaptation floue pour éliminer le phénomène d'oscillation à haute fréquence de la commande dit broutement (*chattering* en Anglais) qui détériore les performances. Néanmoins, on peut utiliser d'autres techniques pour remédier au problème qui se basent sur l'adoucissement de la commande par une fonction non linéaire comme la fonction saturation.

Quant à la commande basée sur le principe de Lyapunov, elle a montré son efficacité, en donnant de bons résultats, en termes de performances et de robustesse de la commande sans faire appel au calcul du modèle dynamique, car elle n'en tient pas en compte explicitement comme dans le cas de la commande par découplage non linéaire. On note également une meilleure robustesse du système dont les essais (incertitude sur la matrice d'inertie, élimination des termes de Coriolis et de centrifugation) n'agissent que très peu sur la poursuite de trajectoire.

Concernant les perspectives possibles à partir de ce travail, on propose l'application d'autres commandes robustes comme la commande par logique floue ou la commande passive ainsi que d'appliquer ces commandes aux manipulateurs redondants.

Références Bibliographiques

- [1] **JOHN J. CRAIG** « Introduction to Robotics: Mechanics and Control », 2nd ed, Addison-Wesley Publishing Company, Canada, 1989.
- [2] **MARK W.SPONG AND M. VIDYASAGAR** « Robot Dynamics and Control », Quinn-Woodbine, USA, 1981.
- [3] **WISAMA KHALIL ET ETIENNE DOMBRE** « Modélisation Identification et Commande des robots », 2^e édition, Hermes Science Publications, Paris, 1999.
- [4] **WEIPING LI AND WILLIAM A.WOLOVICH** « Robotics :Basic and Design »,
- [5] **SCHILING, R.J** « Fundamentals of Robotics : analysis and Control », Prentice Hall, 1990.
- [6] **BRAIN ARMSTRONG, OUSSAMA KHATIB AND JOEL BURDICK** «The Explicit Dynamique Model and Inertial Parameters of the Puma 560 Arm », In Proc. 1986 IEEE Int. Conf. On Robotics and Automation, S an Francisco, CA.
- [7] **BUCK LEY, J.J** « Fundamantal of Robotics : Analyse and Control » J.Wiley. 1989.
- [8] **JEAN-JACQUES E.SLOTINE** « Applied nonlinear control ».
- [9] **MICHEL COTSFTIS** « Comportement et Contrôle des Systèmes Complexes », Diderot Editeur, Art et Sciences, Pris, 1997.
- [10] **L.PRALY** « Fonctions de Lyapunov et Stabilisation de Point d'Equilibre », Ecole Nationale Supérieur des mines de Paris, Centre Automatique et système, 2002.
- [10] **MARK W SPONG** «Motion Control of Robot Manipulators», The Coordinated Science laboratory; University of Illinios at Urbana-Champaingn; USA.
- [11] **MOUNIRE BEN GHALIA AND ALI T.ALOUANI** « A Robust Trajectory Tracking Control of Industrial Robot Manipulators Using fussy logic », IEEE Trans. Robotics and Automation,1995.
- [12] **LFA'96** « Rencontres francophones sur la logique floue et ses applications », Cépaduès-Editions, Toulouse-France, 1996.
- [13] **BRAHIM CHARKI** «Commande des Robots Manipulateurs par Retour d'Etat Estimé », Thèse doctorat, Ecole Centrale de Nantes, 1996.
- [14] **BUHLER , H.** « Réglage Par Logique Floue ». PPUR,1994.
- [15] **MUHAMMAD SUZURI HITAM** « Fuzzy Logic Control of an Idustrial Robot », Proceeding of the 2001 IEEE, International Conference on robotics and Automation, seoul, Korea - May 21-26, 2001.
- [16] **WUWEI CHEN, J K.MILLS, J.CHU, D.SUN** « A Fuzzy Compensator for Uncertainty Of industrial Robots, IEEE Trans. On Automatic Control, 2001, pp 257-262.

Système d'angles d'EULER :(Z-Y-Z)

L'orientation d'un repère R_n dans le repère R_0 est déterminée par la spécification de trois angles α, β, γ correspondant à trois rotations successives comme il est schématisé sur la figure ci-dessous.

On fait une première rotation autour de Z avec un angle α , puis une deuxième rotation autour de Y avec un angle β , et enfin une troisième rotation autour de Z avec un angle γ .

Le plan (X_n, Y_n) coupe le plan (X_0, Y_0) suivant une droite ON appelée *ligne nodale*, perpendiculaire aux axes Z_0 et Z_n . les angles d'EULER sont définies de la façon suivante :

- α (précession) : angle entre Y_0 et ON, avec $0 \leq \alpha < 360^\circ$;
- β (nutation) : angle entre Z_0 et Z_n , avec $0 \leq \beta < 180^\circ$;
- γ (rotation propre) : angle entre ON et Y_n , avec $0 \leq \gamma < 360^\circ$;

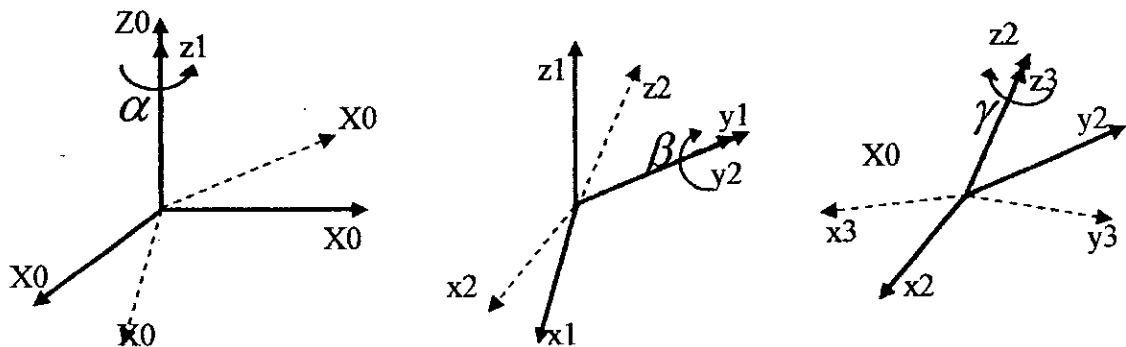


Fig. 1 Les angles d'EULER

Les angles α, γ sont mesurés dans le sens donné par la règle du tire-bouchon, respectivement autour de Z_0 et Z_n . la composition de ces trois rotations permet de calculer la matrice d'orientation :

$$\begin{aligned}
 {}^0R_n &= \text{rot}(z, \alpha) \text{rot}(y, \beta) \text{rot}(z, \gamma) \\
 &= \begin{bmatrix} c\alpha c\beta c\gamma & -c\alpha c\beta s\gamma - s\alpha c\gamma & c\alpha s\beta \\ s\alpha c\beta c\gamma + c\alpha s\gamma & -s\alpha c\beta s\gamma + c\alpha c\gamma & s\alpha s\beta \\ -s\beta c\gamma & s\beta s\gamma & c\beta \end{bmatrix} \quad (1)
 \end{aligned}$$

La matrice de rotation 3x3 s'obtient à partir des transformations homogènes lors du calcul du modèle géométrique direct, elle est sous la forme suivante :

$${}^0R_n = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \quad (2)$$

Pour extraire les angles d'EULER (Z-Y-Z) à partir de la matrice de rotation 0R_n (par identification), trois cas de figure sont à envisager :

Si $s\beta \neq 0$, alors :

$$\begin{aligned} \beta &= A \tan 2(\sqrt{r_{31}^2 + r_{32}^2}, r_{33}), \\ \alpha &= A \tan 2(r_{23}/s\beta, r_{13}/s\beta), \\ \gamma &= A \tan 2(r_{32}/s\beta, -r_{31}/s\beta), \end{aligned} \quad (3)$$

Malgré qu'il existe une autre solution, en utilisant la racine carrée positive dans la formule de β , on calcule toujours la solution pour laquelle $0 \leq \beta < 180^\circ$; si $\beta = 0$ ou 180° , la solution ne peut s'obtenir par (2).

Si $\beta = 0$, alors :

$$\begin{aligned} \beta &= 0; \\ \alpha &= 0; \\ \gamma &= A \tan 2(-r_{12}, r_{11}); \end{aligned} \quad (4)$$

Si $\beta = 180^\circ$, alors :

$$\begin{aligned} \beta &= 180^\circ; \\ \alpha &= 0; \\ \gamma &= A \tan 2(r_{12}, -r_{11}); \end{aligned} \quad (5)$$

ملخص:

العمل المقدم من خلال هذه المنكرة يتمثل في دراسة مشكلة التحكم في ذراع الآلة في الفضاء العملي، قمنا بعرض مختلف طرق تحديد المسارات العملية، واستخراج النماذج الهندسية، الحركية أو الديناميكية المباشرة و الغير المباشرة. اقترحنا ثلاث أنظمة تحكم لضمان مسابرة المصارع المرغوب في الموضوع و الإتجاه العمليين و هي كالتالي: الفصل اللاخطي (العزم المحسوب)، التحكم الصلب باستعمال المنطق المبهم و قانون التحكم حسب مبدأ ليابونوف. كل هذا وفقاً لبنيتين مختلفتين الأولى محسوبة في فضاء المفاصل و الثانية في فضاء العمل. قمنا بالمحاكاة على ذراع الآلة من نوع بيما 560.

كلمات مفتاحية: ذراع الآلة (بيما 560)، الفضاء العملي، المصارع العملي، النماذج الهندسية، الحركية، الديناميكية، العزم المحسوب، مبدأ ليابونوف، المنطق المبهم.

RESUME :

Le travail présenté dans ce mémoire est une étude de la commande d'un bras manipulateur dans l'espace opérationnel. Nous avons présenté les différentes méthodes de génération d'une trajectoire opérationnelle. Nous avons déterminé les modèles : géométrique, cinématique et dynamique (direct et inverse). Nous avons appliqué trois commandes pour commander le robot en position et orientation qui sont : commande par découplage non linéaire, couple calculé robustifié par logique floue et la commande basée sur le principe de Lyapunov. Selon deux structures de commande calculées dans les espaces articulaire et opérationnel. Les simulations ont été faites sur le bras de robot PUMA 560.

Mots clés : bras manipulateur (PUMA560), espace opérationnel, trajectoire opérationnelle, les modèles géométrique, cinématique et dynamique, couple calculé, principe de Lyapunov, la logique floue.

ABSTRACT :

The work presented in this thesis is a study of the control of an arm manipulator in cartesian space. We presented the various methods of cartesian trajectory generation. We determined the models: geometrical, kinematic and dynamic (direct and inverse). We applied three control laws to control the robot in position and orientation which are: computed torque, computed torque robusted by fuzzy logic and the control based on the principle of Lyapunov. According to two structures of control calculated in joint and cartesian spaces, the simulations were made on the arm of robot PUMA 560.

Key words: arm manipulator (PUMA 560), cartesian space, cartesian trajectory, the models kinematic and dynamic, computed torque, lyapunov's principle, fuzzy logic.